Computer Systems Validation

Quality Assurance, Risk Management, and Regulatory Compliance for Pharmaceutical and Healthcare Companies

Computer Systems Validation

Quality Assurance, Risk Management, and Regulatory Compliance for Pharmaceutical and Healthcare Companies

EDITOR

Guy Wingate

Interpharm/CRC

Boca Raton London New York Washington, D.C.

Library of Congress Cataloging-in-Publication Data

Computer systems validation : quality assurance, risk management, and regulatory compliance
for pharmaceutical and healthcare companies / edited by Guy Wingate.
p. cm.
Includes bibliographical references and index.
ISBN 0-8493-1871-8 (alk. paper)
1. Pharmaceutical industry—Management. 2. Pharmaceutical industry—Data processing.
3. Health facilities—Risk management. 4. Risk management—Data processing. I. Wingate, Guy. II. Title.

HD9665.5.C663 2003
338.4'76151'0285—dc22 2003062090

This book contains information obtained from authentic and highly regarded sources. Reprinted material is quoted with permission, and sources are indicated. A wide variety of references are listed. Reasonable efforts have been made to publish reliable data and information, but the author and the publisher cannot assume responsibility for the validity of all materials or for the consequences of their use.

Direct all inquiries to CRC Press LLC, 2000 N.W. Corporate Blvd., Boca Raton, Florida 33431.

Trademark Notice: Product or corporate names may be trademarks or registered trademarks, and are used only for identification and explanation, without intent to infringe.

Visit the CRC Press Web site at www.crcpress.com

No claim to original U.S. Government works
International Standard Book Number 0-8493-1871-8
Library of Congress Card Number 2003062090
Printed in the United States of America 1 2 3 4 5 6 7 8 9 0
Printed on acid-free paper

For Sarah, Katherine, Robert, and Edward

Validation should be viewed as an integral part of the overall computer system's life cycle. It should promote improved process control and not bureaucracy. Good practice and common sense should prevail.

Foreword

Computer technology is all pervasive. It hides behind the smallest button on domestic appliances, and it is found in smart cards and security devices, mobile phones, cash dispensers, PCs, integrated networks, process plant, automobiles, jumbo jets, and power plants. Computerized systems are everywhere. Automation is gathering complexity, innovation, and momentum, and we have to rely on it more and more in our everyday lives. The inexorable rise of computerized systems is also seen in the corporate strategies of pharmaceutical and healthcare companies calling for investment in new technology to improve business efficiency and competitive edge. When such technology is associated with high-risk public safety projects or the production and control of life-saving medicines or devices, we (businesses and regulators) need to know that it is reliable, quality assured, and validated. Easy to say, but the technology (and the terminology) is difficult to understand, let alone prove and qualify, if you are not an electronic systems engineer or a latent Einstein.

Pharmaceutical and healthcare companies have historically engineered their businesses to be profitable while ensuring that quality is built into their medicinal products or devices through the observance of GxPs (viz., GCPS, GLPs, GMPs, etc.), that essentially require computerized systems to be fully documented, defined as to functionality, quality assured, and validated. This book considers the requirements of the various international regulations, guides, and codes in historical perspective and leads the reader into business and project life-cycle issues and activities. This book is invaluable in that it bridges the gap between theory and practice, and it is supported by case studies from experienced professional practitioners and engineers who have had to meet the challenges of proving the quality, structural integrity, and validation of different systems in the "real world" (e.g., process control, QC analysis, integrated real-time applications, business information systems, and networks). The case studies are organized hierarchically from low-level instruments and PLCs through integration to higher-level proprietary electronic document and information management systems, and beyond.

Pharmaceutical and healthcare companies that invest in computerized systems need systems that are delivered on time and within budget, and that fulfull business functional and performance requirements. In their rush to place new products and versions on the market, however, computer software and systems suppliers rarely deliver error-free products. In fact, some two thirds of life-cycle costs can be incurred after delivery of the software and system to the users. Pharmaceutical and healthcare companies do not want lots of downtime, disruption, and escalating costs once a system has been delivered and implemented.[1,2] And, of course, in GxP applications, any deficiencies will be of particular interest during regulatory inspections.

Inspectors and investigators working for the different national regulatory bodies have to apply their national GxPs and regulations when assessing these systems. While these are published, they are not necessarily up to date and, as we all would acknowledge, they are often open to interpretation not only by different inspectors, depending on their background and training, but also on the particular computerized system and application. Regulators need to be satisfied that computerized systems installed in pharmaceutical and healthcare companies are fit for their intended purposes by considering the nature of the application, specifications, quality assurance of the development life-cycle activities, qualification, performance validation, in-use controls, accuracy, and reliability in the context of relevant GxPs. The increasing complexity of (integrated) proprietary computer systems, critical applications, project validation issues, and inspection findings have been considered before, together with the challenge for all parties (as ever) to apply sensible regulations and cost-effective good computer validation practices.[1,3,4]

The pharmaceutical and healthcare industries (including suppliers and developers) have reportedly had some difficulty in ensuring that these projects actually deliver the proposed business benefits, that the systems as built actually meet specifications, and that they are reliable and validated. This is quite apart from determining just how much and what type of validation evidence is required to satisfy the different regulatory bodies, in particular, the FDA. While the GAMP Guide[5] and, to some extent, the PDA 18 report[6] provide the latest interpretation of acceptable development and project guidance in this field (to ensure compliance with the needs of the principal regulatory bodies around the world), and TickIT provides a guide to software quality system construction and certification (using ISO 9001:1994)[7] there is a lack of papers on practical experiences from pharmaceutical and healthcare sector project teams seeking to implement new technology.

Today, both the industry and regulators have a much better understanding[8] of the ways and means to develop and validate computerized systems. Regulatory inspections now have more to do with risk-based assessments of what these systems are being used for in the context of broader GxP requirements rather than software and system validation per se. Inspectors[9] now rarely concentrate on "simply" inspecting computerized systems as an entity on sites; they are more often directly concerned with what the systems are being used for and how they are being used and controlled. Risk-based findings for noncompliances associated with computerized systems will often be linked with other chapters of the EU or PIC/S GMP apart from Annex 11. However, where a detailed inspection of a computerized system is indicated (from risk assessments or other inspections), then that can be arranged as a specialist exercise.

It is interesting to note the ongoing collaboration between ISPE and PDA[10,11] to publish guidance on electronic records and management and to influence opinion. It is to be hoped that the technological implementation of electronic records and electronic signature requirements worldwide will not be frustrated by a lack of understanding and agreement by all stakeholders of the real issues. Recognition must be given to the need for regulated users to have robust information security management practices and a risk-based approach applied to important relevant records and inspection compliance.

I believe this book will be welcomed by novices and experts, suppliers, developers, purchasers, and regulators alike for providing insight into the practical aspects of computerized systems and their life-cycle management. Many staffers assigned to validation projects could also benefit from sharing the experience of other practitioners. Whether you are looking for the missing piece of the jigsaw for your project or guidance on how to meet the regulations in a practical sense, then this information resource (which puts principles into practice) is a good place to start!

Anthony J. Trill
Senior Inspector
U.K. Medicines and Healthcare products Regulatory Authority (MHRA)

REFERENCES

1. Stokes, T., Branning, R.C., Chapman, K.G., Hambloch, H.J., and Trill, A.J. (1994), *Good Computer Validation Practices*: *Common Sense Implementation*, Interpharm Press, Buffalo Grove, IL.
2. Wingate, G.A.S. (1995), Computer Systems Validation: A Historical Perspective, *Pharmaceutical Engineering,* July/August, pp. 8–12.
3. Trill, A.J. (1995), EU GMP Requirements and the Good Automated Manufacturing Practice (GAMP) Supplier Guide for the Validation of Automated Systems for Pharmaceutical Manufacturing, *Pharmaceutical Engineering*, May/June, pp. 56–62.
4. Trill, A.J. (1996), An EU/MCA view of Recent Industry Guides to Computer Validation, Including GAMP 1996, PDA Technical Report 18 and the Validation Toolkit, in Proceedings of PIC/S Seminar, "Inspection of Computer Systems," Sydney, Australia, September.

5. ISPE (2001), *Good Automated Manufacturing Practice Guide for Validation of Automated Systems* (known as GAMP 4), available through www.ispe.org.
6. PDA (1995), The Validation of Computer Related Systems, Technical Report No. 18, *Journal of Pharmaceutical Science and Technology,* 49(1).
7. *TickIT Guide* (2000), A guide to Software Quality Management System Construction and Certification using ISO9001:2000, Issue 5.0, DISC/BSI TickIT Office, London.
8. Pharmaceutical Inspection Co-operation Scheme (2003), Good Practices for Computerised Systems in Regulated GxP Environments, Pharmaceutical Inspection Convention, PI 011-1, Geneva, August.
9. Medicines Control Agency (2002), “Top 10 GMP Inspection Issues,” MCA Seminar — London, September 24, A.J. Trill, “Computerised Systems and GMP — Current Issues.”
10. ISPE/PDA (2002): Good Practice and Compliance for Electronic Records and Signatures: Part 1 — Good Electronic Record Management (GERM), available through www.ispe.org.
11. ISPE/PDA (2001): Good Practice and Compliance for Electronic Records and Signatures: Part 2 — Complying with 21 CFR Part 11, Electronic Records and Electronic Signatures, available through www.ispe.org.

Preface

This book was prompted by an invitation to prepare a second edition of my first book on computer validation, *Validating Automated Manufacturing and Laboratory Applications*. On first reflection, it seemed that there might be little to update, but on further scrutiny I realized there have been considerable developments since 1997, not the least of which are new regulatory requirements for electronic records and electronic signatures. All this has led to a significant update with much new material. The basic structure of the book remains the same. In the first part (Chapters 1 through 18) I present the concepts and principles of computer system validation. The second part (Chapters 19 through 42) consists of case studies contributed by highly experienced industry experts examining the practical applications of these principles to different types of computer systems. The role of risk management throughout the life cycle of a computer system is emphasized not only for the benefit of patient/consumer safety but also in terms of cost-effectiveness. Throughout the book I have added real observations recently made by the FDA on the various topics being discussed.

I owe special thanks to those friends and colleagues who have provided invaluable discussions and explorations of computer validation principles and practices over the years. Validation in the real world is rarely simple and straightforward. My wish is that this book will enjoy equal success with its predecessor in sharing practical solutions to the many and varied challenges that face validation practitioners. In addition to the case study contributors listed, I would particularly like to add my thanks to Sam Brooks (ABB), Ellis Daw (GlaxoSmithKline), Paul D'Eramo (Johnson & Johnson), Howard Garston-Smith (Garston Smith Associates), Jerry Hare (GlaxoSmithKline), Scott Lewis (Eli Lilly), Takayoshi Matsumura (Eisai), Gordon Richman (EduQuest), David Selby (Selby-Hope), Amanda Willcox (GlaxoSmithKline), and Sion Wyn (Conformity).

I am hugely indebted to Ellis Daw, Chris Reid, and especially Howard Garston-Smith and Christine Andreasen (CRC Press) for their proofreading of chapters in this book. They have not only helped improve my grammar but have also prompted inclusions of additional material to better explain some of the validation concepts discussed.

Finally, once more I am indebted to Sarah, my wife, and our family for their love, patience, and support during the preparation of this book. Those who have read my two previous books on computer validation know that my books seem to coincide with the arrival of a new baby in our family. So it is with this book, and I am delighted to include Edward in the dedication.

Guy Wingate
Director, Global Computer Validation
GlaxoSmithKline

The Editor

Guy Wingate, Ph.D., is Director, Global Computer Validation, for GlaxoSmithKline's Manufacturing and Supply. He is responsible for policy and guidelines for computer validation and electronic records/signatures, and for managing compliance support to corporate projects (covering laboratory systems, control systems, IT systems, and computer network infrastructure) for 100 manufacturing sites.

Dr. Wingate graduated from Durham University, Durham, U.K. with B.Sc., M.Sc., and Ph.D. degrees in computing, micro-electronics, and engineering, respectively. He was recruited to join GlaxoWellcome in 1998 to establish a computer validation department serving the U.K. secondary manufacturing sites. Previously, Dr. Wingate was Validation Manager at ICI Eutech. A well-known speaker on computer validation, Dr. Wingate has published two previous books on validation with Interpharm Press. He is a visiting lecturer for the University of Manchester's M.Sc. Pharmaceutical Engineering Advanced Training program, a Chartered Engineer, and a member of the IEE. He is also an active member of the ISPE, where he currently chairs the GAMP Forum Council, which coordinates the various regional GAMP Steering Committees including GAMP Americas, GAMP Europe, and GAMP Japan.

Contributor Biographies

JOHN ANDREWS

Independent Consultant

At the time this book was written, John Andrews was Manager, IT Consulting Service, at KMI, a division of PAREXEL International LLC. His responsibilities included providing consultancy on computer systems validation, compliance, and quality assurance activities within the pharmaceutical, biopharmaceutical, medical device, and other regulated healthcare industries. Mr. Andrews was previously a site Computer System Validation Manager with GlaxoSmithKline, where his responsibilities included all aspects of computer systems validation, from Process Control through Business and Laboratory System Validation. Mr. Andrews also worked for 15 years for SmithKline Beecham Pharmaceuticals, where he held various engineering positions. He is a member of the GAMP 4 Special Interest Group on Process Control and he has sat on the editorial board for GAMP 4.

Contact Information

E-mail: johnandrews2@ntlworld.com

PETER BOSSHARD

Global Quality Assurance Manager, F. Hoffmann-La Roche

Peter Bosshard joined F. Hoffmann-La Roche in 1994 as a pharmacist. He is currently responsible for global-scope quality assurance, which includes audits, GMP compliance assessments of global applications, GMP-Training of IT professionals, and definition of electronic records and signatures strategy. Dr. Bosshard participates in the GAMP D-A-CH Forum (Germany, Austria, Switzerland) and heads its Special Interest Group for Electronic Records and Signatures.

Contact Information

F. Hoffmann-La Roche Ltd.
Global Quality
Building 74/2 OgW 223
Basel CH4070, Switzerland
Tel: +41-61-688-4608
Fax: +41-61-688-8892
E-mail: peter.bosshard@roche.com

ULRICH CASPAR

Project Manager MES, F. Hoffmann-La Roche

Ulrich Caspar joined F. Hoffmann-La Roche in 1984 as a pharmacist. He is currently responsible for an electronic batch recording system used in pharmaceutical production in Basel.

Contact Information

F. Hoffmann-La Roche Ltf
Global Quality
Building 27/424
Basel CH4070, Switzerland
Tel: +41-61-688-6681
Fax: +41-61-688-5103
E-Mail: ulrich.caspar@roche.com

MARK CHERRY

Systems Quality Group Manager U.K. Manufacturing, AstraZeneca

Mark Cherry is responsible for validation of process control, computerized laboratory, and IT systems. Mr. Cherry received his degree in instrumentation and process control engineering in 1987, and worked as a project engineer for Sterling Organics until joining Glaxo in 1990. He was involved in a variety of process control projects within Glaxo until 1995 when he became Engineering Team Manager, responsible for all engineering maintenance activities on a bulk API plant. In 1997 he was appointed Systems and Commissioning Manager for a major capital project within GlaxoWellcome, involving the installation of a large DCS system using the S88.01 approach to batch control. From 1999 to 2001, Mr. Cherry was responsible for computer systems validation for bulk API manufacturing sites within GlaxoWellcome. He is a Chartered Engineer and a member of the Institute of Measurement and Control, the ISPE, and the GAMP European Steering Committee.

Contact Information

AstraZeneca
U.K. Manufacturing
Silk Road Business Park
Macclesfield
Cheshire SK10 2NA, U.K.
Tel: +44 (0) 1625 230882
Fax: +44 (0) 1625 512137
E-mail: mark.cherry@astrazeneca.com

CHRIS CLARK

Head of Quality Assurance, NAPP Pharmaceuticals

Chris Clark graduated from Lancaster University with a B.Sc. degree in biochemistry. He has more than 24 years of QA experience in the pharmaceutical/healthcare industries, beginning with Sterling-Winthrop, then Baxter Healthcare Limited, and finally joining NAPP in 1993. As Head of Quality Assurance, he is responsible for the company's Quality Management System, a role covering all major functions of the company, ensuring compliance to current international regulatory requirements for GMP, GLP, and GCP. Mr. Clark has been involved in a variety of computer-related projects, including the local implementation of ORACLE® Applications 11i, an international Enterprise Document Management system, an international implementation of the ORACLE® Clinical Data Management System, and membership of an international 21 CFR Part 11 Task Force. A member of the GAMP European Steering Committee and Council, Mr. Clark speaks regularly at conferences on the qualification and validation of computerized systems.

Contact Information

NAPP Pharmaceuticals Limited
Cambridge Science Park
Cambridge CB4 0GW, U.K.
Tel: 01223 424444
Fax: 01223 424441
E-Mail: chris.clark@napp.co.uk

PETER COADY

Principal Consultant, P. J. Coady & Associates

Peter Coady is a consultant to the pharmaceutical industry (including GlaxoSmithKline, its merger constituent companies, and Pfizer) specializing in IT and automated systems validation, electronic records and signatures assessments and remediation, and supplier auditing to GAMP, ISO9001,

and TickIT. He has more than 20 years of industrial experience. His career has been centered on project management and computer systems validation, and he was employed as Manager, Electrical, Instrumentation and Systems Group at AMEC (formerly Matthew Hall Engineering) prior to becoming a consultant. Mr. Coady is actively involved in the quality arena and is an independent Lead Quality Management System (QMS) Assessor, Lead TickIT Assessor, and Team Leader for Lloyd's Register Quality Assurance Limited. He has a B.Sc. honors degree and is a Chartered Engineer and a European Engineer. He is a Fellow of the InstMC, a Fellow of the IMechE, a member of the ISPE, and an Associate of the IQA, and he is registered as a Lead TickIT Auditor by the IRCA. He is a GAMP Forum and GAMP Europe Supplier Group Steering Committee member, and represents GAMP on the BSI DISC TickIT Technical Committee BRD/3.

Contact Information

P. J. Coady & Associates
15 Cutter Avenue
Warsash
Southampton, Hampshire SO31 9BA, U.K.
Tel/Fax: +44-1489-572047
Mobile: +44-7710-133-118
E-mail: peter@coadyassociates.com

TONY DE CLAIRE

Principal Consultant, Mi Services Group

Tony de Claire is a Principal Consultant with Mi Services Group, an organization that provides worldwide compliance and validation computer consultancy across the spectrum of system applications in pharmaceuticals, biologicals, and medical devices. As a "user" he led the manufacturing automation and information systems group for SmithKline Beecham's corporate engineering, before moving into consultancy with KMI-Parexel and APDC Consulting. Mr. de Claire is a member of the Institute of Measurement and Control (InstMC), a Senior Member of the International Society for Measurement and Control (ISA), a member of the International Society of Pharmaceutical Engineers (ISPE), and a trained TickIT Auditor.

Contact Information

Mi Services Group
Turnhams Green Business Park
Pincents Lane
Calcot
Reading, Berkshire RG31 4UH, U.K.
Tel: +44-1903-533633
Mobile: +44-7786-250-014
E-Mail: tony.de.claire@mi-services.com

ROGER DEAN

System Support Manager, Pfizer

Roger Dean is a Fellow of Royal Institute of Chemistry, an Associate of Institute of Managers, and a Graduate of Royal Institute of Chemistry by examination. Mr. Dean has spent 14 years in Quality Control/Analytical Chemistry and 2 years in Production Chemistry at Beecham Pharmaceuticals, and 11 years in Quality Operations/Analytical Chemistry and 9 years in IT support and projects at Pfizer Limited (Pharmaceuticals). Mr. Dean has also been involved in project managing the implementation of an EDMS system with significant involvement in its design and validation and also in several other validation projects.

Contact Information

Pfizer Limited
Ramsgate Road
Sandwich, Kent CT13 9NJ, U.K.
Tel: +44 1304 646770
Fax: +44 1304 655585
E-mail: roger.dean@pfizer.com

CHRISTOPHER EVANS

Site Auditing and Compliance Support Manager, GlaxoSmithKline

Christopher Evans joined GlaxoSmithKline in July 1999 following 27 years of service with ICI plc. He is currently responsible for managing Computer Compliance audits for GlaxoSmithKline sites and Contract Manufacturing sites around the world. Mr. Evans has broad international experience in the establishing and managing teams for, and providing technical expertise to, validation projects in primary and secondary manufacturing facilities. He has worked for a number of major pharmaceutical manufacturers in the U.K. and Europe including Pfizer, Astra-Zeneca, Roche, and Napp Laboratories. Mr. Evans was the lead for the two Special Interest Groups covering Software/Hardware Categories and Supplier Auditing for GAMP 4. He is also currently a member of the GAMP Process Control Special Interest Group.

Contact Information

GlaxoSmithKline
Harmire Road
Barnard Castle
County Durham DL12 8XD, U.K.
Tel: +44 (0) 1833 692955
Fax: +44 (0) 1833 692935
E-mail: ce58727@GSK.com

JOAN EVANS

Principal Consultant, ABB

Joan Evans qualified as a chemical engineer at University College Dublin (Ireland) and has extensive experience in the manufacturing industry in project management, quality management, line management, and consultancy positions. In 1995 she transferred to the Life Sciences group of Eutech (now ABB), the U.K.'s cutting edge provider of specialist computer validation services. Ms. Evans is responsible for the management and technical leadership of a range of assignments for blue chip companies, specializing in tailored compliance services for computer systems across the research, manufacturing, and distribution spectrum. She is also internal product champion for ABB Eutech's Electronic Records/Electronic Signatures consultancy offering.

Contact Information

ABB Eutech Process Solutions
Pavilion 9
Belasis Hall Technology Park
P.O. Box 99
Billingham, Cleveland TS23 4YS, U.K.
Tel: +44 (0) 1642 372008
Fax: +44 (0) 1642 372166
E-mail: joan.evans@gb.abb.com

ROBERT FRETZ
Head of Process Automation and MES, F. Hoffmann-La Roche

Robert Fretz joined F. Hoffmann-La Roche more than 30 years ago as a chemical engineer. He is presently responsible for Process Automation in all chemical and galenical manufacturing sites and leads the corporate Manufacturing Execution systems program. Mr. Fretz has broad international experience in all levels of control/automation projects from instrumentation to the enterprise level. Many of these projects included computerized system validation. He co-authored the Hoffmann-La Roche corporate guideline on Process Automation Qualification.

Contact Information

Dept. PTET
Hoffmann-La Roche
Basel CH4070, Switzerland
Phone: +41 61 688 4850
Fax: +41-61-687 07 39
E-mail: robert.fretz@roche.com

STEPHEN C. GILES
Team Leader — Systems Engineering, Pfizer

Stephen C. Giles has worked in the Instrumentation and Control Sector for 20 years. He became involved in process automation following the installation of a highly automated containment plant at Pfizer, Sandwich, U.K. in 1988. On completion of the project commissioning phase, he moved to the Bulk Manufacturing Department where, over the next 9 years, he held a variety of posts within the manufacturing plants before moving back to the Engineering Department where he now has Discipline responsibility for all Capital Automation Projects within the Manufacturing Division.

Contact Information

Pfizer Limited.
PGM - Sandwich, Project Engineering, IPC 606
Ramsgate Road
Sandwich, Kent CT13 9NJ, U.K.
Tel: 01304-646378
Fax: 01304-656176
E-mail: steve.giles@pfizer.com

LUDWIG HUBER
Product Marketing Manager, Agilent Technologies

Ludwig Huber is an international expert on Laboratory Validation and Compliance, and has been the head of the compliance program at Hewlett-Packard and Agilent Technologies for more than 10 years. The author of numerous publications on chromatography and regulatory issues in laboratories, Dr. Huber prepared and executed HP's seminar program on validation, conducted in more than 20 countries with more than 100,000 attendees. He has been a member of the U.S. PDA task force on 21 CFR Part 11 and the GAMP Special Interest Group for Laboratory Equipment, and he has served on the advisory board for the European Compliance Academy.

Contact Information

Agilent Technologies
Postfach 1280
D-76337 Waldbronn, Germany
Tel: +497802980582
Fax: +497802981948
E-Mail: ludwig_huber@agilent.com

ADRIAN KAVANAGH
ERES Specialist, Independent Consultant

Adrian Kavanagh assists a major pharmacutical company in its computer system remediation activities across its European sites. Prior to assuming this role in November 2000, he was embedded within a number of multinational corporations, both pharmaceutical and other industries. These assignments included specifying IT and automation standards, Y2K preparation, project management, and system design. Mr. Kavanagh previously worked in the automotive industry where he managed IT and automation for large turnkey projects.

Contact Information
Tel: +44 1256 401098
Fax: +44 208 7739092
E-Mail: akavanagh@lineone.net

LOUISE KILLA
Pharmaceutical Consultant, LogicaCMG

Louise Killa is a Senior IT Consultant within the Pharmaceutical Sector at LogicaCMG specializing in the validation of commercial and distribution supply chain applications. Her expertise covers various aspects of GxP software development and delivery of different computer systems from R&D through to Commercial Distribution. She joined LogicaCMG in 1997 and has more than 10 years of experience in software development, quality systems, and project management. She received a Master's Degree in Transportation from the University of Wales College at Cardiff. She is an ISO 9001 Lead Auditor, a member of the International Society of Pharmaceutical Engineers, and an active member of the GAMP Europe Forum.

Contact Information
Industry Distribution & Transport Business Unit
LogicaCMG
Chaucer House
The Office Park
Springfield Drive
Leatherhead, Surrey KT22 7LP, U.K.
Tel: +44 207 6379111
Fax: +44 1372 369757
E-mail: Louise.Killa@LogicaCMG.com

BOB McDOWALL
Principal, McDowall Consulting

Bob McDowall has more than 30 years of experience working as an Analytical Chemist, including 15 years in the pharmaceutical industry, working for two multinational companies. He has more than 20 years of experience working with specifying and implementing computerized systems and 17 years of experience with computerized systems validation. Since 1993 Mr. McDowall has been the Principal of McDowall Consulting, a consultancy specializing in, among other areas, the validation of chromatography data systems. Mr. McDowall is also a trained auditor. His expertise has been recognized with the 1997 LIMS Award. He is also on the Editorial Advisory Boards of *Quality Assurance Journal*, *American Pharmaceutical Review,* and *LC-GC* international journals. He is the author of more than 150 papers and book chapters.

Contact Information

McDowall Consulting
73 Murray Avenue
Bromley, Kent BR1 3DJ, U.K.
Tel./Fax: +44 20-8313-0934
E-Mail: r_d_mcdowall @compuserve.com

BARBARA A. MULLENDORE

Director — Corporate Quality Systems, Watson Pharmaceuticals

Barbara A. Mullendore is responsible for corporate policy-making, computer validation, document management, and other quality systems within Watson Pharmaceuticals. Prior to this, she was Global Quality Manager, Development Information Systems, R&D, for AstraZeneca, where she coordinated Quality Management and Compliance across the international R&D IS organization. Ms. Mullendore has 20 years of experience and increasing responsibility in the pharmaceutical and medical device industry, spanning the areas of Manufacturing, Quality Assurance/Compliance, and Information Services/Information Technology. She holds a B.A. degree in Communications from Cabrini College, and she is pursuing an M.Ed. at Penn State University. Ms. Mullendore is a member of the American Society of Quality (ASQ), the Parenteral Drug Association (PDA), and the International Society for Pharmaceutical Engineering (ISPE). She is also a member of the Software Process Improvement Network (SPIN) associated with ASQ and is co-chair of the GAMP Americas R&D/Clinical/Regulatory Special Interest Group. She is a member of the Editorial Advisory Board of *The Journal of Validation Technology* and has published numerous papers and presented extensively on computer validation, 21 CFR Part 11, and related topics.

Contact Information

Watson Pharmaceuticals, Inc.
311 Bonnie Circle
P.O. Box 1900
Corona, CA 92878-1900, U.S.A.
Tel: 001909493-4016
Fax: 001909493 5819
E-Mail: bmullendore@watsonpharm.com

PETER OWEN

Manufacturing IT and Process Automation Leader, Eli Lilly

Peter Owen has worked in the pharmaceutical industry for 16 years for large multinational pharmaceutical corporations. He has held a number of senior roles focusing on manufacturing IT and process automation, many of which have been leadership roles relating to computer system compliance and remediation activities. Most recently Mr. Owen played a leadership role in the formation and management of a corporate-wide project related to Electronic Signatures and Records compliance. Other assignments have included specifying IT and Automation standards; IT Manager, developing a global strategy for process automation development and life cycle management; Y2K preparation; project management; and system development. He worked previously in the oil and gas industry.

Contact Information

Eli Lilly
Manufacturing Quality and Infomatics
Main Site
Kingsclere Road
Basingstoke, Hants RG21 6XA, U.K.
Tel.: +44 (0)7771 344944
Fax: +44 208 7739092
E-mail: owen_peter@lilly.com

ARTHUR D. PEREZ

Executive Expert, IT Quality Assurance, Novartis Pharmaceuticals

Arthur D. Perez received his doctorate in organic chemistry from the University of Michigan in 1983. He has worked for Novartis (starting at Ciba–Geigy) for 20 years, first as a Process Research chemist, then in support of Chemical Manufacturing (where he was first exposed to validation as it pertains to chemical processes), and finally moving into Computer Validation. After 5 years in the Quality Assurance department, Dr. Perez moved to IT where he continues the never-ending quest for quality in computerized systems. He has held leadership roles in computer validation in both corporate and public forums. He is currently the chairman of GAMP Americas and a member of the international GAMP Council.

Contact Information

Novartis Pharmaceuticals
One Health Plaza
East Hanover, New Jersey 07936, U.S.A.
Tel: 001 862 778-3509
Fax: 001 862 778-3273
E-mail: arthur.perez@pharma.novartis.com

CHRIS REID

Director and Principal Consultant, Integrity Solutions

Chris Reid works with Integrity Solutions Ltd, providers of Quality and Compliance Services to healthcare industries. Mr. Reid currently works with leading global organizations developing and implementing quality and compliance strategies including assessment of corporate IT infrastructure, policy development, people development and implementation of risk-based processes and systems. He has worked with many leading and smaller healthcare organizations during his career. Mr. Reid graduated with a degree in computer science and entered the healthcare industry when he joined ICI in 1987 as a senior software engineer, and later became Manager of Pharmaceutical Manufacturing Controls. Subsequently, he joined a leading validation consultancy as Process and Control Systems Validation Manager, where he played a significant role in establishing a highly reputable business.

Contact Information

Integrity Solutions Ltd
P.O. Box 71
Middlesborough, Cleveland TS7 0XY, U.K.
Tel: 01642 320233
Fax: 01642 320233
Email: creid@integrity-solutions.co.uk

TONY RICHARDS

Engineering Operations Manager, AstraZeneca R&D

Tony Richards joined AstraZeneca, a pharmaceutical R&D facility, in 1994. At that time the Engineering Department was engaged in a major change program driven by the Engineering Quality Project. Major facets of the change program included a commitment to customer service through the introduction of multidisciplinary teams, assessment centers, a teamwork training program, reliability-centered maintenance (RCM), a Maintenance Management System, electronic maintenance documentation, and outsourcing maintenance. Previously, Mr. Richards worked in the manufacturing and nuclear industry.

Contact Information

AstraZeneca
R&D Charnwood
Engineering Dept
Bakewell Road
Loughborough, Leicestershire LE11 5RH, U.K.
Tel: +44 1509644420
Fax: +44 1509645579
E-Mail: tony.richards@charnwood.gb.astra.com

OWEN SALVAGE

Senior Consultant, Lifesciences, ABB

Owen Salvage has more than 15 years of experience working with computer technology applications in the pharmaceutical industry. His engineering experience includes 10 years with ICI and Zeneca and overseas, managing an IT group serving CSR in Australia and New Zealand. Since returning to the U.K. and joining ABB, Mr. Salvage has worked primarily with IT groups supporting the installation of global IT systems projects. A Chartered Engineer with the Institute of Electrical Engineers, Mr. Salvage holds a B.Sc. in electronic engineering from Salford University. He has a Diploma in Management and is currently completing an M.B.A. from the University of Durham.

Contact Information

ABB
Belasis Hall Technology Park
Billingham, Cleveland TS23 4YS, U.K.
Tel: +44 1642-372000
Fax: +44 1642-372166
E-mail: owen.salvage@gb.abb.com

RON SAVAGE

Head — Quality Technology Strategy, GlaxoSmithKline

Ron Savage heads the team responsible for developing and implementing the strategy for technology implementation in the Quality function of the Manufacturing & Supply division of GlaxoSmithKline. In this role, he interfaces between Quality and IT functions to identify opportunities for business improvement through technology delivery to more than 100 manufacturing sites. He recently completed a 2-year appointment as manager of a project to deliver a major LIMS upgrade to sites in the former GlaxoWellcome manufacturing division. Mr. Savage was previously Validation Manager for the primary manufacturing division of GlaxoWellcome. He has worked in the pharmaceutical industry for more than 20 years, holding posts in the Technical, Production, and Quality functions. He is a Chartered Engineer, a member of The Institute of Chemical Engineers, a Chartered Biologist, and a member of The Institute of Biology.

Contact Information

GlaxoSmithKline
North Lonsdale Rd.
Ulverston, Cumbria LA12 9DR, U.K.
Tel: +44 1229482062
Fax: +44 1229482004
E-mail: ron.w.savage@gsk.com

NICOLA SIGNORILE

Site IT Manager, Aventis
Nicola Signorile is IT Manager of the Aventis site in southern Italy, which manufactures secondary pharmaceuticals and is subject to FDA inspections. Mr. Signorile spent 3 years as a consultant dealing with information systems and ERP/MRP II (Manufacturing Resource Planning) before joining Aventis' IT function 10 years ago. Previously, he spent 4 years developing control software on data network communication systems for NATO and as a network systems integrator for a commercial aerospace company.

Contact Information

Gruppo Lepetit S.p.A
03012 Anagani (FR)
Localita Valcanello, Italy
Tel: +39775 760309
Fax: +39775 760 224
E-Mail: nicolarosario.signorile@hmrag.com

ROB STEPHENSON

Regulatory Systems Team Leader, Pfizer
Rob Stephenson is currently responsible for the implementation and operational management of regulatory IT systems within Pfizer's U.K. manufacturing facility in Sandwich, Kent. After obtaining his Ph.D. in physics he joined the Boots Company in 1977 and, since then, he has worked in several capacities within the pharmaceutical and personal product sectors for companies such as Eli Lilly, Unilever, and Coty. Mr. Stephenson became involved with computer validation as a QC officer operating within Pfizer's IT group, where he was also the local (manufacturing) site coordinator for its 21 CFR Part 11 initiative. He is a member of the GAMP Council and GAMP Europe Steering Committee.

Contact Information

Pfizer Ltd (ipc 081)
Ramsgate Road
Sandwich, Kent CT13 9NJ, U.K.
Tel: +44 1304 648059
Fax: +44 1304 655585
E-mail: robert.stephenson@pfizer.com

ANTHONY J. TRILL

Senior Inspector, Medicines and Healthcare products Regulatory Agency
Anthony J. Trill joined the Medicines Inspectorate in 1984 and since 1988 has had a leadership responsibility for MHRA GMP standards and inspection guidance relating to computerized systems. He also carries out routine regional GMP inspection work, both in the U.K. and abroad. Before joining the MHRA, he worked for more than 18 years for three multinational pharmaceutical companies in R&D, new product and process development, production, QA, and technical services

in management and technical roles. During his industrial career he was a member of the ABPI's Technical Committee in the U.K. Mr. Trill has lectured widely and published on a variety of topics, including innovation, validation, automated systems, and general GMP compliance matters. He has been a member of several review panels for quality critical guidance, including ICSE, TickIT, FRESCO, and the GAMP Guide. Mr. Trill is also a member of the GAMP Forum Steering Committee and the Editorial Advisory Board to Pharmaceutical Technology — Europe (Advanstar Publications). He is the PE006 working party leader for PIC/S, which is developing a guideline across the GxP disciplines for international Inspectorates entitled "Good Practices for Computerized Systems in Regulated 'GXP' Environments" (Ref: PI 011-1). Mr. Trill holds a B.Sc. (Honours) in pharmacy from the University of Aston and an M.Sc. in pharmaceutical technology from the University of London. He is an IRCA Lead Auditor and eligible as an EC Qualified Person.

Contact Information

MHRA (Inspection and Enforcement)
North-West Regional Office, Room 209
Chantry House
City Road
Chester CH1 3AQ, U.K.
Tel: +44 1244 351515
Fax: +44 1244 319762
E-mail: tony.trill@mca.gsi.gov.uk

Abbreviations

4GL	Fourth Generation Language
ABAP	Advanced Business Application Program (SAP R/3)
ABB	Asea Brown Boveri
ABO	Blood Groups: A, AB, B, O
ABPI	Association of the British Pharmaceutical Industry
ACDM	Association for Clinical Data Management
ACRPI	Association for Clinical Research in the Pharmaceutical Industry
ACS	Application Configuration Specification
A/D	Analog to Digital
ADE	Application Development Environment
AGV	Automated Guided Vehicle
AIX	Advanced Interactive eXecutive, a version of UNIX produced by IBM
ALARP	As Low As Reasonably Practical
ANSI	American National Standards Institute
API	Active Pharmaceutical Ingredient
APV	Arbeitsgemeinschaft für Pharmazeutische Verfahrenstechnik
AQAP	Association of Quality Assurance Professionals
ASAP	Accelerated SAP R/3 application development methodology
ASCII	American Standard Code for Information Interchange
ASTM	American Society for Testing and Materials
AUI	Application User Interface
BARQA	British Association for Research Quality Assurance
BASEEFA	British Approvals Service for Electrical Equipment in Flammable Atmospheres
BASIC	Beginners All-purpose Symbolic Instruction Code
BCD	Binary Coded Decimal
BCS	British Computer Society
BGA	Bundesgesundheitsamt (German Federal Health Office)
BIOS	Basic Input Output System
BIRA	British Institute of Regulatory Affairs
BMS	Building Management System
BNC	Boyonet Neil Concelman
BOM	Bill of Materials
BPC	Bulk Pharmaceutical Chemicals
BPR	Business Process Re-engineering
BS	British Standard
b/s	bits per second
BSI	British Standards Institution
CA	Certification Agency
CAD	Computer Aided Design
CAE	Computer Aided Engineering
CAM	Computer Aided Manufacturing
CANDA	Computer Assisted NDA (United States)
CAPA	Corrective And Preventative Action
CASE	Computer-Aided Software Engineering

CBER	Center for Biologics Evaluation and Research, FDA
CCTA	Central Computer and Telecommunications Agency
CD	Compact Disk
CDDI	Copper Distributed Data Interface
CDER	Center for Drug Evaluation and Research, FDA
CDMS	Clinical Database Management System
CDRH	Centre for Devices and Radiological Health
CD-ROM	Compact Disk — Read Only Memory
CD(-RW)	Compact Disk — rewritable
CDS	Chromatography Data System
CE	Communauté Européene (EU Medical Device Mark)
CE	Capillary Electrophoresis
CEFIC	Chemical European Federation Industry Council
CENELEC	European Committee for Electrotechnical Standardization
CFR	United States Code of Federal Regulation
CGM	Computer Graphics Metafile
cGMP	Current Good Manufacturing Practice
CHAZOP	Computer Hazard and Operability Study
CIM	Computer Integrated Manufacturing
CIP	Clean In Place
CISPR	International Special Committee on Radio Interference (part of IEC)
CMM	Capability Maturity Model
CO	Costing
COBOL	Common Business Oriented Language
COM	Component Object Model
COQ	Cost of Quality
COTS	Commercial Off-The-Shelf
CPG	Compliance Policy Guide (United States)
CPU	Central Processing Unit
CRC	Cross Redundancy Check
CRM	Certified Reference Material
CROMERR	Cross-Median Electronic Reporting and Record-Keeping
CSA	Canadian Standards Association
CSV	Computer System Validation
CSVC	Computer Systems Validation Committee (of PhRMA)
CTQ	Critical to Quality
CV	Curriculum Vitae
DAC	Digital to Analog Converter
DACH	German-speaking countries of Germany (D), Austria (A), and Switzerland (CH)
DAD	Diode Array Detector
DAM	Data Acquisition Method
DAT	Digital Audio Tape
DBA	Database Administrator
DBMS	Database Management System
D-COM	Distributed Component Object Model
DCS	Distributed Control System
DDMAC	Division of Drug Marketing, Advertising and Communications
DECnet	Digital Equipment Corporation Network
DIA	Drug Information Association
DLL	Dynamic Link Library
DLT	Digital Linear Tape

DoH	U.K. Department of Health
DOS	Disk Operating System
DPMO	Defects Per Million Opportunities
DQ	Design Qualification
DR	Design Review
DRP	Distribution Requirement Planning
DSL	Digital Subscriber Line
DSP	Digital Signal Processing
DVD	Digital Video Disk
DXF	Data Exchange File
EAM	Engineering Asset Management
EAN	European Article Number
EBRS	Electronic Batch Record System
EC	European Community
EDI	Electronic Data Interchange
EDMS	Electronic Document Management System
EEC	European Economic Community
EEPROM	Electronically Erasable Programmable Read Only Memory
EFPIA	European Federation of Pharmaceutical Industry Association
EFTA	European Free Trade Association
EIA	Electronics Industries Association
EISA	Extended Industry Standard Architecture
ELA	Establishment License Application
ELD	Engineering Line Diagram
EMC	Electro-Magnetic Compatibility
EMEA	European Medicines Evaluation Agency
EMI	Electro-Magnetic Interference
EMS	Engineering Management System
ENCRESS	European Network of Clubs for Reliability and Safety of Software
EOLC	Environmental/Operation Life Cycle
EPA	U.S. Environmental Protection Agency
EPROM	Electronic Programmable Read Only Memory
ERD	Entity Relationship Diagram
ERES	Electronic Records, Electronic Signatures
ERP	Enterprise Resource Planning
ESD	Electro-Static Discharge
ESD	Emergency Shutdown
EU	European Union
FAT	Factory Acceptance Testing
FATS	Factory Acceptance Test Specification
FAX	Facsimile Transmission
FDA	U.S. Food and Drug Administration
FD&C	U.S. Food, Drug, and Cosmetics Act
FDDI	Fiber Distributed Data Interface
FDS	Functional Design Specification
FEFO	First Expired First Out
FFT	Fast Fourier Transform
FI	Finance
FIFO	First In–First Out
FM	Factory Mutual Research Corporation
FMEA	Failure Mode Effect Analysis

FORTRAN	Formula Translator
FS	Functional Specification
FTE	Full-Time Employee
FT-IR	Fourier Transform — Infrared
FTP/IP	File Transfer Protocol/Internet Protocol
GALP	Good Automated Laboratory Practice
GAMP	Good Automated Manufacturing Practice
GB	Giga-Byte
GC	Gas Chromatography
GCP	Good Clinical Practice
GDP	Good Distribution Practice
GEP	Good Engineering Practice
GERM	Good Electronic Record Management
GIGO	Garbage In, Garbage Out
GLP	Good Laboratory Practice
GMA	Gesellschaft Meβ- und Automatisierungstechnik
GMP	Good Manufacturing Practice
GPIB	General Purpose Interface Bus
GPP	Good Programming Practice
GUI	Graphical User Interface
GxP	GCP/GDP/GLP/GMP
HACCP	Hazard Analysis and Critical Control Point
HATS	Hardware Acceptance Test Specification
HAZOP	Hazard and Operability Study
HDS	Hardware Design Specification
HIV	Human Immunodeficiency Virus
HMI	Human Machine Interface
HP	Hewlett-Packard
HPB	Canadian Health Products Branch Inspectorate
HPLC	High Performance Liquid Chromatography
HPUX	Hewlett-Packard UNIX
HSE	U.K. Health and Safety Executive
HTML	Hyper Text Markup Language
HVAC	Heating, Ventilation, and Air Conditioning
IAPP	Information Asset Protection Policies
IBM	International Business Machines
ICH	International Conference on Harmonization
IChemE	U.K. Institution of Chemical Engineers
ICI	Imperial Chemical Industries
ICS	Integrated Control System
ICSE	U.K. Interdepartmental Committee on Software Engineering
ICT	Information and Communications Technologies
ID	Identification
IEC	International Electrotechnical Commission
IEE	U.K. Institution for Electrical Engineers
IEEE	Institute of Electrical and Electronic Engineers
IETF	Internet Engineering Task Force
IIP	Investors in People
IKS	Swiss Agency for Therapeutic Products (also known as SwissMedic)
IMechE	U.K. Institution for Mechanical Engineers
INS	Instrument File Format

InstMC	U.K. Institution for Measurement and Control
InterNIC	Internet Network Information Center
I/O	Input/Output
IP	Index of Protection
IP	Ingress Protection
IP	Internet Protocol
IPC	Industrial Personal Computer
IPng	IP Next Generation
IPR	Intellectual Property Rights
IPSE	Integrated Project Support Environment
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
IPX	Internet Packet eXchange
IQ	Installation Qualification
IQA	U.K. Institute of Quality Assurance
IRCA	International Register of Certificated Auditors
IS	Intrinsically Safe
ISA	Industry Standard Architecture bus (also known as AT bus)
ISA	Instrument Society of America
ISM	Industrial, Scientific, and Medical
ISO	International Standards Organization
ISP	Internet Service Provider
ISPE	International Society for Pharmaceutical Engineering
IT	Information Technology
ITIL	Information Technology Infrastructure Library
ITT	Invitation to Tender
IVRS	Interactive Voice Recognition System
IVT	Institute of Validation Technology
JAD	Joint Application Development
JETT	North American Joint Equipment Transition Team
JIT	Just In Time
JPEG	Joint Photographic Experts Group
JPMA	Japanese Pharmaceutical Managers Association
JSD	Jackson Development Method
KOSEISHO	Ministry of Health and Welfare (Japan)
KPI	Key Performance Indicator
KT	Kepner Tregoe
LAN	Local Area Network
LAT	Local Area Transport, a DEC proprietary Ethernet protocol
LC	Liquid Chromatography
LIMS	Laboratory Information Management System
L/R	Inductance/Resistance Ration
MAU	Media Attachment Unit
MASCOT	Modular Approach to Software Construction, Operation, and Test
MB	Mega-Byte
Mb/s	Mega bits per second
MC	Main cross connect room
MCA	Micro Channel Architecture
MCA	U.K. Medicines Control Agency
MCC	Motor Control Center
MD	Message Digital, an algorithm to verify data integrity

MDA	U.K. Medical Device Agency
MDAC	Microsoft Data Access Components
MES	Manufacturing Execution System
MHLW	Japanese Ministry for Health, Labor, and Welfare
MHRA	U.K. Medicines and Healthcare products Regulatory Authority
MHW	Japanese Ministry for Health and Welfare
MIME	Multipurpose Internet Mail Extension
MIS	Management Information System
MM	Materials Management
MMI	Man Machine Interface (see HMI)
MMS	Maintenance Management System
MODEM	Modulator-Demodulator Units
MPA	Swedish Medical Products Agency
MPI	Manufacturing Performance Improvement
MPS	Master Production Schedule
MRA	Mutual Recognition Agreement
MRP	Materials Requirements Planning
MRP II	Manufacturing Resource Planning
MRM	Multiple Reaction Monitoring
MSAU/MAU	IBM's Multi-Station Access Unit (Token Ring hubs)
MTTF	Mean Time To Failure
NAMAS	U.K. National Measurement Accreditation Service
NAMUR	Normenarbeitsgemeinschaft für Meβ- und Regelungstechnik
NATO	North Atlantic Treaty Organization
NDA	U.S. New Drug Application
NetBEUI	NetBIOS Extended User Interface
NetBIOS	Network Basic Input/Output System
NIC	Network Interface Card
NIST	National Institute of Standards and Technology
NIR	Near Infra-Red
NMR	Nuclear Magnetic Resonance
NOS	Network Operating System
NPL	National Physics Laboratory
NSA	U.S. National Security Agency
NT	New Technology
NTL	National Testing Laboratory
OCR	Optical Character Recognition
OCS	Open Control System
OECD	Organisation for Economic Co-operation and Development
OEM	Original Equipment Manufacturer
OICM	Swiss Office Intercantonal de Controle des Medicaments
OLE	Object Linking and Embedding
O&M	Operation and Maintenance
OMM	Object Management Mechanism
OOS	Out Of Specification
OQ	Operational Qualification
OS	Operating System
OSI	Open System Interconnect
OTC	Over The Counter
OTS	Off The Shelf
OWASP	Open Web Application Security Project

PAI	Pre-Approval Inspection
PAR	Proven Acceptable Range
PAT	Process Analytical Technology
PC	Personal Computer
PCI	Peripheral Component Interconnect
PCX	Graphics File Format
PDA	Parenteral Drug Association
PDA	Personal Digital Assistant
PDF	Portable Document Format
PDI	Pre-Delivery Inspection
PhRMA	Pharmaceutical Research and Manufacturing Association
PIC	Pharmaceutical Inspection Convention
PIC/S	Pharmaceutical Inspection Co-operation Scheme
PICSVF	U.K. Pharmaceutical Industry Computer System Validation Forum
PID	Proportional, Integral, Derivative (Loop)
P&ID	Process Instrumentation Diagram
PIR	Purchase Item Receipt
PKI	Public Key Infrastructure
PLC	Programmable Logic Controller
PMA	Pharmaceutical Manufacturers Association
POD	Proof of Delivery
PP-PI	Production Planning — Process Industries
PQ	Performance Qualification
PQG	Pharmaceutical Quality Group (part of IQA)
PRINCE2	Projects In Controlled Environments 2
PRM	Process Route Maps
PSI	Statisticians in the Pharmaceutical Industry
PSU	Power Supply Unit
PTB	Physikalische-Technische Bundesanstalt
PTT	Public Telephone and Telecommunications
PV	Performance Verification
QA	Quality Assurance
QC	Quality Control
QM	Quality Management
QMS	Quality Management System
QP	European Union Qualified Person
QS	Quality System
QSIT	FDA Quality System Inspection Technique
QTS	Quality Tracking System
RAD	Rapid Application Development
RAD	Role Activity Diagram
RAID	Redundant Array of Inexpensive Disks
RAM	Random Access Memory
RCCP	Rough Cut Capacity Planning
RCM	Reliability Centered Maintenance
R&D	Research and Development
RDB	Relational Database
RDT	Radio Data Terminal
RF	Radio Frequency
RFI	Radio Frequency Interference
RFID	Radio Frequency Identification

RFP	Request for Proposal
RH	Relative Humidity
ROM	Read Only Memory
RP	German Federal Ministry for Health
RPharmS	U.K. Royal Pharmacy Society
RPN	Risk Priority Number
RSA	Rivest, Shamir, Adleman Public-Key Cryptosystem
RSC	U.K. Royal Society of Chemists
RTD	Radio Data Terminal
RTF	Rich Text Format
RTL/2	Real-Time Language, Version 2
RTM	Requirements Traceability Matrix
RTSASD	Real-Time System-Analysis System-Design
SAA	Standards Association of Australia
SAM	Software Assessment Method
SAP	Systems, Applications, Products in Data Processing (Company)
SAP R/3	An ERP system developed by SAP
SaRS	U.K. Safety and Reliability Society
SAS	Statistical Analysis System
SAT	Site Acceptance Testing
SATS	System Acceptance Test Specification
SCADA	Supervisory Control and Data Acquisition
SCR	Source Code Review
SD	Sales and Distribution
SDLC	Software Development Life Cycle
SDS	Software Design Specification
SEI	Carnegie Mellon University’s Software Engineering Institute
SFC	Sequential Function Chart
SGML	Standard Generalized MarkUp Language
SHA	Secure Hash Algorithm
SHE	Safety, Health & Environment
SIP	Sterilization In Place
SKU	Stock Keeping Unit
SLA	Service Level Agreement
SLC	System Life Cycle
SM	Section Manager
SMART	Specific, Measurable, Achievable, Recorded, Traceable
SMDS	Software Module Design Specification
S/MIME	Simple Multipurpose Internet Mail Extension
SMS	Microsoft’s System Management Server
SMTP	Simple Mail Transfer Protocol
SNA	Systems Network Architecture
SOP	Standard Operating System
S&OP	Sales and Operations Planning
SOUP	Software Of Unknown Pedigree
SPC	Statistical Process Control
SPICE	Software Process Improvement Capability d’Etermination
SPIN	Software Process Improvement Network
SPSS	Statistical Product and Service Solutions
SQA	Society of Quality Assurance
SQAP	Software Quality and Productivity Analysis

SQL	Software Query Language
STARTS	Software Tools for Large Real-Time Systems
STD	Software Technology Diagnosis
STEP	STandard for Exchange of Product model data in ISO 10303
STP	Shielded Twisted Pair
StRD	Statistical Reference Dataset
SWEBOK	Software Engineering Body of Knowledge
TC	Terminal Cross connect room
T&C	Threats and Controls
TCP	Transmission Control Protocol
TCP/IP	Internet Protocol/Transmission Control Protocol
TCU	Temperature Control Unit
TIA	Telecommunications Industry Association
TIFF	Tagged Image File Format
TIR	Test Incident Report
TGA	Australian Therapeutic Goods Administration
TÜV	Technischer Überwachungs-Verein
UAT	User Acceptance Testing
UCITA	U.S. Uniform Computer Information Transactions Act
U.K.	United Kingdom
UL	Underwriters Laboratories Inc.
ULD	Utility Line Diagrams
UPC	Universal Product Code
UPS	Uninterruptible Power Supply
URL	Universal Resource Locator
URS	User Requirement Specification
U.S.	United States (of America)
U.S.A.	United States of America
USD	United States Dollars
UTP	Unshielded Twisted Pair
UV	Ultra Violet
VBA	Visual Basic
VDS	Validation Determination Statement
VDU	Visual Display Unit
VMP	Validation Master Plan
VMS	Virtual Memory System
VP	Validation Plan
VPN	Virtual Private Network
VR	Validation Report
VSR	Validation Summary Report
V-MAN	Validation Management
WAN	Wide Area Network
WAO	Work Station Area Outlet
WAP	Wireless Application Protocol
WFI	Water For Injection
WHA	World Health Agreement
WHO	World Health Organisation
WIFF	Waveform Interchange File Format
WIP	Work In Progress
WMF	Windows Metafile Format
WML	Wireless Markup Language

WORM	Write Once, Read Many
WWW	World Wide Web
WYSIWYG	What You See Is What You Get
XML	Extensible Markup Language
Y2K	Year 2000

Contents

1 Why Validate?

CONTENTS

Computer systems support billions of dollars of pharmaceutical and healthcare sales revenues. Over the past 30 years, the pharmaceutical and healthcare industries have increasingly used computers to support the development and manufacturing of their products. Within research environments, computer systems are used to speed up product development, reducing the time between the registration of a patent and product approval and, hence, optimizing the time available to manufacture a product under a patent. Computer systems are also used within the production environment to improve manufacturing performance, reduce production costs, and improve product quality. It is important that these systems are validated as fit for purpose from a business and regulatory perspective. Regulatory authorities treat lack of validation as a serious deviation. Pharmaceutical and healthcare companies need a balanced, proactive, and coordinated strategy that addresses short, medium, long-term, internal, and external needs and priorities.

STRATEGIC ADVANTAGE

Many computer systems have been implemented on the promise of giving pharmaceutical and healthcare companies a competitive advantage. Claimed benefits in the business case usually include:

- Built-in quality controls to ensure that the process is followed correctly, reducing human error and the need to inspect for quality in drug and healthcare products. This reduces rejections, reworks, and recalls, and supports the introduction of further efficiencies (e.g., Six Sigma).
- Standardization of production practices to build consistent ways of working, thereby facilitating the movement of products from development to production and between production sites. This is increasingly important for large multisite manufacturing organizations that are rationalizing their operations.
- Reducing the cost of sales by removing non-value-added activities (e.g., quality inspections, exception handling, rework, and scrap).
- Increasing the velocity of product through the supply chain by reducing process errors and wait times, and by improving scheduling.
- Elimination of duplicate effort by working on establishing electronic master records and thus avoiding the need for the presentation of information in various paper formats, each of which must be controlled.

Unfortunately, the claimed return on investment has rarely fulfilled expectations; nevertheless, significant benefits have been realized.

TODAY'S COMPUTING ENVIRONMENT

The mapping of systems within any one organization will vary. The range of applications found in research and development, pharmaceutical manufacturing organizations, consumer healthcare manufacturing, and distribution organizations is illustrated in Figure 1.1. These applications are increasing based on Commercial Off-The-Shelf (COTS) products and can broadly be divided into the following generic types:

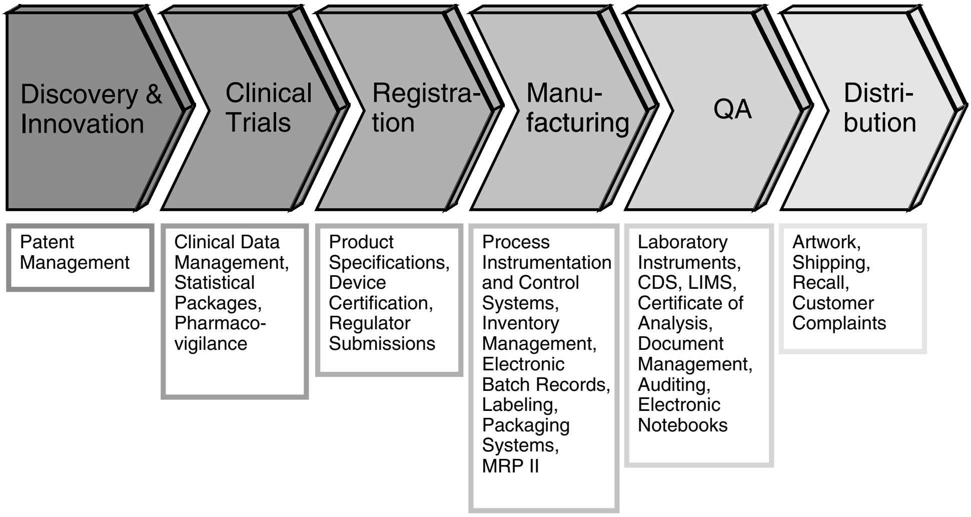

Discovery & Innovation
Clinical Trials
Registra-tion
Manu-facturing
QA
Distri-bution
Patent Management
Clinical Data Management, Statistical Packages, Pharmaco-vigilance
Product Specifications, Device Certification, Regulator Submissions
Process Instrumentation and Control Systems, Inventory Management, Electronic Batch Records, Labeling, Packaging Systems, MRP II
Laboratory Instruments, CDS, LIMS, Certificate of Analysis, Document Management, Auditing, Electronic Notebooks
Artwork, Shipping, Recall, Customer Complaints

FIGURE 1.1 Computer System Applications.

- Laboratory application (e.g., analytical, measurement)
- Control system (e.g., PLC, SCADA, DCS)
- Desktop application (e.g., spreadsheets, databases, and Web applications)
- IT system (e.g., ERP, MRP II, LIMS, EDMS)
- Computer network infrastructure (e.g., servers, networks, clients)

Computer systems such as these can account for significant capital costs. Such assets deserve the closest attention and the most careful management. Efficient validation within an enterprise strategy is the key to achieving cost-effective and compliant implementations. How to do this and, indeed, the provision of practical advice and guidance on validating computer systems in general (based on extensive industry experience) are the main aims of this book.

Rudimentary Computer System Characteristics

Computer systems share some basic hardware and software characteristics that must be understood in order to appreciate the quality and compliance issues discussed in this book.

First, it is important to grasp that the proportion of hardware costs is, on the whole, reducing as a percentage of the lifetime cost of computer systems, as illustrated in Figure 1.2. Computer systems are now less reliant on bespoke hardware than was the case until quite recently, and now consist largely of an assembly of standard components that are then configured to meet their business objective. Standard software products are more readily available than ever before, although these products are often customized with bespoke interfaces to enable them to link into other computer systems. Software products are also becoming larger and more sophisticated. With the use of ever larger and more complex software applications the task of maintenance has also increased, especially as many vendors of commercial software have acquired the habit of releasing their products to market while significant numbers of known errors still remain. The effective subsequent management of defect-correction patch installations and other code changes can be challenging.

While software shares many of the same engineering tasks as hardware, it is nevertheless different.[1] The quality of hardware is highly dependent on design, development, and manufacture. The quality of software is also highly dependent on design and development, but its manufacture consists of replication, a process whose validity can easily be verified. For software, the hardest part is not replicating identical copies but rather the design and development of software being

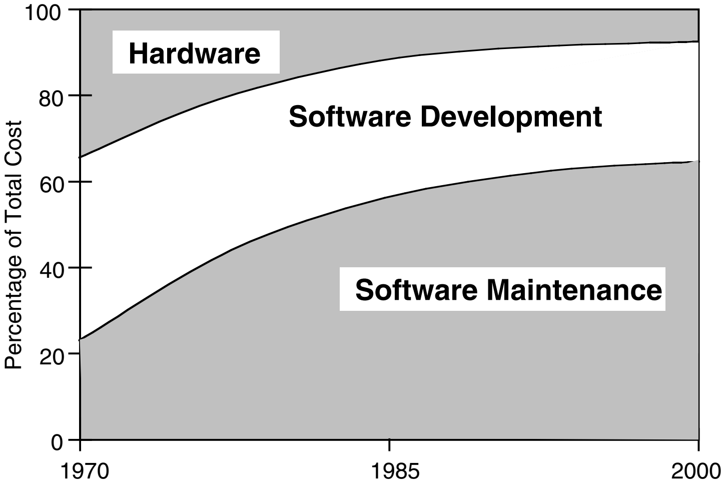


FIGURE 1.2 Changing Proportions of Software and Hardware Costs.

copied to predetermined specifications. Then, again, software does not wear out like hardware. On the contrary, it often improves over time as defects are discovered and corrected.

One of the most significant features of software is branching — its ability to execute alternative series of instructions based on different logic states and/or inputs. This feature contributes heavily to another characteristic of software: its complexity. Even short programs can be very complex. Comprehensive testing is seldom practical, and latent defects may remain hidden within unexercised and untested software pathways. Quality management practices are therefore essential to ensure with sufficient confidence that software is fit for purpose.

Another related characteristic of software is the speed and ease with which it can be changed. This characteristic can lead both software and nonsoftware professionals to the false impression that software problems can be easily corrected. This is true at one level, but there are complications. Repairs made to correct software defects actually establish a new design. Because of this, seemingly insignificant changes in the software code can create unexpected and very significant defects to arise mysteriously elsewhere in the software.

PROBLEMS IN IMPLEMENTING COMPUTER SYSTEMS

The Standish Group surveys have consistently reported in recent years that less than one third of computer system projects are on time, without overspending, with all planned functionality present. Perhaps worse is the assertion that over one third of applications are never even delivered at all (see Figure 1.3). Even if a project appears superficially to have been successful, that does not imply that the application it delivered will be used long enough to repay the investment. Many business cases for new software-related products require a return on investment within 3 years, but in practice a high proportion of systems have a shorter life than this. Computer technology and business IT strategies tend to be very dynamic. In such a changing environment, applications tend to be quickly labeled as inflexible and/or redundant, requiring replacement by new and more sophisticated systems long before they have paid for themselves.

Quality management systems must be mature and robust to mitigate the risk of unsuccessful projects. Factors that critically determine the likelihood of success of computer projects are summarized in Table 1.1. Lack of user input can almost be guaranteed to result in an incomplete user requirement specification, a foundation of sand upon which only the shakiest edifice can be built. Those with only general skills should not be deployed on critical tasks such as quality assurance, testing, and project management. Specific technical expertise and proven competence are required for handling new technology. Good team communication is also vital. Ineffective supplier management and poor relationships with subcontractors can aggravate an already weak technical

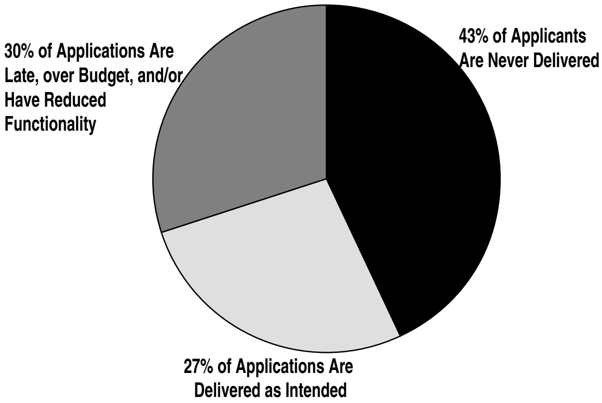


FIGURE 1.3 Project Outcomes.

TABLE 1.1
Factors That Affect Project Success

Successful Project	Unsuccessful Project
User Involvement	Lack of User Input
Executive Management Support	Poor Project Management
Clear Statement of Requirements	Changing Requirements
Proper Planning	Lack of Executive Support
Realistic Expectations	Technological Incompetence
Smaller Project Milestones	Lack of Resources
Competent Staff	Unrealistic Schedule Pressure

base. Software upgrades are often conceived of to rectify hardware deficiencies rather than to seek a more appropriate hardware solution. Teams often mistakenly focus on innovation rather than on cost and risk containment.

Gaining acceptance of quality management is vital.[2] Both the heart and mind of senior management must be behind the use and benefits of quality management systems. There is more than enough evidence to make the case that quality management systems work. Without clear leadership at an executive level, however, it will be almost impossible to overcome statements like "We don't have time for paperwork," "Surely good practice is good enough," "We can't afford the luxury of procedures," "The documentation being asked for is not practical," "Too formalized an approach will undermine flexibility, slow projects down, and increase costs," and "The concept is good and we hope to use it some day, but not just yet."

Simply monitoring quality performance is just not adequate. The effectiveness of quality management systems should be actively managed and performance improvement opportunities seized. Business benefits should more than compensate for any investment in quality. Senior management, system owners, project managers, and anyone else involved with computer projects need to appreciate this. This book will help explain what needs to be done to successfully achieve quality and compliance of computer systems in the pharmaceutical and healthcare industries.

GOOD PRACTICE

Quality Assurance

The achievement of quality in a product should be based on the adoption of good practices. Neither these (whether in relation to computer systems or not) nor the concept of quality, were invented by the pharmaceutical and healthcare industry. Good computer practices existed long before pharmaceutical and healthcare industry regulations required their application. The basic underlying premise is that quality cannot be tested into computer systems once developed. On the contrary, it must be built in right from start. Defects are much cheaper to correct during the early stages of system development than to have them left to be weeded out just before release or, worse, by disaffected customers. The additional cost generated by ensuring that the system is sound at every stage in its development, from conception to testing, is far less than the cost and effort of fixing the computer system afterward, not forgetting the hidden losses suffered through customer disaffection. So do not wait until the end of the project to put things right!

Sir John Harvey-Jones, chairman of the former industrial chemicals giant ICI, summed this up pithily: "The nice thing about not planning is that failure then comes as a complete surprise." This said, it is important to appreciate that planning does not come naturally to many people and the temptation to jump in to code development before the groundwork of properly defining requirements has been completed often proves irresistible. This tendency can be exacerbated by managers

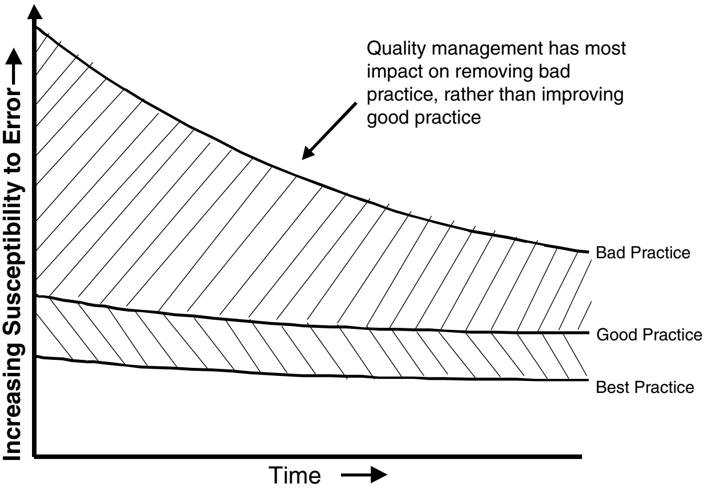


FIGURE 1.4 Benefits of Software Quality Management.[3]

expecting too much too soon. A degree of self-discipline is required because short-cutting the quality process will almost certainly wreak havoc later on.

Quality Management System

As illustrated in Figure 1.4, adopting good practices will progressively engineer out the bad practices and deliver measurable cost benefits. Projects conducted without an underpinning quality management system have a variable and unpredictable success rate. As such, quality management needs to address the well-worn issues of:

- Requirements misunderstanding
- Scope creep
- Development risk
- Quality of software of unknown pedigree
- Uncontrolled change
- Design errors
- Too much or too little documentation
- Project progress reporting and action planning
- Resource inadequacy
- Regulatory inspection readiness
- Ease of system operation and maintenance
- Planning ahead for retirement and decommissioning

GxP Philosophy

Pharmaceutical and healthcare regulations require the adoption of quality practices. Good Practices are associated with clinical trials of drug products (Good Clinical Practices — GCP), the manufacture of licensed drug products (Good Manufacturing Practices — GMP), distribution and onward warehousing of drug products (Good Distribution Practices — GDP), and associated laboratory operations (Good Laboratory Practices — GLP). They are applied to a healthcare industry that includes biotechnology and cosmetic products, medical devices, diagnostic systems, Bulk Pharmaceutical Chemicals (BPCs), and finished pharmaceuticals for both human and veterinary use. Collectively these good practices are known as the GxP. The philosophy behind GxP is to ensure that drug products

> are consistently produced and controlled to the quality standards [safety, quality and efficacy] appropriate to their use.[4]

The pharmaceutical industry is subject to GxP regulations such as the World Health Organisation's (WHO) resolution WHA 22.50; the European Union's (EU) GMP Directive 91/356/EEC; the Japanese Manual on Computer Systems in Drug Manufacturing; the U.S. Code of Federal Regulations Title 21, Parts 210 and 211; and Medicinal Products — Part 1 of the Australian Code of Good Manufacturing for Therapeutic Goods. GMP is enforced on the ground by the national regulatory authorities. Well-known GMP regulatory authorities in the pharmaceutical industry include the U.S. Food and Drug Administration (FDA), the U.K. Medicines and Healthcare products Regulatory Authority (MHRA), and the Australian Therapeutic Goods Administration (TGA). The regulatory authorities can prevent the sale of any product in their respective country if they consider its manufacture non-GxP compliant. To pharmaceutical and healthcare companies, GxP is nothing less than a license-to-operate matter.

Duty of Care

Like other financial governing bodies around the world, the London Stock Exchange requires pharmaceutical and healthcare companies to comply with laws and regulations including those dealing with GxP and consumer protection.[5] Collectively these are often portrayed as the exercise of a "duty of care" through operating in a responsible and reasonable manner. This duty of care embraces the use of computer systems because of the crucial role they play in determining the quality of drug and healthcare products. Failure in this duty of care implies, at best, negligence or incompetence; at worst it may infer fraud, and may subject senior personnel to prosecution and legal penalty. However, the net of responsibility falls wider than the pharmaceutical or healthcare company involved. It may jointly or individually include equipment hardware suppliers, software suppliers, system integrators, and system users. Notwithstanding this, GxP regulators hold pharmaceutical and healthcare companies solely accountable for GxP compliance despite the unavoidable supplier dependencies. Examples of matters where such accountability may be cited include deficient design; defective construction; weak or inadequate inspection; incomplete, ambiguous, or confusing user instructions provided by supplier; software installed on an inappropriate hardware platform; the inappropriate use of a system; or the neglect of operational instructions.

Validation

The process of demonstrating GxP has become known as *validation* and involves

> *establishing documented evidence which provides a high degree of assurance that a specific process will consistently produce a product meeting its pre-determined specifications and quality attributes*[6]

and

> *demonstrating that a computerized system is suitable for its intended purpose.*[7]

This definition embraces all uses of computer systems and has been widely adopted, albeit with modifications, by the various GxP regulatory authorities around the world. These are sometimes dubbed computerized systems. The creation of validatable software is, in the first instance, largely a matter of the software developer adopting the basic principles of good software engineering practices under formal documented quality assurance supervision.

Pharmaceutical and healthcare companies must then, in turn, themselves validate all the computer systems used to fulfill operations governed by GxP regulations. Software and hardware must comply with GxP requirements for manufacturing records and equipment, respectively.

This typically affects computer systems that monitor and/or control drug production whose malfunction could possibly affect the safety, quality, and efficacy (during manufacture) or batch tracking (during distribution) of drug products. Other computer systems applications, however, will also be affected. Examples include computer systems used to hold and distribute operating procedures, computer systems used to schedule training and/or determine whether individuals have the necessary competencies to fulfill a particular job role documented in a job specification, and computer systems used to issue company user identities for controlling access to other computer systems. It is thus clear that the list of potential computer system applications requiring validation is extensive. Indeed it has led some individual regulators to suggest that a simpler approach would be to declare that *all* computer systems used within a manufacturing environment, whatever their application, must be validated.

Strong Project Management

To be effective, computer validation must bring together representatives from several disparate groups and complementary disciplines. First and foremost among these are users, despite the fact that they may show no interest in the technology of the computer system and prefer to think of it as a "black" box. Also vital for their endorsement and participation are senior management, who probably do not understand why a remote regulator is interested in the company's software development. Third, the team must include project managers and computer specialists, often overwhelmed by their own workloads and reluctant to shoulder additional tasks. Finally, there must be personnel from the Quality Assurance Department, who may understand better than anyone the operational and compliance benefits that validation will bring.

All these individuals from their diverse backgrounds need to be welded together into a harmonious team under the clear leadership of an empowered Project Manager. Senior management backing, however, is not sufficient on its own to ensure success. Project Managers must motivate their staff. A key success factor here is likely to be their evident willingness to protect the project from unnecessary bureaucracy. They should not acquiesce in the adoption of second-rate ways of working that can clearly be improved. Validation should be as simple as possible but without compromising quality and compliance. This said, it is important to ensure that all project staff are aware of the key project success criteria, and that they understand the fundamentals of GxP principles and practices. From the very start of the project, the Project Managers must avoid the creeping cancer of the sort of culture where "Why document it? I don't know if it works yet!" is heard in the early stages while "Why document it? I already know it works!" are the cries of resistance to validation disciplines later on.

Once the project is under way a positive attitude toward keeping the project timetable on schedule, costs within the budget, and an emerging product with full functionality and compliance needs to be maintained. Project changes must be carefully managed; otherwise, the rate of change overtakes the rate of progress. Careful management of available resources is also very important. Without the necessary skilled resources the project will not run according to schedule. Project Managers need to be aware that the productivity of part-time staff is rarely equivalent to that of full-time staff. Finally, Project Managers must not be too optimistic during the early stages of a project but bear in mind that most projects progress quickly until they are 90% complete. A strong Project Manager will need determination and commitment to drive the project to completion while maintaining quality and compliance.

Keeping Current

Validation practices must keep pace with the technical advances that are occurring constantly within industry. The complexity of computer systems, however, renders them vulnerable to deficiencies in development and operation (e.g., poor specification capture, design errors, and poor maintenance practice). As the use of computer systems increases, so does the potential for public health and safety problems with pharmaceutical and healthcare products. It is not surprising, therefore, that regulatory authorities require validation of computer systems — in other words, documentary evidence of professionalism concerning both their development and operation.[4,8] Even without the requirements for validation, computer systems are extremely difficult to "get right" the first time. All of this must be achieved without delaying their commissioning and operation in what are often fast-track projects with stringent budgets. While there is an unavoidable overhead cost associated with validation, all of this can be offset by business process improvements (manufacturing throughput, laboratory throughput, supply response time, etc.) that constitute tangible financial benefits, equal to or greater than the cost of validation.

Regulatory Observations

Mike Wyrick, chairman of the PDA Computer Validation Committee, published the following top ten quality noncompliance observations recorded by U.S. Food and Drug Administration (FDA) inspectors.[9] The data were collated from over 700 inspection citations issued between 1984 and 1999, and from conference presentations by European inspectors who highlighted similar issues.

1. Testing and Qualification
2. Development Methodology
3. Validation Methodology and Planning
4. Change Control/Management
5. Quality Assurance and Auditing
6. Operating Procedures
7. Security
8. Hardware, Equipment Records, and Maintenance
9. Training, Education, and Experience
10. Electronic Records; Electronic Signatures

Any quality management system must clearly address all these matters because regulatory observations related to computer systems are steadily increasing year by year. Figure 1.5 shows the distribution of FDA observations about computer systems tabulated between 1989 and 1999. This information has been made available through the U.S. Government's Freedom of Information Act. Similar data are not released to the public by other national regulatory authorities, but it is now apparent that regulatory scrutiny of computer systems is increasing right across the global pharmaceutical and healthcare industries.

Before 2000, considerably less than half of regulatory inspections by the FDA and U.K. MHRA included computer systems validation. Today some major pharmaceutical companies are reporting that two thirds of FDA and U.K. MHRA inspections now include some aspect of computer systems validation and this figure is rising annually. This trend is set to continue. Indeed, many regulatory authorities have sent inspectors on training programs dealing with computer systems validation in France, Germany, Norway, Poland, Singapore, and Sweden. As a result, we can expect more regulatory inspections to cover computer systems than ever before. In the future, perhaps up to one fifth of FDA/MHRA inspection time could well be routinely devoted to assessing how various computer systems are used and the steps taken to validate them.[10]

Buyer Beware

Contrary to a widespread misconception, the FDA itself does not approve computer systems; neither does the FDA certify suppliers or consultants. Pharmaceutical and healthcare companies have

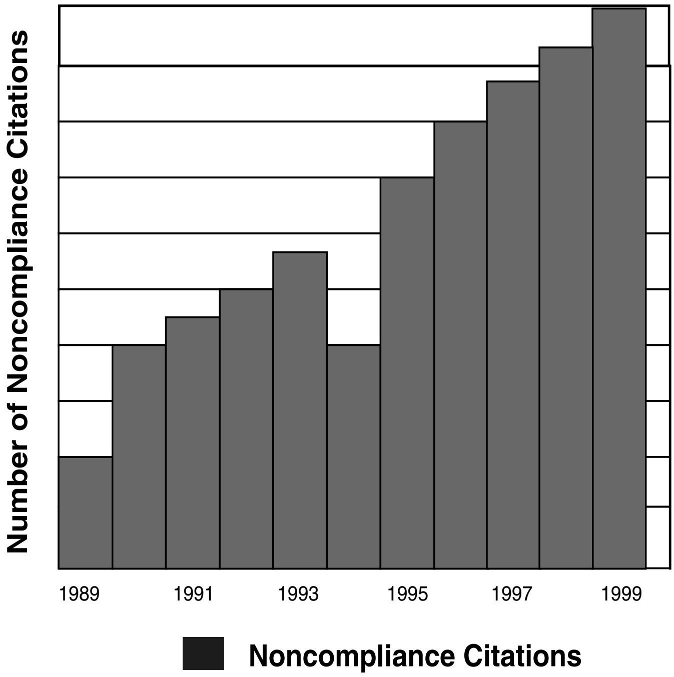


FIGURE 1.5 Increasing Inspections Findings.

always been and remain accountable for validation in many areas, including computer systems. Audits and certifications, however rigorously applied and conscientiously implemented, are no substitutes for validation.

COSTS AND BENEFITS

The costs and benefits of validating computer systems is a subject of many debates and much misunderstanding. The design, development, and commissioning of computer systems can account for up to 20% of the cost of a production plant. With such a large investment, it is important that not only regulatory compliance but also the benefits of improved manufacturing efficiency and product quality be demonstrated convincingly.

Misconceptions

- *Validation is a new development.* In fact, IBM established the concept of a methodology for validation for computer systems in the 1950s. Computer validation has been a requirement in the pharmaceutical and healthcare industries for about 20 years.
- *Validation of pharmaceutical and healthcare computer systems has been specially developed by the FDA to protect their domestic markets from foreign competition.* Recent international free-trade agreements should prevent such restrictive trade and have the power, if invoked, to take offending countries to binding arbitration.
- *ISO 9000 accreditation for quality management fully satisfies the requirements of validation for GxP.* This is not true in relation to the 1994 standards, although ISO 9001: 1994 (supply of goods or services) and ISO 9000-3: 1997(supply of software) and their replacements, ISO 9001: 2000 and ISO 9004: 2000, do provide a good basis for validation.

- *Validation is a one-time event that concludes with a "certification" that the system is validated.* This misconception is usually based on the premise that validation is regulated in the same manner as standards and certification by bodies such as the German TÜV (Technischer Überwachungs-Verein). The GxP regulatory authorities do not certify validation. Validation is an ongoing activity covering development, operation, and maintenance.
- *Validation incurs unnecessary paperwork.* We need to face up to the fact that when validation is poorly implemented there may be some truth in the cynical epithet that "GMP means just Great Mounds of Paper ('Never mind the quality, just feel the thickness of the documents!')." Of course we could retort that when done properly, validation leads to the sort of GMP that means "Getting More Product." Validation that loses sight of its objectives and becomes a bureaucratic and self-serving paper-generation exercise deserves all the contempt it gets. Every document that is created must make a unique contribution to increasing the level of assurance that the system is fit for its intended purpose. That is the acid test of its usefulness, and if it does not meet it, scrap it.

Cost of Validation

Following on from this thought, validation effort is not necessarily proportional to amount of documentation produced. Rather, the level of effort should correspond to the complexity and criticality of the computer system, as well as to its value and the degree of dependency that the plant or organization has on the system. Validation is intended to constitute a reasonable effort by striving to provide a "high degree of assurance"; it is not intended to achieve perfection or absolute proof, nor can such expectations ever be realized.

Firms have generally been reluctant to publish the costs they attribute to validation, but some case studies have been published related to inspected systems, as can be found in the second part of this book. Based on this information and the author's own experience, the following validation metrics have emerged:

- Efficient computer validation should not normally exceed 10 to 20% of development costs when performed concurrently with development. Inefficient validation can easily consume 30% or more of development costs.
- Computer validation costs are estimated to range from 40 to 60% of development costs when performed retrospectively on an existing system. That is, retrospective validation typically costs up to eight times more than prospective validation.
- Computer validation costs can be considerably higher than those metrics quoted above if bespoke functionality has to be incorporated for electronic record and electronic signature compliance.

Many pharmaceutical and healthcare companies attribute higher costs to validation. One reason why higher costs may be quoted is that these include the cost of implementing basic quality assurance practices that should already be in place. A review of major computer validation noncompliance identified by regulators demonstrates that fundamental management controls are often missing or failing. The above metrics are predicated on the assumption that basic quality assurance practices are already in place.

Cost of Failure

The failure to validate to a regulator's satisfaction can have significant financial implications. Noncompliance incidents may lead to delays in the issue of a license, or its withdrawal, and thus an embargo on the distribution of a product in the relevant marketplace (e.g., the U.S.).

Between 1999 and 2002, the percentage of withheld new drug applications by FDA attributable, at least in part, to general validation deficiencies covering process, equipment, computers, etc., rose from 30% to over 75%.[11] The financial consequences of correcting deficient validation might at first glance seem small compared to the typical investment of U.S. $800 million to bring a new drug to market.[12] The real financial impact is the loss in sales revenue arising from a prohibition to market the product. For top-selling drugs in production, citations for noncompliance by GxP regulatory authorities can cost their owner upwards of U.S. $2 million per day in lost sales revenue. One FDA Warning Letter cost the pharmaceutical manufacturer concerned over U.S. $200 million to replace and validate a multisite networked computer system.

The trick is to cost-effectively conduct sufficient validation to ensure GxP compliance but, as illustrated in Figure 1.6, there is always debate over how much is sufficient to fulfill the regulator's expectations. Excessive validation may increase confidence in regulatory compliance, but it does not come cheap. Inadequate validation may actually be cheaper but, in the long term, the cost of regulatory noncompliance could be devastating. This book aims to clarify how much validation is sufficient, to suggest how it can be cost-effectively organized, and also to discuss areas of debate.

There are numerous stakeholders with an interest in successful GxP inspection outcome. GxP noncompliance is likely to reduce public confidence in the pharmaceutical and healthcare industry and the offending company. Political pressures may result in improved industry practices, influence the inspection approaches and methods of regulatory authorities, and review the acceptability of validation standards and guides. The standing of regulatory authorities may be affected if they fail to notice incidents of noncompliance that lead directly to substandard drug products being distributed and used. Associated legal liabilities may arise for both the regulator and offending company. The company's corporate reputation may take years to recover. Drug sales are likely to fall as the consumers of the products, the prescribers, and their patients become uneasy about the quality and consistency of supply. Market confidence in the offending company will be reduced and the brand image tarnished. The reputation of distributors may also be undermined through "guilt by association" with the offending company. Insurance premiums for the company are likely to increase. As an overall consequence, the jobs of all those working for the company and associated suppliers will be less secure.

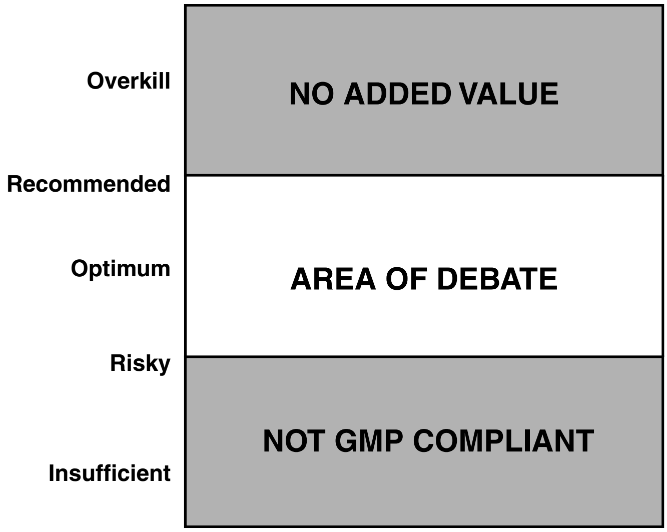


FIGURE 1.6 How Much Validation Is Enough?

Benefits of a Structured Approach to Validation

A structured approach to validation should be delivered efficiently and effectively:

- Less time will be spent defining the boundaries and defending different levels of validation to regulators. Residual noncompliance that slips through the net should be easily discovered through internal audit before a regulator discovers them.
- Suggested compromises to the level of validation from projects and support functions will be more transparent. Noncompliant practices should be reduced.
- Validation skills are more transferable between different computer systems, a key issue where specialist computer validation resources are rare.
- Adopting a standard approach also allows the impact of new and developing regulations and computer technology on the usual GxP validation protocol to be more easily assessed and necessary corrective actions taken in a consistent and timely manner.

Measuring Success

Few metrics have been collected to demonstrate the benefits of validation. At a fundamental level, however, the good practices invoked by validation should ensure computer systems are right first time, every time. Indeed, if the computer system and its plant are already ahead of schedule, the firm could start production earlier than originally planned, and perhaps earn itself U.S. $2 million per day in additional sales for a top-selling drug — not an inconsiderable sum!

Anecdotal evidence of the benefits of effective validation abounds. We may cite the case of two tablet manufacturing and filling lines at GlaxoSmithKline, each controlled by identical computer systems.[3] These lines were installed on different occasions: one was validated from conception to hand-over, while the other was installed without validation. Figure 1.7 illustrates the actual effects of validation by comparing these two otherwise similar projects. In this instance benefits were wide ranging and included the following:

- Improved productivity
- Waste reduction
- Reduced manpower

The profit and loss result to the company was such that the investment in validation for the first line was recovered in just 4 weeks whereas for the second line the payback period from adopting an unvalidated approach was far longer! In another case, validation facilitated a change from traditional stock management practices to the more modern just-in-time (JIT) supply management organization. The payback period of validation costs here may not be as short as for other projects, but the point is that validation should more than pay for itself in the long term through improved operational efficiencies.

Other anecdotal evidence can be quoted to show that validation delivers a *maintenance dividend.*[3] A survey of over 300 applications by Weinberg Associates suggests that maintenance savings within 4 years generally offset the investment in validation. An example of such a maintenance dividend is illustrated by a production planning system at ICI that adopted the principles of validation for about half of its 800 computer programs. Halfway through the project management abandoned the quality approach because there was no perceived project benefit. The total operational life of the system was later examined. It was found that maintenance costs for the software adopting the principles of validation were about 90% *less* than the comparable costs for the remainder of the software. Similar data have been found for MRP II systems. With poor-quality

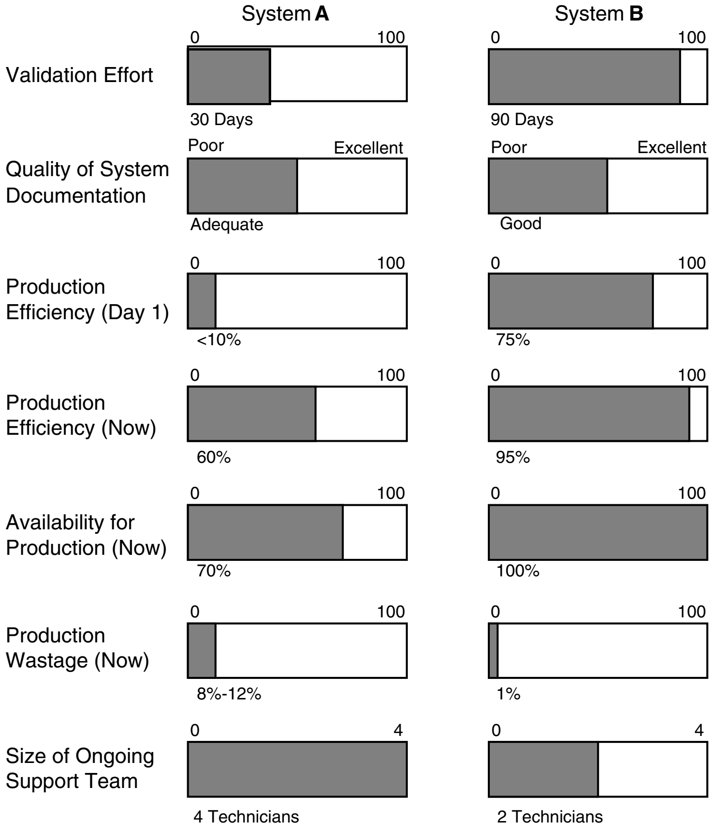


FIGURE 1.7 Anecdotal Computer Validation Benefits.

software typically accounting for 50 to 60% of maintenance costs, validation really does make good business sense.

So much for the tangible, measurable benefits. But validation also yields intangible benefits. *Production staff* work within a quality environment where there are fewer unplanned activities and thus a reduced level of stress. Human error is consequently reduced and productivity improved. It would appear that a significant proportion of production failures could be attributed to human error.[3] *Users of drug products* can rely on a regular supply of consistently high quality. A brand loyalty thus ensues. *Regulators* will develop confidence in the assurance of company product standards. During GxP inspections, regulators will have a positive expectation for compliance, rather than a sense of foreboding of noncompliance. A *pharmaceutical or healthcare company's* track record of GxP compliance will further enhance its corporate reputation and develop a general industry confidence in the company. *Suppliers* who successfully support GxP within such pharmaceutical and healthcare companies will also enhance their own reputations.

Ultimately, GxP compliance protects public health. The scope of this protection exercise is enormous; on average, Europeans and North Americans (including children) currently each receive in excess of ten prescription items per year and purchase six over-the-counter (OTC) medicines. In recent years the pharmaceutical and healthcare industries have avoided major public health incidents, with the notorious exception perhaps of the HIV-infected blood bank scandal in the late 1980s and early 1990s. Validation for GxP has played its part in establishing this track record. Validation for GxP has vindicated itself time after time.

TABLE 1.2
Benefits of Cooperation[3]

Customer	Supplier
Meet user needs	Satisfy customer
Be easier to set up	Be handed over sooner
Be in production sooner	Be paid sooner
Break down less often	Fewer warranty visits
Be easier to repair	Shorter warranty visits
Be easier to further develop	Be easier to modify or upgrade
Be used more efficiently	Good reference site for new customers
Cheaper overall	Cheaper overall
Preferred supplier	Repeat business

GOOD BUSINESS SENSE

Validation should bring benefits to pharmaceutical and healthcare companies, their suppliers, and the end users of drug products. Pharmaceutical and healthcare companies should be able to rely on implementing computer systems correctly first time, every time. Meanwhile suppliers and vendors should, by building in quality rather than vainly trying to test it at the end, be able to reduce delivery costs in the same way. A quality approach should improve time and budget management. There should be savings because of better customer satisfaction and fewer defects that result in:

- Product recalls
- Product returns
- Customer complaints

Customer satisfaction would tend to lead to further orders and promotion of the company within the industry as a highly competent supplier. It is far cheaper to retain an existing customer than to secure a new one.

To avoid insufficient and excessive standards of work and to avoid duplicating tasks in whole or in part, pharmaceutical and healthcare companies and suppliers should work together in partnership. They must be able to work as a team and, as such, must be able to communicate effectively. The parallel mutual benefits between customers and suppliers are outlined in Table 1.2.

A further benefit is the reduced time and effort needed to audit the GxP compliance of a computer system, which yields a distinct marketing advantage to suppliers and gives pharmaceutical and healthcare companies visibility of their own validation capability before and during any GxP regulatory inspection.

PERSISTENT REGULATORY NONCOMPLIANCE

Pharmaceutical and healthcare companies should beware of persistent regulatory noncompliance. We hesitate to cite an actual example, but the lesson is of such importance and the data so clear that the entire industry must heed the warnings of its own history. Between 1993 and 1999, for instance, Abbott Laboratories failed to comply with the FDA's GMP and Quality System (QS) regulations. Despite formal Warning Letters from the FDA issued in 1993, 1997, and 1999, the company failed to correct the problems. The deficiencies identified by the FDA are listed in Table 1.3 and can be summarized under four main topics:

TABLE 1.3
Abbott Laboratories Warning Letters

Date of Inspection	Date of Warning Letter	Deficiencies Found
January 13 to February 14, 1999	March 17, 1997	• Failure to investigate contamination • Failure to maintain equipment • Failure to maintain documents • Failure to maintain an accurate label count • Inadequate methods to describe procedures fully • Failure to record all data collected
September 8 to November 4, 1998	March 17, 1999	• Failure to establish and maintain procedures • Failure to capture all trends and incidents for corrective and preventative action • Failure to validate a process • Failure to establish a quality plan • Degree of control over a process not adequate to assure conformance

- Failure to adequately establish procedures for quality audits, to conduct such audits, and to determine the effectiveness of the quality system
- Failure to develop, conduct, control, and monitor production processes to ensure conformance to specifications
- Failure to validate processes with a high degree of assurance and to document validation activities and results
- Failure to adequately establish and maintain procedures for implementing corrective and preventative action

Repeated promises of corrective actions from senior management had not been delivered to the FDA's satisfaction. The FDA's patience finally snapped during the summer of 1999, when it obtained a court order to ensure that the firm's processes were at last brought into compliance in a timely and orderly fashion. The outcome was a *consent decree* that included:

- A monetary penalty of $100 million payable within 10 days of the decree being entered by the court in its public records
- Agreement to bring pharmaceutical operations into compliance, with failure to do so incurring a further fine of $15,000 per manufacturing process day (up to a capped total of $10 million)
- Medically essential products that had not been brought into compliance within 1 year after the court decree would forfeit 16% of gross proceeds generated by those product sales

Abbott was given 60 days to submit a master compliance plan, validation plan, and protocol for approval by the FDA. This was to be followed by a further period of 30 days after which a final audit report to implement the corrective actions identified would be required, or a period of 10 days in which to submit a schedule for implementing the corrective actions. No further opportunity was afforded the firm to put its own house in order and then to apply to the FDA for reinspection. The FDA also insisted that Abbott hire an independent expert to review progress and report directly back to the FDA on progress against the firm's action plans and the correction of deficiencies. All progress reports had to be submitted by the independent expert to the FDA and Abbott simultaneously, with draft progress reports and comments being retained and made available for FDA inspection. This consent decree graphically demonstrates the extent and severity of the FDA's normal jurisdiction, and how its remit can be extended on a case-by-case basis.

The Abbott affair also established a new process of "disgorgement" (or forfeit) of profits for products that are allowed to stay on the market while corrective actions are implemented. It is important to note in this particular case that the FDA did not question the integrity of Abbott's manufactured products, but rather the whole question of noncompliance surrounded by a lack of GxP. Jane Henney, FDA Commissioner for Food and Drugs, has said, "This action underscores FDA's strong commitment to the enforcement of laws designed to protect patients and consumers. These violations do not necessarily mean that Abbott's products will harm patients, but the firm's failure to follow good manufacturing requirements decreases the level of assurance."[13]

While the Abbott incident did not directly involve computer systems validation, persistent computer systems validation noncompliance should be regarded just as seriously. Other pharmaceutical manufacturing companies such as Wyeth-Ayerst, Johnson & Johnson, and Schering-Plough have found to their detriment that persistent noncompliance can result in a most unpleasant outcome. Initial fines by the FDA on behalf of the U.S. Government have ranged from $30 to $575 million with ultimate costs as high as $1.5 billion (typically 16 to 24% of sales revenue). Senior executives have been identified as individual defendants in U.S. prosecutions. Beware that contempt of the law in the form of GxP practices does not threaten your business!

WIDER APPLICABILITY

Validation concepts, although not the terminology, are also expected for computer, control, and laboratory systems that have the potential to seriously affect Safety, Health, and Environmental (SHE) data protection and financial control. Pharmaceutical and healthcare companies may want to institute common ways of working across these requirements.

REFERENCES

1. U.S. Food and Drug Administration [FDA] (2002), General Principles for Software Validation, Final Guidance for Industry, January.
2. Bennatan, E.M. (2000), *On Time, Within Budget: Software Project Management Practices and Techniques*, Third Edition, John Wiley & Sons, New York.
3. Wingate, G.A.S. (1997), *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice*, Interpharm Press, Buffalo Grove, NY.
4. European Union (2002), Guide to Good Manufacturing Practice for EU Directive 2001/83/EC, *Community Code Relating to Medicinal Products for Human Use,* Volume 4.
5. The Institute of Chartered Accountants in England and Wales (1999), Internal Control: Guidance for Directors on the Combined Code, ISBN 1-84152-010-1.
6. FDA (1987), *General Principles of Validation*, Food and Drug Administration, Center for Drug Evaluation and Research, Rockville, MD.
7. OECD (1995), *The Application of the Principles of GLP to Computerised Systems*, No. 10 OECD Series on Principles of Good Laboratory Practice and Compliance Monitoring, GLP Consensus Document, Environmental Monograph No. 116. Organisation for Economic Co-operation and Development Environmental Directorate, Paris.
8. U.S. Code of Federal Regulations Title 21: Part 211, Current Good Manufacturing Practice for Finished Pharmaceuticals.
9. Wyrick, M.L. (2000), Assessing Progress Towards Harmonisation of Validation Governance in the Global Pharmaceutical Industry, Developing a Business Driven Approach to Computer System Validation for cGMP in Pharmaceuticals, Business Intelligence, March 29–30, London.
10. David Begg Associates (2001), Computers and Automated Systems Quality and Compliance, York (U.K.), June 5–7.

11. FDA (2002), American Society for Quality, Food, Drug, and Cosmetic Division Conference, February.
12. Faiella, C. (2002), Pharmaceutical Industry Report: The Next Pharmaceutical Agenda, Ernst & Young (www.ey.com).
13. FDA (1999), Abbott Laboratories Signs Consent Decree with FDA; Agrees to Correct Manufacturing Deficiencies, Press Release, November 2.

2 History of Computer Validation

CONTENTS

Computer-related systems and equipment began to be introduced into wide-scale use in the pharmaceutical industry in the 1970s. At that time they were mainly used to provide real-time process control and monitoring of production processes. Applications were usually bespoke, having been specially developed for the purpose in an *ad hoc* fashion as required. Later during that decade, computers were adopted as information management systems, based on database engines then available. A common example was their use as primitive electronic batch record systems. Computer systems of that era continued to be based on dedicated hardware such as personal computers (PCs), programmable logic controllers (PLCs), and mainframe computers. As time passed, applications began to make use of newly available Commercial Off-The-Shelf (COTS) software. Despite this increasing use of standard software packages many projects were regularly over budget, late, and — far more seriously — did not fulfill their original user requirements. Improved levels of management acumen and control in this arena were clearly required from both business and regulatory standpoints.

A REGULATORY PERSPECTIVE

SIGNIFICANT REGULATORY DEVELOPMENTS

A general awareness within both the industry and the regulatory community of the need to validate computer systems began to emerge formally in 1979 when the U.S. introduced GMP regulatory legislation that specifically referred to automated equipment.[1] The first widely publicized FDA citation (a formal written regulatory criticism of a perceived noncompliance with the regulations) for computer validation noncompliance was issued in 1985. However, as early as 1982 the FDA publicly stated that it was “nervous” if computer systems were unvalidated.[2] Table 2.1 provides a chronology of regulatory guidance that followed as it affected Good Manufacturing Practice (GMP), Good Distribution Practice (GDP), Good Laboratory Practice (GLP), Good Clinical Practice (GCP), Medical Devices, and Electronic Records and Electronic Signatures (ERES).

GOOD MANUFACTURING PRACTICE (GMP)

In 1983 the FDA issued what became known as the *Blue Book* (because of the color of its cover).[3] This publication gave guidance to inspectors on what was reasonable to accept as validation evidence for computer systems. The Blue Book formally introduced the expectation of a documented life-cycle approach to validation. The aim was to build quality into software from the earliest stages of the life cycle (quality assurance) rather than vainly trying to test quality in at the end (quality control).

Since the FDA had adopted a rather proactive position on computer systems validation, the Pharmaceutical Manufacturers Association (PMA) in the U.S. responded by forming a Computer Systems Validation Committee. This was charged with representing and coordinating the industry’s views. The result was a joint FDA/PMA conference in 1984 at which computer systems validation was extensively discussed. Consequently, a Position Paper reflecting an industry perspective was published the following year. The publication presented an alternative life cycle that included an approach for validating both new and existing computer systems.[4] These came to be dubbed as prospective and retrospective validation, respectively. GxP legislation is unusual in that its regulatory requirements must be met not only in new production facilities but also in those facilities built partially or entirely before the legislation was enacted.

Throughout the 1980s the locus of debate on computer systems validation was primarily in the U.S. Kenneth Chapman, Computer Validation Manager at Pfizer’s U.S. research and development facility at Groton, Connecticut, published a paper[5] summarizing the progress made thus far. During this period the FDA also clarified its position on the following GxP issues:

- Input/output checking[6]
- Batch records[7]
- Applying GxP to hardware and software[6]
- Supplier responsibility[6]
- Application-specific software inspection[6]
- FDA investigation of computer systems[7]
- Software development activities[8]

The European Commission and Australia issued GMP codes of practice in 1989 and 1990, respectively,[9] that complement the U.S. GMP guidance. The European code was subsequently reissued in 1991 as a directive overriding the GMP legislation of Member States.[10] Annex 11 of

TABLE 2.1
Chronology of Published Regulatory Guidance

Date	Significant Regulatory Guidance Publications
1980	First FDA Compliance Policy Guide (CPG) on computer systems — *Computerized Prescription Record Keeping by Pharmacies*
1982	FDA issues CPG on *Identification of Persons on Records*
	FDA issues CPG on *Input/Output Checking*
1983	FDA publishes the Blue Book: *Guide to Inspection of Computerized Systems in Drug Manufacturing*
1984	FDA issues CPG on *cGMP Applicability to Hardware and Software*
1985	FDA issues CPG on *Vendor Responsibility*
1987	FDA issues CPG on Source Code and updates all previous CPGs related to computerized drug processing
1987	FDA publishes technical report on *Software Development Activities*
1988	Japanese MHW issues *GLP Inspection of Computer Systems*
1989	U.K. DoH GLP Monitoring Unit publishes *The Application of GLP Principles to Computer Systems*
1990	Australian TGA publishes *Code of GMP for Therapeutic Goods*, which includes expectations for computer systems
1991	EU publishes *GCP for Trials on Medicinal Products*, which includes computer system expectations
1992	Japanese MHW issues *Computer Control Guidelines for Drug Manufacturing*
1993	EU publishes *GMP for Medicinal Products*, which include computer system expectations
1995	U.S. EPA releases final version of *Good Automated Laboratory Practice*
	FDA publishes *Glossary of Computer Terminology*
	OECD publishes its own updated version of *The Application of GLP Principles to Computerised Systems*
1997	FDA issues 21 CFR Part 11 for *Electronic Records and Electronic Signatures* with preamble discussion and guidance
	FDA publishes draft software validation expectations for medical devices
	Japanese MHW issues expectations for *Retention of Electronic Records*
	MCA issues guidance on *Year 2000 Conformity*
	MCA issues guidance on *Electronic Signatures*
1998	FDA *Premarket Submissions for Software Contained in Medical Devices*
	FDA publishes *Computerized Systems for Food Processing*
1999	FDA issues CPG *Enforcement Policy for 21 CFR Part 11*
	FDA publishes *Computerized Systems Used in Clinical Trials*
	FDA issues CPG *Year 2000 Computer Compliance*
2000	ICH publishes *GMP for API* with section on computer systems
2001	EU adopts PIC/S guidance on *Validation and Qualification*
	FDA starts to publish draft guidance to accompany 21 CFR Part 11
2002	FDA publishes *General Principles of Software Validation for Medical Devices*
2003	FDA revokes previous Part 11 guidance and issue scope and applicability guidance
	U.S. EPA issues *Cross-Media Electronic Reporting and Record Keeping Rule* (CROMERRR)
	PIC/S issues *Good Practices for Computerised Systems in Regulated "GxP" Environments*

the European GMP code covering computerized systems was an extract from the joint European regulators' Pharmaceutical Inspection Conference *Guide to Good Manufacturing Practice for Pharmaceutical Products*.[11] It covers requirements for:

- Personnel
- Validation life cycle
- Computer system operating environment
- System description
- Software quality assurance
- Built-in entry and processing checks
- Testing

- System security
- Verification of critical data
- Audit trails for data entry and amendments
- Data integrity
- Data backups
- Business continuity planning
- Recovery procedures
- Error tracking
- Supplier contracts
- Release security

EU GMP Annex 11 on computerized systems was later complemented by Annex 15 on Validation and Qualification.[12] This addition outlined the expectations for:

- Validation Master Plan
- Design Qualification
- Installation Qualification
- Operational Qualification
- Performance Qualification
- Validation Reports

The Japanese Ministry of Health and Welfare (MHW) [later Ministry of Health, Labor and Welfare (MHLW)] issued its computer validation guideline in 1993.[13] It specifically avoided the "validation" and "qualification" terminology, although it covered rudimentary validation requirements:

- System development manual
- Development schedule
- System engineering documentation
- Program (software) specification
- Test planning (function, capability, reliability, operation)
- Operational controls
- Document storage and retention times

A significant difference between the MHW guideline and other FDA and EU guidance, however, is that the Japanese guidance only applies to networked applications; stand-alone applications such as PLCs embedded in equipment and laboratory instrumentation are excused compliance. This position is currently being reviewed, and it is likely that those stand-alone systems that manage electronic records will also require validation.

The issue of computer systems validation assumed a high profile within industry in Europe in 1991 when two European pharmaceutical manufacturers were temporarily prohibited from exporting their products to the U.S. This was because their computer systems were found not to comply with regulatory expectations. The position of the FDA was clear: these manufacturers had failed to satisfy FDA concerns that computer systems should:

- Perform accurately and reliably
- Be secure from unauthorized or inadvertent changes
- Provide adequate documentation of the processes they implemented

The manufacturers had sincerely believed that their interpretation of the GMP legislation fulfilled the legal requirements, but the FDA's interpretation of GMP and its practical implications was not the same! Hence, the terminology adopted in modern cGMP (current Good Manufacturing Practice) legislation has been standardized and defined in order to foster *a consensus of understanding* of the actual validation practices and standards expected by regulatory authorities.

Because the regulatory authorities appreciated the difficulty in determining what constitutes sufficient validation for different computer systems, they continued to issue guidance materials. For instance, in 1995 the FDA revised its GMP Code of Federal Regulations affecting computer systems to acknowledge the complexity and reliability of contemporary systems.[1] Pharmaceutical and healthcare companies could now justify a baseline for validation based on the technology risk and the operational track record of the computer system in question. This development was widely welcomed by practitioners. The FDA also issued a *Glossary of Computer Terminology*[14] and a guideline for computerized systems used in food manufacturing.[15]

The International Conference on Harmonization (ICH), representing FDA, EU, and Japanese regulatory authorities, produced guidance in 2000 on the manufacture of active pharmaceutical ingredients (APIs) that included computer validation expectations.[16] Not too surprisingly, essential principles were unchanged compared to computer validation for finished drug products. The key topics covered were:

- Initial validation and operational compliance
- Testing and qualification
- Security and data integrity
- Change control
- Calibration of equipment and instrumentation

Future developments are envisaged by the Australian TGA and U.K. Medicines and Healthcare products Regulatory Agency (MHRA) regulatory authorities. TGA intends to supersede its GMP computer validation regulatory guidance with the Pharmaceutical Inspection Convention Scheme (PIC/S) GxP computer guidance described later in this chapter. The MHRA, meanwhile, is proposing that the EU GMP Annex 11 on computerized systems be updated to better reflect the core principles incorporated in the PIC/S guidance. Other members of PIC/S are also expected to adopt the GxP computer validation guidance. Current computer validation guidance from PIC/S, FDA, and the Japanese Pharmaceutical Manufacturers Association, now being discussed with MHLW, all reflect consensus on the GAMP Guide providing industry sector good practice.

Good Distribution Practice (GDP)

The U.S. issued specific distribution requirements in 1990 to complement the basic provisions established earlier in its GMPs.[17] Although computer validation is not specifically identified as a requirement, separate electronic record/signature legislation (21 CFR Part 11) requires computer systems handling defined distribution records to be validated. Records should be sufficient to track the origin and destination of medicinal products primarily in support of possible customer returns and product recall. Key GDP topics identified in the U.S. legislation that validation should address in relation to computer systems include:

- Procedural controls
- Records retention and retrieval
- Security controls
- Disaster recovery
- Temperature monitoring of products in storage and transit

The EU has also developed a specific Directive for the wholesale distribution of medicinal products.[18] The Directive places similar emphasis on the topics specified in U.S. legislation with the addition of calibration for monitoring devices. Computer validation is expected and EU GMP Annex 11 on Computerized Systems applies. The EU GMP and GDP directives were consolidated in 2001[19,20] but with no change to computer validation expectations (EU Directives 2001/82/EC and 2001/83/EC).

Good Laboratory Practice (GLP)

The U.S. GLP regulations do not have specific requirements for computer systems validation. Instead the U.S. GLPs refer to equipment requirements and these are applied to instrumentation and computer systems as appropriate. For instance, 21 CFR 58 requires:[21]

- Appropriate design
- Adequate data processing capacity
- Suitable physical location for operation, inspection, cleaning, and maintenance
- Adequate testing
- Calibration
- User procedures
- Defined remedial action for failure or malfunction

In 1998 the Japanese MHW published an annex to its GLP regulations specifically on computer systems.[22] The guide included specific recommendations for prospective validation of internally developed computer systems and externally purchased computer systems. It specifically calls for in-built testing functionality within applications to be documented and approved. Retrospective validation is also discussed.

One year later the U.K. Department of Health (DoH) GLP Monitoring Unit published its expectation for computer systems in laboratories conducting human health and environmental safety studies.[23] It identifies laboratory management responsibilities for:

- Identification of computer systems
- Defined specifications for computer systems
- Control procedures for software programs
- Security access on computer systems
- Archiving records
- Quality assurance
- Staff training

The OECD issued a consensus document for GLP computer validation[24] in 1995. It was prepared with the cooperation of the European Agency for the Evaluation of Medicinal Products (EMEA), the FDA, and Japan's MHW. The OECD document should be considered an extension of the EU GMP Annex 11, giving practical advice on how to meet various regulatory requirements. Topics included in the OECD document not previously covered within Annex 11 include:

- Management responsibilities including the duties of Quality Assurance
- Training records for personnel involved in computer systems validation
- Reliable communication interfaces between integrated systems/peripherals
- Retrospective evaluation of validation requirements for existing systems
- Maintaining the validated status of a computer system during its operational life
- Documentation requirements for management policies and source code
- Ten basic procedural controls for computer validation
- Archiving requirements for software, data, and supporting documentation

Publication of the OECD document coincided with the final publication of the U.S. Environmental Protection Agency's (EPA) *Good Automated Laboratory Practices (GALPs) Guide*.[25] This guide had been in draft for 6 years and was aimed at laboratory systems that are used to collect and manage data. The EPA GALP document describes a sequence of nine steps for managing computer validation involving one or more laboratories. It begins with collating system inventories, followed by conducting a compliance gap analysis, and concludes with advice on auditing and inspections. The principles for computer validation adopted by both the OECD and the EPA are almost identical.

Good Clinical Practice (GCP)

The EU published its Good Clinical Practice for Trials of Medicinal Products in 1991.[26] The U.S. had no single equivalent to the EU GCP document. Computer system requirements in the EU GCPs included:

- Validated
- Detailed description of use
- Authorized data entry
- Data maintenance
- User documentation
- Data migration
- Archiving

In 1996 the ICH, representing FDA, EU, and Japanese regulatory authorities, published GCP expectations that included factors affecting computer validation.[27] The FDA, EU, and Japan have all adopted the ICH GCPs. Three key principles that are relevant to computer applications are outlined:

- Qualified personnel
- Data integrity
- Comprehensive quality assurance

Specific requirements in support of these principles are given for:

- Record and report creation, maintenance and archive
- Quality management systems and procedures
- Data handling and record keeping (including requirements for electronic data)
- Internal audit and deviation reporting

The latest FDA publication on computer systems validation was issued in 1999 and dealt with clinical software.[28] This guidance complements the ICH GCP requirements affecting computer validation that we discussed earlier. Although designated for GCP applications, this material has implications for most GxP computerized systems as it presents the FDA's current expectations. Topics covered include data entry, security, system controls, and training, with some specific advice on electronic records and the certification of electronic signatures.

Medical Devices

By 1991 the sophistication of medical devices and their dependency on software led the FDA to issue specific guidance on computer validation.[29] The guidance had similarities with the ISO 9000 quality principles and included expectations on:

- Environmental control for computer hardware
- Calibration
- Master software and production copies

The software master and production copy guidance can be applied elsewhere to GxP spreadsheets.

The FDA issued updated software validation guidance in 2002.[30] It provides not only technical guidance but also attempts to provide the rationale behind its thinking on these expectations. The scope of guidance includes software resident on the medical device and software used to support the manufacturing process for medical devices. The benefit of a quality management system is extolled and a full life cycle is promoted covering project, operation, and maintenance. The key role that a good User Requirement Specification (URS) plays is emphasized. Risk management is a central theme, and different approaches appropriate to bespoke (custom) software and COTS software are presented. The use of conventional validation and qualification terminology is discussed and specifically not mandated.

European medical device requirements are defined in EU Directive 93/42/EEC published in 1994.[31] National legislation and regulatory authorities such as the U.K. Medicines and Healthcare products Regulatory Authority (MHRA) enforce this directive. Four classes of device are defined, each of which requires a visible CE marking of conformity. CE marking of conformity means that the manufacturer is satisfied that the medical device conforms to relevant EU Directives and that it is fit for purpose. Technical documentation supporting conformity related to automated devices includes:

- Quality control procedures
- Specifications including appropriate drawings and circuit diagrams
- Risk analysis that the device will be fit for purpose
- Manufacturing and test records to show compliance with the defined procedures and specifications
- Qualification tests relevant to the intended use of the device
- Periodic review of operability and compliance

The U.K. MDA (later MHRA) highlights the presumed establishment of, and compliance with, a Quality Management System (QMS) covering:[32]

- Design/development
- Production
- Installation
- Final inspection and testing
- Servicing

EU regulatory authorities can audit manufacturers for compliance, or alternatively they can delegate authority for deciding whether the CE marking is appropriate to approved competent third-party organizations.

Electronic Records and Electronic Signatures (ERES)

The FDA's long-awaited regulation on electronic records and electronic signatures (21 CFR Part 11) finally became effective on August 20, 1997,[33] 4 years after the first issue of draft guidance. The regulation applies to both new and existing systems used in the pharmaceutical and healthcare industries. Topics covered by the regulation include:

- Electronic Records
- Data Integrity

- Access Control
- Physical Security
- Training and Operating Procedures
- Electronic Signatures
- Biometrics
- Validation

Compliance with the 21 CFR Part 11 regulation required computer systems to manage records entirely electronically; management of printed copies of electronic records (known as hybrid systems) was not deemed acceptable as a long-term solution. This meant many computer systems needed custom developments either by pharmaceutical and healthcare companies or their suppliers to satisfy the regulation. Where such developments were not possible, systems had to be replaced. In 1999 a Compliance Policy Guide (CPG) was issued describing the FDA's enforcement strategy for 21 CFR Part 11.[34] This acknowledged it would take industry a period of time to come into full compliance with the regulation.

During 2001 and 2002 the FDA attempted to address industry concerns regarding the practical application of 21 CFR Part 11[35–37] by publishing a number of draft guidance documents covering terminology, validation, timestamps, electronic records maintenance, and electronic copies of electronic records. These draft guidance documents were subsequently withdrawn together with the FDA's enforcement policy on Part 11 in favor of a new, narrower interpretation of the scope and application of electronic record requirements.[38] The 21 CFR Part 11 regulation itself remained unchanged. Electronic records only come under the revised FDA interpretation of 21 CFR Part 11 when business processes use them in precedence over printed records. FDA enforcement now concentrates on those aspects of 21 CFR Part 11 supporting Predicate Rule requirements for secure and reliable records and that portion of 21 CFR Part 11 that describes electronic signatures requirements.

Japan issued requirements regarding retention of electronic records on magnetic media in 1997.[39] These addressed:

- Protection of electronic records against being accidentally/unintentionally overwritten, deleted, or confused with other records
- Reproducing electronic records as paper copies or electronically displayed copies
- Controlling magnetic media to preserve integrity of stored electronic records

Europe issued its own directive on electronic signatures in 1999.[40] Although originally developed for e-commerce, it is also applicable to GDP and GMP applications. The MCA is clarifying the scope of its use for GCP and GLP applications, and at least a minimum degree of applicability is certain. European regulatory authorities refer to ISO 17799 on information security management[41] and reference to advice on the admissibility of electronic records.[42]

At the time of publication of this book there is a general reappraisal of electronic record and electronic signature requirements by various regulatory authorities around the world. The U.S. EPA, for instance, is preparing to issue regulatory requirements on handling Cross-Media Electronic Reporting and Record-keeping (CROMERR). Electronic records are now to be separated from electronic reporting (submissions) to the agency, and the focus of the regulation moved to the latter. Keynote developments that may affect other electronic record/signature regulations under review include:

- Electronic signatures are only required where they are required on equivalent paper reports
- There must be no delegation of devices to create an individual's electronic signature by another person

- Individuals must be given formal notice not to compromise their unique signature (as equivalent to handwritten signature) with the possibility of individuals being asked to sign a printed statement to this effect with a handwritten signature

The FDA is meanwhile reviewing 21 CFR Part 11 with a view to return to the principles of record controls already identified in Predicate Rules. Key topics being reviewed include:

- The scope of what constitutes a record
- The need for reprocessability of electronic records
- Practical ways of handling long-term archiving

The Japanese MHLW is considering similar topics as it looks to extend its current electronic record/signature requirements based on various discussions with industry.[43] Its basic expectations are the same as 21 CFR Part 11 but with a couple of key differences. First, electronic records are defined as data stored for the purpose of long-term retention. Transitory data that might be stored for a number of cycles or printed to paper without maintaining an electronic copy are not considered electronic records. Second, reprocessing electronic records is not required so long as sufficient information is captured to provide evidence to support the original construction of the record.

In addition, PIC/S has also prepared guidance discussing regulatory expectations for electronic record and electronic signature.[44] The PIC/S document has a wider scope than just ERES and is discussed in more detail in the next section of this chapter. Before then, it is worth noting that the ERES requirements are basically the same as U.S. 21 CFR Part 11 with two notable exceptions. First, PIC/S places emphasis on the application of a risk assessment process to identify the electronic records to which audit trail and other ERES requirements apply. Justifications and rationales will be audited during inspections. This is a different approach to that being adopted by the FDA. The FDA suggests in the preamble to 21 CFR Part 11 and subsequent guidance that ERES requirements apply to records identified in predicate rules and have developed a guidance note specifying these records.[45] This specification of what precisely constitutes electronic records subject to ERES requirements is being made independently of the computer applications to which they relate and is proving difficult to define. Second, there is no expectation that pharmaceutical and healthcare companies formally notify their regulatory authorities about their use of electronic signatures and legal equivalence to handwritten signatures. However, there is an expectation that sites being inspected will maintain inventories of computer systems that identify ERES applicability.

Collective GxP Requirements

In most countries, some discretion is permitted in interpreting GxP requirements. If a court is to successfully prosecute a pharmaceutical or healthcare company, it must be convinced that there is sufficient evidence to prove that the company has deliberately intended to flout the governing legislation. In the U.S., however, the declared judgment of a court of law on supplementary GxP information, including guidance, was to regard it as substantive. The effect of this principle was that the FDA's *advisory opinions* became binding on the industry. However, in August 1990 the FDA announced that it no longer considered such advisory opinions as binding on companies, since the implied restrictions that this would impose on commercial businesses would be unconstitutional. From then onward the FDA's interpretations of the regulations in the Compliance Policy Guides, Guides to Investigators, and other publications by its authors have been regarded as being for *guidance* only.

Two experienced regulatory inspectors, Ronald Tetzlaff and Anthony Trill, have published papers describing inspection practices for computer systems adopted by the FDA and MCA (later MHRA), respectively.[46,47] These papers present a comprehensive perspective on the current validation expectations of the regulatory authorities. Topics covered included the following:

- Life-cycle approach
- Quality management
- Procedures
- Training
- Validation protocols
- Qualification evidence
- Change control
- Audit trail
- Ongoing evaluation

While a common methodology for computer systems validation incorporating these topics has become established within the pharmaceutical and healthcare industries, the regulatory authorities continued to be disappointed with their findings in companies:

> *The major problems observed are in the validation of various manufacturing and control operations. We are seeing more problems now with operations run by computers and the validation of these computer programs.*[48]

Their disappointed expectations tend to comprise one or more of the following:

- Incomplete documents
- Insufficient detail in documentation
- Absent documentary evidence

The PIC/S has recently issued an internationally harmonized validation guide for the validation, control, and use of computer systems in GxP-regulated applications.[44] The guidance is intended for both "external" reference (i.e., pharmaceutical and healthcare companies, and their suppliers) and for "internal" use by regulatory authorities. Contributors to the guidance include the German RP, Swedish MPA, Swiss IKS, U.K. MHRA, and U.S. FDA regulatory authorities. The contents cover:

- Life-cycle project management and documentation
- Software and hardware selection considerations
- User requirements
- Functional specification
- Testing
- Operational considerations
- Electronic records and electronic signatures
- Personnel
- Inspection considerations with inspector aides-mémoire
- References and glossary

The PIC/S guide encourages GxP inspectors to take a *holistic* approach to inspections, not just confining the inspection to the computer itself but also considering the wider equipment/processes associated with the use of the system. GAMP Forum and Parenteral Drug Association (PDA) scalable guidance are endorsed and the benefits of ISO 9000-3 and IEEE 1298 for software quality management recognized.

If adopted, the PIC/S guide will be applicable to Australia, Canada, the European Union, Finland, Hungary, Iceland, Liechtenstein, Malaysia, Norway, Romania, Singapore, Slovak Republic, Sweden, and Switzerland. It will be used to train regulatory authorities and as the basis to update

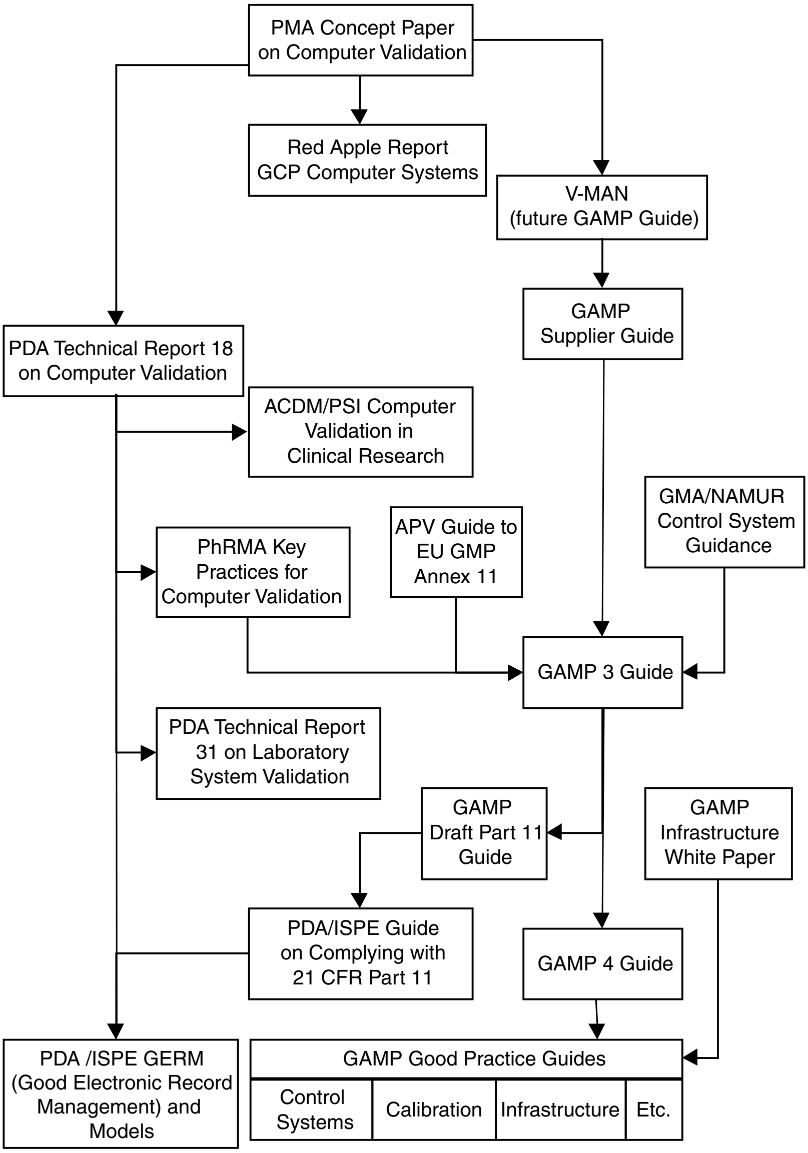

PMA Concept Paper on Computer Validation
Red Apple Report GCP Computer Systems
V-MAN (future GAMP Guide)
GAMP Supplier Guide
PDA Technical Report 18 on Computer Validation
ACDM/PSI Computer Validation in Clinical Research
GMA/NAMUR Control System Guidance
APV Guide to EU GMP Annex 11
PhRMA Key Practices for Computer Validation
GAMP 3 Guide
PDA Technical Report 31 on Laboratory System Validation
GAMP Draft Part 11 Guide
GAMP Infrastructure White Paper
PDA/ISPE Guide on Complying with 21 CFR Part 11
GAMP 4 Guide
PDA /ISPE GERM (Good Electronic Record Management) and Models
GAMP Good Practice Guides
Control Systems
Calibration
Infrastructure
Etc.

FIGURE 2.1 Industry Guidance Genealogy.

EU GMP Annex 11. Meanwhile it is important to recognize that the role of the PIC/S document is to provide guidance and support to existing national regulations.

DEVELOPING INDUSTRY GUIDANCE

Important industry developments are described below in the chronological order of their development. Figure 2.1 presents the sequence of developments and how they are interrelated.

PMA's Concept Paper

Responding to the FDA's proactive position on computer systems validation, the U.S. PMA formed a Computer Systems Validation Committee to represent and coordinate the industry's viewpoint.

The results were a joint FDA/PMA Conference in 1984 discussing computer validation and, 2 years later, the publication of an industry perspective.[49] The publication presented a life-cycle approach for validating both new and existing computer systems. Guidance on existing systems was required because the FDA expected both existing and new computer systems to be brought into line with its expectations for validation, and its requirements were not limited to just prospective validation of new computer systems.

Red Apple Consensus Report

In 1987 a workshop was convened at the Red Apple Conference Center in Heber Springs, Arkansas, U.S.A., with the explicit purpose of generating a validation guide for GCP/GLP computerized data systems. The output of the workshop was published in 1988 and became known as the Red Apple Report. It covered:

- System development life cycle including operation and maintenance
- System verification and validation including retrospective validation
- Quality role during development, operation, and inspections
- Special topics discussing centralized systems, distributed systems, and systems integration
- Security access including data integrity
- Computer communications within and between systems

The FDA contributed to the guide and focused on the practicalities of validation. It set the tone for future industry guidance, demonstrating the benefit of collaboration between regulatory authorities and industry.

GAMP Supplier Guide

The U.K. Pharmaceutical Industry Computer Systems Validation Forum (PICSVF — also known as the U.K. FORUM) was established in 1991 to facilitate the exchange of validation knowledge and to promote the development of a supplier guide for computer systems validation projects. Suppliers were struggling to understand and implement the various interpretations and requirements of GxP presented by companies. The guide was a collaborative effort between pharmaceutical companies and the U.K. MCA; supplier organizations were not directly involved.

PICSVF developed a prototype supplier guide called V-MAN (Validation Management) that was first circulated as a draft within the U.K. and then Europe. The first issue (version 1.0 in paper form) was released in 1994 and a further issue on CD (version 2.0) was released through the auspices of the ISPE in 1996.[50] By this time the PICSVF had renamed itself the GAMP Forum (which is certainly easier to pronounce!), and its guide became known as the GAMP Guide. The guidance included a number of outline validation procedures that were intended to be used as the basis of developing final procedures. This was the first time such hands-on guidance had been produced.

PDA's Technical Report on Computer Validation

The U.S. PDA issued Technical Report 18 that described a framework for computer validation in 1995.[51] It discussed the requirements for pharmaceutical companies to have computer validation policies and procedures. The practical application of the PMA life cycle is discussed but at a high level. The Technical Report also provided detailed checklists for developing System Requirements/Specification or conducting Vendor Evaluations.

PhRMA's Computer Validation Key Practices

The paradigm shifts in computer validation between 1986 and 1997 were chronicled in a "key practice" document published by PhRMA's Computer Systems Validation Committee.[52] It noted two fundamental changes in the application and validation of computers. First, there was a distinct move away from "closed" to more "open" computer system architectures. Second, there was a need for a validation life-cycle model that was independent of the various project methodologies that might be used. With this in mind, life-cycle validation phases had to be described in terms of inputs, processes, and outputs (plans, activities, and reports). Attention needed to be focused on critical practices such as "testing" and to define who ought to be responsible for those practices. Change control now had a significant correspondence with configuration management, while management rather than administration was required for security. Similarly, suppliers of computerized systems required managing rather than just being selected and nothing more. New concepts now appearing were risk analysis, management of record retention and disposal, and planning for the reengineering, replacement, or retirement of a computer system.

GAMP 3

The third edition of the GAMP Guide was published in 1998.[53] The nature of the Guide extended supplier requirements to include guidance for the pharmaceutical and healthcare companies. Emphasis was placed on the partnership necessary by pharmaceutical and healthcare companies and their suppliers to successfully achieve validation. GAMP 3 addressed most of PhRMA's key practices and included new material on:

- The forthcoming millennium (Y2K) bug
- Mutual roles of pharmaceutical and healthcare company and supplier when validating applications based on COTS software products (introduction of the X-Model)
- Applying the V-Model life cycle to large IT systems such as MRP II and LIMS with focus on critical data items associated with drug product quality
- Validating process control systems (procedures and checklists) developed by a joint committee of two German organizations devoted to the use of measurement, computers, and control devices (GMA and NAMUR)
- A detailed interpretation of EU GMP Annex 11 (computerized systems) written by the German Association for Pharmaceutical Process Engineering (APV)

A year earlier the GAMP Forum supported the formation of the Supplier Forum dedicated to the needs of suppliers. It was run along the same lines as the GAMP Forum and eventually became an official part of the GAMP Forum in 2001. Guidance materials produced specifically by suppliers for suppliers included user specifications, testing, and receiving customer audits.[54–56]

PDA/GAMP Harmonized Terminology

During 1999, a joint PDA/GAMP working group published a harmonized glossary of technical terms in this field.[57] These terms are included in the Glossary of this book.

GAMP Infrastructure and PDA Network Guidance

In the same year the GAMP Forum published draft guidance on the validation of IT infrastructure[58] and the PDA published a paper on network qualification.[59] The appearance of this guidance was timely in the light of the widespread introduction of Manufacturing Resource Planning (MRP II) systems and other distributed client server applications. These applications depended heavily on the management and control of the so-called *desktop environment* (workstations, networks, servers, and associated services). The GAMP Forum and the PDA agreed that infrastructure should not be validated as such but rather qualified in support of validated applications. The areas defined as needing most attention were:

- Configuration management
- Change control
- Installation qualification

PDA Technical Report on Laboratory Systems

In 1999 the PDA published Technical Report 31 on validating computerized laboratory data acquisition systems.[60] The key themes behind the document were scalability and risk management, although it was not until the new millennium that regulators and industry alike tackled these themes head on.

GAMP 4

The GAMP Forum published the fourth version of its Guide, GAMP 4, at a public seminar in Amsterdam in December 2001.[61] The new edition is modular so that new elements can be added or existing guidance updated without the need for a total revision (see Figure 2.2).

The top tier of GAMP 4, the Guide itself is a single document that draws together the body of knowledge for computer validation: key validation principles and practices, and how at a conceptual level they can be applied to determine the extent and scope of validation for different types of system, ensuring that validation is scaleable. The APV interpretation of EU GMP Annex 11 on computerized systems is included in the Guide. The strategic framework includes a collated set of supporting, rationalized, and revised Quality Management Procedures for computer systems validation. There are three categorises of procedures: Management, Development, and Operation.

The second tier of GAMP 4 consists of a number of modules each presenting guidance for a practical implementation of computer validation (good practice modules). Modules will include global systems validation (MRP II, LIMS, etc.), process control system validation (including GMA/NAMUR), validating analytical laboratory systems, calibration, and compliant infrastructure.

Workshop training materials form the third tier of GAMP 4. These materials will be taken from ISPE Seminars and are aimed at facilitating hands-on training with real-world examples dealing with both mundane and novel issues. This material will be produced as part of ISPE Seminars.

GAMP 4 places particular focus on streamlined cooperation between users and suppliers, scalable computer validation, and development of validation strategies based on risk management. The GAMP Guide has wide consensus and is now used as a reference text by most regulatory authorities interested in computer validation, including the FDA and the U.K. MHRA. Its success has been accompanied by a substantial growth in the size and geographic spread of the GAMP

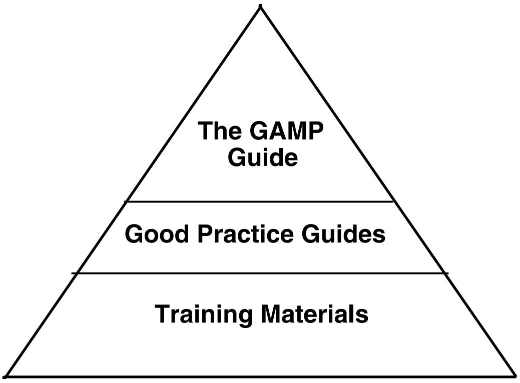


FIGURE 2.2 GAMP 4 Modular Hierarchy.

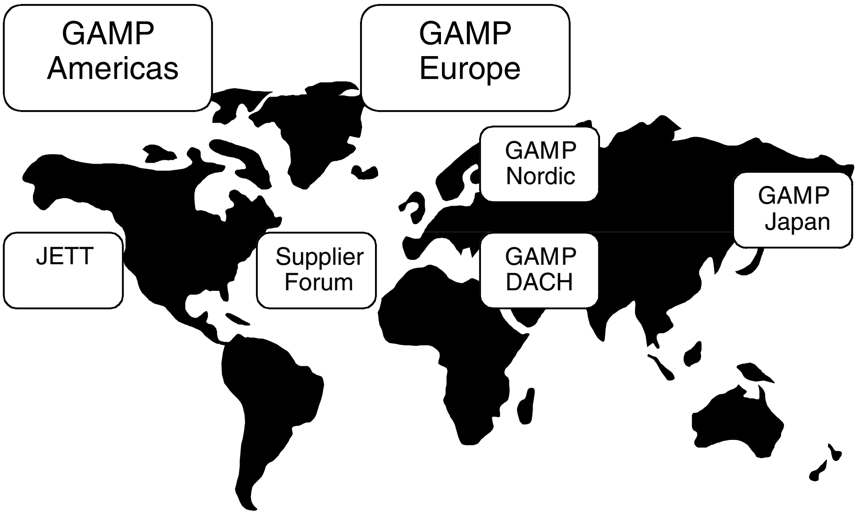


FIGURE 2.3 Current GAMP Family.

Forum (see Figure 2.3). The GAMP Forum was formally incorporated into ISPE at the beginning of 2001 and is now a formal technical subcommittee of ISPE.

PDA/ISPE Good Practice and Compliance for Part 11

In 1998 the GAMP Forum issued a draft document to solicit industry feedback on the practical implementation of this Rule.[62] The need for guidance was exacerbated by the FDA's issue of a Compliance Policy Guide describing how it intended to enforce 21 CFR Part 11 in the summer of 1999.[63] The main problems raised by the GAMP Forum regarding compliance with the regulation are discussed in Chapter 13; they are:

- Password Expiry
- Retention of Data
- Audit Trails
- User Profiles
- Timeouts
- Virus Management
- Electronic Signatures
- Timestamps in Multiple Time Zone Systems
- E-mail
- Hybrid Systems

The work was finally published in 2001 as part of the collaborative PDA/ISPE series of guidance on Good Practice and Compliance for Electronic Records and Signatures.[64]

Other work of an ongoing nature is being undertaken by the PDA, which has established a Part 11 Task Group. Guidance material on Good Electronic Record Management (GERM) has been developed and copublished with ISPE to complement the GAMP Forum's Part 11 Guidance document.[65] The main topics covered are:

- Definition of Electronic Records
- Organizational Controls
- Operations and Infrastructure

- Transactions (audit trails, sequence checks, electronic signatures, etc.)
- Record Retention
- Personnel Qualification and Training
- Hybrid Systems and Controls

In particular a number of current good practices are suggested as a means of focusing the organization's immediate actions to satisfy Part 11. The GERM guidance will be complemented by "Models" guidance on how to achieve practical compliance with a variety of computer systems.

Body of Knowledge

Table 2.2 collates the key components of the body of knowledge that have been established for computer validation. It lists both regulatory requirements and the industry's own guidance. There are many other publications available, and the selection presented does not pretend to be exhaustive (e.g., draft regulatory requirements and industry guidance have not been included). Table 2.3 relates the topics covered by regulatory requirements and industry guidance to the chapters in this book, with the hope that this will aid readers in locating key sources of information they might wish to refer to after reading this book.

PIVOTAL INSPECTION INCIDENTS

Figure 2.4 provides an overview of the changing topical issues that have faced pharmaceutical and healthcare companies over the last 20 years. The pivotal inspections behind these topical issues are reviewed below.

The first high profile examples of computer validation noncompliance were publicized in 1985 and occurred in the U.S. The incidents involved a laboratory system at the Wyeth nonclinical laboratory[66] and a computerized dispensing system at Boehringer Ingleheim's production facility.[67] The noncompliance concerned:

- Inadequate hardware and software validation
- No revalidation after significant hardware and software changes
- Lack of environmental controls
- No backup power supply to prevent uncontrolled system shutdowns
- Unsecured laboratory computer programs and raw data
- Inadequate security procedures for the HVAC computer system

The potentially devastating consequences of GxP noncompliance in relation to computer systems became apparent in 1988. Deficient software in a data management system controlling a blood-plasma bank could have led to the issue blood infected with HIV (Human Immunodeficiency Virus).[5] Similarly, computer systems are capable of endangering public health by manufacturing and erroneously releasing drug products of deficient quality.

Eli Lilly's site at Indianapolis, Indiana, U.S.A., had an important inspection by the FDA in 1990. This included a review of the company's computer systems validation, which resulted in the company not only developing a computer validation methodology with supporting procedures but also establishing a program to validate its other computer systems across the world to this new standard. In the following year Europe had its first taste of the new expectations for computer systems validation.

Computer systems used at production sites in Italy and England belonging to ICI Pharmaceuticals, Glaxo Laboratories, and other pharmaceutical companies were found to be noncompliant during an inspection program conducted by FDA inspector Ronald Tetzlaff in 1991. The computer

TABLE 2.2
Body of Knowledge — Regulatory Expectations

			Supporting Processes						Project Delivery						Operation and Maintenance											Phase-Out			
Title	Date of Publication	Applies to	Training	Document Management	Change Control	Configuration Management	Self-Inspection	Managing Deviations	Project Initiation & Validation Determination	User Requirements & Supplier Selection	Design & Development	Coding, Configuration, & Build	Development Testing	User Qualification & Authorization to Use	Performance Monitoring	Repair & Preventative Maintenance	Upgrades, Bug-Fixes & Patches	Data Maintenance	Backups & Restoration	Archiving & Retention	Business Continuity Planning	Security	Contracts & Service Level Agreements	User Procedures	Periodic Review & Revalidation	Retirement	Replacement	Decommissioning	Electronic Records & Signatures
FDA Blue Book for Drug Manufacturing	1983	Computerized Systems	–	X	–	–	–	–	–	X	X	–	–	X	X	–	–	–	X	X	X	X	–	–	X	–	–	–	X
FDA Software Development Activities	1987	Computerized Systems	–	X	X	–	X	–	X	X	X	X	X	X	–	X	X	–	–	–	–	–	–	–	–	–	–	–	–
Australian TGA GMP for Therapeutics Section 900	1990	Computerized Systems	X	X	X	X	–	–	X	X	X	X	X	X	–	–	X	–	–	–	X	X	–	–	X		X	–	X
EU GMP Guide for Medicinal Products Annex 11	1992	Computerized Systems	X	–	X	–	–	–	X	X	X	X	X	X	–	–	X	–	X	–	X	X	X	–	–	–	X	–	X
Japanese MHLW Drug Manufacturing Guidance	1993	Computerized Systems	X	X	X	–	X	X	X	X	X	–	X	X	–	X	–	–	X	X	X	X	X	X	–	–	–	–	–
OECD GLP for Computerized Systems	1995	Laboratory Systems	X	–	–	–	–	X	–	–	X	–	–	X	X	X	X	–	X	X	X	X	–	X	X	–	X	–	X
FDA Computerized Systems for Food Processing	1998	Computerized Systems	X	X	X	–	X	X	X	X	X	X	X	X	X	X	X	–	–	–	X	X	–	X	X	–	–	–	X
FDA Computerized Systems Used in Clinical Trials	1999	Clinical Data Systems	X	–	–	–	–	–	–	X	X	–	X	X	–	X	X	–	X	X	X	X	–	X	X	–	X	–	X
FDA Medical Device Software Validation	2002	Medical Devices	–	X	X	X	–	X	X	X	X	X	X	X	–	X	–	–	–	X	–	–	–	–	–	–	–	–	X
PIC/S Good Practices for GxP Computerized Systems	2003	Computerized Systems	X	X	X	X	X	X	X	X	X	X	X	X	–	–	X	X	X	X	X	X	X	X	X	X	X	–	X

Note: – = not mentioned, X = mentioned by topic, with or without supplementary guidance.

TABLE 2.3
Body of Knowledge — Industry Guidance

			Supporting Processes						Project Delivery						Operation and Maintenance											Phase-Out			
Title	Date of Publication	Applies to	Training	Document Management	Change Control	Configuration Management	Self-Inspection	Managing Deviations	Project Initiation & Validation Determination	User Requirements & Supplier Selection	Design & Development	Coding, Configuration, & Build	Development Testing	User Qualification & Authorization to Use	Performance Monitoring	Repair & Preventative Maintenance	Upgrades, Bug-Fixes & Patches	Data Maintenance	Backups & Restoration	Archiving & Retention	Business Continuity Planning	Security	Contracts & Service Level Agreements	User Procedures	Periodic Review & Revalidation	Retirement	Replacement	Decommissioning	Electronic Records & Signatures
APV Guide to Annex 11	1996	Computerized Systems	X	–	X	X	–	–	X	X	X	X	X	X	X	–	–	–	X	–	X	X	–	X	–	–	X	–	X
GMA/NAMUR Control Systems Guidance	1997	PLCs and DCSs	X	–	X	–	–	–	–	X	X	X	X	X	–	X	–	–	X	–	–	X	–	–	–	–	–	–	–
ACDM/PSI Computer Validation in Clinical Research	1997	Clinical Systems	X	X	X	X	–	–	X	X	X	X	X	X	–	–	–	–	–	–	X	X	–	X	X	–	–	X	–
GAMP 4	2001	Computerized Systems	X	X	X	X	–	–	X	X	X	X	X	X	X	–	X	–	X	X	X	X	X	–	X	–	–	X	–
PDA/ISPE ERES Guide Part 2 (GAMP Guide)	2001	Computerized Systems	–	X	X	–	–	–	X	–	–	–	–	–	–	–	–	–	X	X	–	X	–	X	–	–	–	–	X
PDA/ISPE ERES Guide Part 1 (GERM)	2002	Computerized Systems	X	X	X	X	–	–	X	–	–	–	–	–	–	–	–	X	X	X	X	X	–	X	–	X	–	–	X
GAMP Calibration Guide	2002	Equipment & Instrumentation	–	–	–	–	–	–	–	–	–	–	–	X		X	–	–	–	–	–	–	–	–	–	–	–	–	–
JPMA GMP ERES Guideline	2002	Laboratory & IT Systems	–	X	X	–	–	–	X								X		X	X		X		X	–	–	–	–	X

Note: ACDI/PDI evaluation is based on generic guidance provided and excludes case study examples. – = not mentioned; X = mentioned by topic, with or without supplementary guidance.

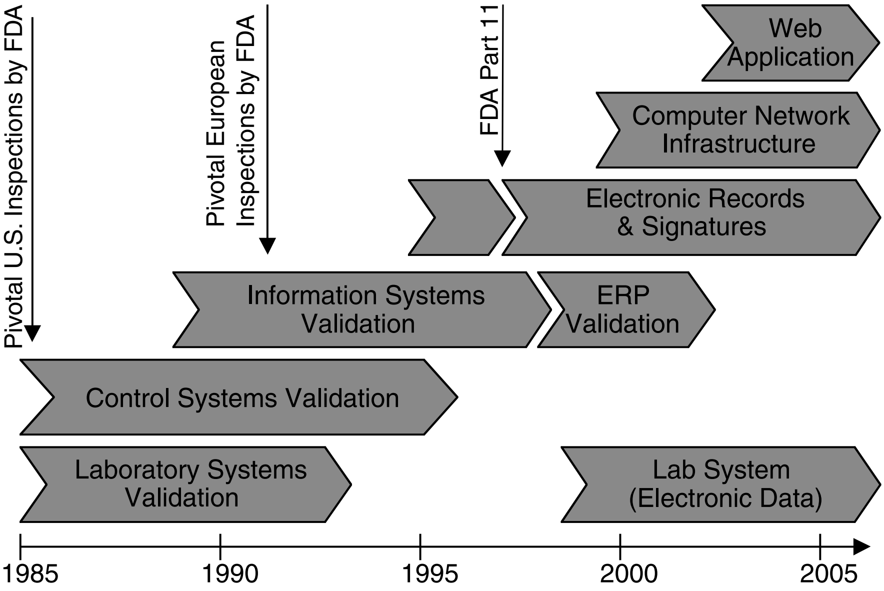

Pivotal U.S. Inspections by FDA
Pivotal European Inspections by FDA
FDA Part 11
Web Application
Computer Network Infrastructure
Electronic Records & Signatures
Information Systems Validation
ERP Validation
Control Systems Validation
Laboratory Systems Validation
Lab System (Electronic Data)
1985
1990
1995
2000
2005

FIGURE 2.4 Topical Issues Timeline.

systems affected were sterilizers and laboratory computer systems. These computer validation incidents were the first to be widely publicized in Europe; the weaknesses identified included the following:[49]

- No formal documents defining the validation requirements for computer systems.
- Specifications to define the intended software operation were not available.
- No qualification protocols defining testing or acceptance criteria were apparent.
- Accuracy checks, input-output checks, and alarm testing measures were inadequate.
- Change and version control measures for software were inadequate.
- No final review of the evidence demonstrating validation had occurred.

It was thus not possible to demonstrate that the particular drug products under manufacture with the aid of these systems were achieving their respective quality, safety, and efficacy requirements.

Later, in 1995, other very prominent and public instances of GxP noncompliance affecting computer systems occurred. These involved a water purification plant, an integrated materials control and electronic batch record system, and a tabletting control system. These were owned and operated at the Burroughs-Wellcome plant in the U.S. and the Ciba-Geigy site in Switzerland. Defects revealed by the FDA inspection included the following:

- Technical documents lacked formulae and method details.
- Design documents for systems had not been maintained in line with plant changes.
- Test parameters had been altered without documented authorization.
- Test procedures had not been kept current with specifications.
- Imprecise specification to vendor had resulted in a weak, vulnerable system design.

The lesson to the pharmaceutical and healthcare industries then was that the problems identified in the high-profile incidents cited earlier were continuing to occur, and that improvements were urgently needed over the entire industry as a whole.

The first significant adverse inspection finding for an MRP II system was published by the FDA at the end of 1997. It concerned GlaxoWellcome's retrospective validation of that system many years earlier. The defects identified were:[68]

- No original planning documents
- No systems overview documents
- No structural and functional designs
- Only a small fraction of bespoke software had undergone a detailed review
- Poor programming practice
- Inconsistent reviews of documentation had taken place
- Software version control had been inadequate

The company then decided that one of the possible options — the retrospective validation of the MRP II system — was not feasible. The company took an alternative course and inaugurated a system replacement project, coupled with interim manual procedures to take control of the functionality being provided by the MRP II system. The cost and disruption to the company's operations was huge. It painfully illustrated the impact of the noncompliance on the MRP II system. Many other pharmaceutical companies have had their MRP II computer validation censured during inspections, with common weaknesses being cited that include a lack of user authorizations, poor or absent security management, training, networks, and infrastructure support.

During the years 1997/1998 several inspectors expressed a wish to extend the scope of their regulatory inspections in the future to examine IT Departments and Data Centers that are supporting IT implementations, looking at their operation and maintenance.

In the approach to Year 2000, regulatory authorities tended to survey and monitor the preparations being made by pharmaceutical and healthcare companies to cope with the feared millennium bug rather than conduct detailed computer validation inspections. This factor seems to have delayed comprehensive inspections of computer validation. A few sporadic citations for the noncompliance of isolated computer systems arose in 1999, but there were no prominent censures in this area.

Since the start of the new millennium, regulatory authorities have returned to conducting more comprehensive examinations of computer systems. The most significant inspections are typically focused in MRP II or LIMS network applications and include supporting computer network infrastructure. Recent inspections at Eli Lilly, Argus Pharmaceuticals, and Solvay have all taken this approach. The expectations on networks are basic but have often been unsatisfied:

- Network topology diagrams
- Server and router specifications including configuration details
- Network qualification (generally not enough detail)
- Access security across network
- Backup and recovery of data across the network
- Data Center disaster recovery

The FDA has been interested in Web enablement of computer applications. Several inspections questioned client controls for intranet and Internet access. The FDA also appears to be looking more critically at distribution systems for pharmaceutical and healthcare products. The MHRA has noted that validation requirements for distribution systems are generally not well understood and consequently the validation of these systems is often deficient. Distribution systems are vital components of the supply chain and companies should address their validation accordingly. In particular, focus should be placed on validating product return and recall processes.

A recent FDA inspection of the Swedish manufacturer Pharmacia (in Sweden) over a 4-week period in the summer of 2000 examined:

- Manufacturing Resource Planning System (MRP II)
- Warehouse Management System
- Materials Management System
- Local and Wide Area Networks
- Production Control Systems
- Environment Monitoring and Control Systems
- Laboratory Information Management System (LIMS)

The defects highlighted in subsequent Warning Letters sent to Pharmacia involved:

- System descriptions
- System designs (functional and structural)
- Functional testing and qualification
- Inventory of items (hardware and software) constituting computer system
- Configuration management including version control
- Change control
- Document control
- Traceability through validation documentation
- Retention of electronic data
- Appropriate management of retrospective validation

Pharmacia avoided any consequential curtailment to its manufacturing capability through being inhibited from using noncompliant computer systems by implementing interim remedial measures while the unsatisfactory systems were upgraded or replaced. GlaxoWellcome had adopted the same approach 3 years earlier when faced with a similar situation. Large-scale upgrade or replacement programs may take more than a year to complete, often because the sheer physical size of a project and the associated skills shortages that continue to beset industry generally limit the rate of implementation. The use of interim measures to facilitate the continued use of noncompliant computer systems during this transition period has proved to be a broadly acceptable way of coping with noncompliance.

The American Red Cross also had significant issues raised during an inspection in 2002 of its blood processing computer systems.

There seems to be a growing intent to understand computer system dependencies on GxP processes, the governance of compliance across multiple sites, and a desire to inspect central development and support groups within pharmaceutical and healthcare company organizations. Many regulators are clearly frustrated with the lack of progress demonstrated by industry in terms of commitment to achieving and maintaining computer validation. Even with the emerging more pragmatic regulatory expectations concerning electronic record/signature compliance, the whole subject of computer validation seems set to remain a hot topic.

REFERENCES

1. U.S. Code of Federal Regulations Title 21: Part 211, *Current Good Manufacturing Practice for Finished Pharmaceuticals*.
2. Fry, C.B. (1982), What We See That Makes Us Nervous, Guest Editorial, *Pharmaceutical Engineering* (May): 10–11.
3. FDA (1983), *Guide to Inspection of Computerised Systems in Drug Processing*, Technical Report, Reference Materials and Training Aids for Investigators, Food and Drug Administration, Rockville, MD.

4. PMA (1986), Validation Concepts for Computer Systems used in the Manufacture of Drug Products, In *PMA Proceedings: Concepts and Principles for the Validation of Computer Systems Used in the Manufacture and Control of Drug Products*, *Pharmaceutical Technology* (May): 24–34.
5. Chapman, K. (1991), A History of Validation in the United States: Parts 1 and 2 — Validation of Computer-Related Systems, *Pharmaceutical Technology.*
6. Compliance Policy Guides, Computerised Drug Processing, 7132a: *Input/Output Checking* (Guide 07, 1982); *Identification of 'Persons' on Batch Production and Control Records* (Guide 08, 1982); *CGMP Applicability to Hardware and Software* (Guide 11, 1984); *Vendor Responsibility* (Guide 12, 1985); *Source Code for Process Control Application Programs* (Guide 15, 1987), Food and Drug Administration, Rockville, MD.
7. Clark, S.A. (1988), Computer Systems Validation: An Investigator's View, *Pharmaceutical Technology*, 12 (1): 60–66.
8. FDA (1987), *Software Development Activities*, Technical Report, Reference Materials and Training Aids for Investigators, Food and Drug Administration, Rockville, MD.
9. TGA (1990), *Australian Code of Good Manufacturing for Therapeutic Goods*, Medicinal Products — Part 1, Therapeutic Goods Administration, Wooden, Australia.
10. European Union Guide to Directive 91/356/EEC (1991), *European Commission Directive Laying Down the Principles of Good Manufacturing Practice for Medicinal Products for Human Use*.
11. PIC (1992), *Guide to Good Manufacturing Practice for Pharmaceutical Products*, Convention for the Mutual Recognition of Inspection in Respect of the Manufacture of Pharmaceutical Products (PIC), Document PH 5/92, Pharmaceutical Inspection Convention.
12. European Union, *Annex 15 — Qualification and Validation*, European Union Guide to Directive 91/356/EEC.
13. Japanese Ministry of Health and Welfare (1993), *Guideline on Control of Computerised Systems in Drug Manufacturing*, Manual for Control of Computerised Systems in GMP, Audit Manual for Manufacturers of Pharmaceutical Product with Computer Systems.
14. FDA (1995), *Glossary of Computerized System and Software Development Terminology*, August.
15. FDA (1998), *Guideline to Inspections of Computerized Systems Used in Food Processing Industry*, October.
16. International Conference on Harmonization (2000), *Good Manufacturing Practice Guide for Active Pharmaceutical Ingredients*, ICH Harmonised Tripartite Guideline, November.
17. U.S. Code of Federal Regulations Title 21: Part 205, *Guidance for State Licensing of Wholesale Prescription Drug Distributors*.
18. EU Directive 92/25/EEC on Wholesale Distribution of Medicinal Products for Human Use.
19. EU Directive 2001/82/EC, Community Code Relating to Medicinal Products for Veterinary Use.
20. EU Directive 2001/83/EC, Community Code Relating to Medicinal Products for Human Use.
21. U.S. Code of Federal Regulations Title 21: Part 58, *Good Laboratory Practice for Nonclinical Laboratory Studies*.
22. Koseisho (1988), *Good Laboratory Practice Attachment: GLP Inspection of Computerised Systems*, Pharmaceutical Affairs Bureau, Japanese Ministry of Health and Welfare, Tokyo.
23. U.K. Department of Health and Social Security (1989), *The Application of GLP Principles to Computer Systems*, GLP Monitoring Unit, United Kingdom Compliance Programme, London.
24. OECD (1995), *The Application of the Principles of GLP to Computerised Systems*, No. 10 OECD Series on Principles of Good Laboratory Practice and Compliance Monitoring, GLP Consensus Document, Environmental Monograph No. 116, Organisation for Economic Co-operation and Development Environmental Directorate, Paris.
25. EPA (1995), *Good Automated Laboratory Practices: Principles and Guidance to Regulations for Ensuring Data Integrity in Automated Laboratory Operations with Implementation Guide*, U.S. Environmental Protection Agency, Research Triangle Park, NC.
26. EU (1991), Good Clinical Practice for Trials of Medicinal Products.
27. ICH (1996), *Guideline for Good Clinical Practice*, ICH Harmonised Tripartite Guideline, International Conference on Harmonization of Technical Requirements for Registration of Pharmaceuticals for Human Use.
28. FDA (1999), *Computerized Systems Used in Clinical Trials*, Guidance for Industry, Food and Drug Administration, Rockville, MD.

29. FDA (1991), *Medical Device Good Manufacturing Practices Manual*, 5th Edition, ISBN 0-16-035844-2, U.S. Government Printing Office, Washington, D.C.
30. FDA (2002), *General Principles for Software Validation (Medical Devices)*, Final Guidance for Industry, January.
31. EU Directive 93/42/EEC, *Medical Devices*.
32. U.K. Medical Device Agency (1994), Standards, Bulletin No. 13, November.
33. FDA (1997), *Electronic Signatures and Electronic Records*, Code of Federal Regulation Title 21: Part 11, Food and Drug Administration, Rockville, MD.
34. Compliance Policy Guide (1999), Enforcement Policy, 7153.17: *21 CFR Part 11 Electronic Records, Electronic Signatures*, Food and Drug Administration, Rockville, MD.
35. PhRMA (2001), Proposed FDA Guidance on the Scope and Implementation of 21 CFR Part 11, Letter to FDA dated October 29.
36. *The Gold Sheet* (2002), December, F-D-C Reports, Inc., Boulder, CO.
37. ISPE (2002), Risk-Based Approach to 21 CFR Part 11, White Paper, December, published by ISPE (www.ispe.org).
38. FDA (2003), Part 11 Electronic Records, Electronic Signatures — Scope and Application, Guidance for Industry (www.fda.gov).
39. Japanese Ministry of Health and Welfare (1997), *Regard to Retention of Records by Using Magnetic Media*, Concerning Manufacturing Control and Quality Control of Pharmaceutical Products and Medical Devices, Open Letter to Every Prefectural Health Lead Officer, Inspection & Guidance Division of Pharmaceutical and Medical Safety Bureau, September 18.
40. Directive 1999/93/EC of the European Parliament and of the Council of December 13, 1999 on a *Community Framework for Electronic Signatures*, Official Journal of the European Communities, January 19, 2000.
41. BS 7799 (1999), Part 1: Code of Practice for Information Security Practice; Part 2: Specification for Information Security Management Systems, British Standards Institute, London.
42. *Code of Practice Concerning the Legal Admissibility and Evidential Weight of Information Stored Electronically*, PD 0008 BSI-DISC, London.
43. Japanese Pharmaceutical Manufacturers Association (2002), *Guideline for the Application of ERES in Production Control and Quality Control for Human Drug Manufacturing*.
44. Pharmaceutical Inspection Co-operation Scheme (2003), *Good Practices for Computerised Systems in Regulated GxP Environments*, Pharmaceutical Inspection Convention, PI 011-1, Geneva, August.
45. FDA (2002), Agency Information Collection Activities; Submission for Office of Management and Budget Review; Comments Request; CGMP Regulations for Finished Pharmaceuticals, Docket No. 02N-0007.
46. Tetzlaff, R.F. (1992), GxP Documentation Requirements for Automated Systems: Parts 1, 2, and 3, *Pharmaceutical Technology*, 16 (3): 112–124, 16 (4): 60–72, 16 (5): 70–82.
47. Trill, A.J. (1993), Computerized Systems and GxP — A UK Perspective: Part 1: Background, Standards and Methods; Part 2: Inspection Findings; Part 3: Best Practices and Topical Issues, *Pharmaceutical Technology International*, 5 (2): 12–26, 5 (3): 49–63, 5 (5): 17–30.
48. International Society for Pharmaceutical Engineering (1994), An Interview with Richard Klug, *Pharmaceutical Engineering*, 14 (3): 26–31.
49. Pharmaceutical Manufacturers Association (1986), *Validation Concepts for Computer Systems Used in Manufacture of Drug Products*, PMA Proceedings: Concepts and Principles for the Validation of Computer Systems used in the Manufacture and Control of Drug Products, April 20–23, Chicago.
50. U.K. GAMP Forum (1996), *Supplier Guide for Validation of Computer Systems in Pharmaceutical Manufacture*, Second Version, International Society for Pharmaceutical Engineers, The Hague.
51. PDA (1995), Validation of Computer Related Systems, *PDA Journal of Pharmaceutical Science and Technology*, Technical Report No. 18.
52. Grigonis, G.J., Subak, E.J., and Wyrick, M.L. (1997), *Validation Key Practices for Computer Systems Used in Regulated Operations*, *Pharmaceutical Technology*, 21(6).
53. U.K. GAMP Forum, *Supplier Guide for Validation of Computer Systems in Pharmaceutical Manufacture*, Third Edition — GAMP 3, 1998, International Society for Pharmaceutical Engineers, The Hague.

54. Supplier Forum (2000), *Guidance Notes on Supplier Audits Conducted by Customers*, available through GAMP Forum (www.ispe.org).
55. Supplier Forum (2000), *User Requirements Specifications — Guidance Notes on the Preparation*, available through GAMP Forum (www.ispe.org).
56. Supplier Forum (2000), *Guidance Notes on the Preparation and Use of Test Specifications and Test Documentation*, available through GAMP Forum (www.ispe.org).
57. PDA (1999), A Globally Harmonised Glossary of Terms for Communicating Computer Validation Key Practices, *PDA Journal of Pharmaceutical Science and Technology*, 53 (2), March/April.
58. GAMP Forum (1999), Compliance for IT Infrastructure, *Pharmaceutical Engineering*, 19(6).
59. Crosson, J.E., Campbell, M.W., and Noonan, T. (1999), Network Management in an FDA-Regulated Environment, *PDA Journal of Pharmaceutical Science and Technology,* 53 (6), November/December.
60. PDA (1999), Validation and Qualification of Computer Laboratory Data Acquisition Systems, Technical Report 31, *PDA Journal of Pharmaceutical Science and Technology*, 53 (4), July/August.
61. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (www.ispe.org).
62. GAMP Forum (1999), *Complying with 21 CFR Part 11 Electronic Records and Electronic Signatures*, First Draft: Consultative Document to Solicit Feedback, December.
63. FDA, Compliance Policy Guide, Computerized Drug Processing, 7153: *Enforcement Policy: 21 CFR Part 11; Electronic Records; Electronic Signatures* (Guide 17, 1999), Food and Drug Administration, Rockville, MD.
64. GAMP Forum (2001): *Good Practice and Compliance for Electronic Records and Signatures: Part 2 — Complying with 21 CFR Part 11*, Electronic Records and Electronic Signatures, published by ISPE and PDA (www.ispe.org).
65. PDA (2002): *Good Practice and Compliance for Electronic Records and Signatures: Part 1 — Good Electronic Record Management (GERM)*, published by ISPE and PDA (www.pda.org).
66. *The Gold Sheet* (1995), January, Boulder, MD: F-D-C Reports, Inc.
67. Branning, R.C. (1986), Experience in an FDA Inspection, *PMA Proceedings: Concepts and Principles for the Validation of Computer Systems Used in the Manufacture and Control of Drug Products*, Pharmaceutical Manufacturers Association.
68. Wingate, G.A.S. (2000), *Corporate Computer Systems Validation: Good IT Practice for the Pharmaceutical Industry*, Interpharm Press, Buffalo Grove, IL.

3 Organization and Management

CONTENTS

Pharmaceutical and healthcare companies still have substantial concerns over the most appropriate way to address the organization of validation and how to enable management to fulfill its responsibilities. This chapter suggests a typical approach that satisfies a company's accountabilities. However, in presenting this offering we do not purport to suggest that this is the only acceptable approach. The needs of pharmaceutical and healthcare companies will vary since they depend on many different factors, including the scope of the validation requirement, the availability of suitably skilled corporate and contract staff, and cost.

ORGANIZATIONAL RESPONSIBILITIES

Pharmaceutical and healthcare companies should appoint a senior management representative with specific responsibility for ensuring that the requirements of GxP are implemented and maintained. This individual, who often takes the job title of Computer Validation Director, must wholeheartedly champion the cause of GxP. The authority and responsibility of this senior position should be clearly defined and recorded.

It is very important that this senior manager has the authority to block the release of drug products on the grounds of noncompliant validation since this can compromise the quality of drug and healthcare products. Without such a level of authority, the individual will have to rely solely on his or her powers of personal persuasion with others who may, themselves, be under acute production or sales pressures to permit the release of the drug product. Bitter experience shows that this persuasion will seldom be enough. As a result, it has long been taken for granted in the industry that quality managers must have the authority to place an embargo on drug products that they deem to be substandard. The senior manager responsible for GxP must possess a similar level of empowerment; many companies achieve this by placing their GxP personnel within their Quality Control/Assurance management hierarchy.

The senior manager responsible for validation is expected to recruit appropriately qualified and experienced staff to conduct the validation tasks and ensure that these are properly and effectively carried out.[1] Specific responsibility for validating computer systems should be assigned to a manager who is suitably qualified — someone with relevant computer systems experience and appropriate training.

In many organizations the senior manager responsible for GxP may also have other duties. The company must formally acknowledge and concede that these other duties do not excuse or relieve his or her responsibility for validation to the GxP regulatory authorities. In the phrase forever associated with President Harry S. Truman, *the buck stops here!*

The expected role of senior managers is defined by the ISO 9001 standard, which states that management shall

> *… define and document [the company's] policy and objectives for, and commitment to, quality and ensure that this policy is understood, implemented and maintained at all levels in the organisation.*[2]

While this is useful, more practical guidance is available. An interpretation of the ISO requirements is presented below based on work by Teri Stokes:[3]

1. Establish a Validation Working Party (work group) to define company validation policy, including a statement of commitment to such policies and objectives, and any associated company plans.
2. Establish a Company Validation Committee to set the company computer validation strategy, approve the validation policy, provide oversight, and agree on funding models.
3. Establish Site Validation Steering Committees to prepare an inventory of systems, set priorities, establish site validation master plans, approve validation procedures, assign resources, and monitor progress.

4. Develop an awareness/education program for delivering this document to all senior managers and to their employees.
5. Monitor progress, priorities, resources, and funding against company objectives and strategy.

Company computer validation steering committees should be multidisciplinary teams with representatives from research, production, engineering, quality control/assurance, and business support. Site validation committees should also be multidisciplined with representatives from technical operations, IT, laboratory management, engineering, operational quality, and QA validation. Members of both working groups and validation committees should be trained via internal or external courses so that they gain an understanding of the basic principles of validation. Alternatively, members might attend a validation conference where they could also meet validation practitioners from other companies. Major conferences are regularly organized by International Society of Pharmaceutical Engineering (ISPE), Parenteral Drug Association (PDA), and Institute of Validation Technology (IVT) throughout Europe and the U.S. External consultants with specialist validation knowledge and experience may also be engaged to support the working group and validation committee, perhaps assisting with internal training courses.

Recent Inspection Findings

- Inadequate organizational structure to ensure quality system requirements met. [FDA 483, 2002]
- Failure to have a Quality Control Unit adequate to perform its functions and responsibilities, as required by 21 CFR 211.22, as demonstrated by the number and type of inspectional observations. [FDA Warning Letter, 2002]
- Appointment of management representatives was not documented. [FDA Warning Letter, 2001]
- Management personnel did not know whether some products handled in the facility were regulated by the U.S. Food and Drug Administration or not. [FDA Warning Letter, 2001]

COMPLIANCE STRATEGY

From the outset the aim must be to make validation as cost-effective as practicable. Several pharmaceutical companies who rushed into the validation of computer systems in the late 1980s and early 1990s discovered to their own cost that inefficient validation programs are hugely expensive, involving much more work than is really necessary.

Figure 3.1 illustrates three basic compliance strategies by comparing the cost associated with compliance (prospective validation) against the cost associated with noncompliance (the combined impact of retrospective validation and business disruption).

Point A in the graph denotes the break-even point where the cost of noncompliance equals the cost of compliance. This may appear to indicate the ideal amount of validation effort, an effort that just delivers compliance but constrains cost by going no further. Is this really the ideal that should be aimed at? Validation requirements enforced by the various regulatory authorities are interpretative, not prescriptive. Further, regulatory inspections are never exhaustive; practical limitations of time and resources mean that inspections can only examine a part of a company's operation. Therefore, they cannot ensure the exposure of all noncompliances. Assessing where point A really is thus inspired guesswork, something of an art rather than a science. Aiming at point A but missing it will mean either that compliance is not achieved or that money has been wasted. An alternative strategy that is often adopted after a serious noncompliance has been revealed by a regulatory authority is to aim for point C. This point represents an exhaustive validation effort in a climate of zero tolerance

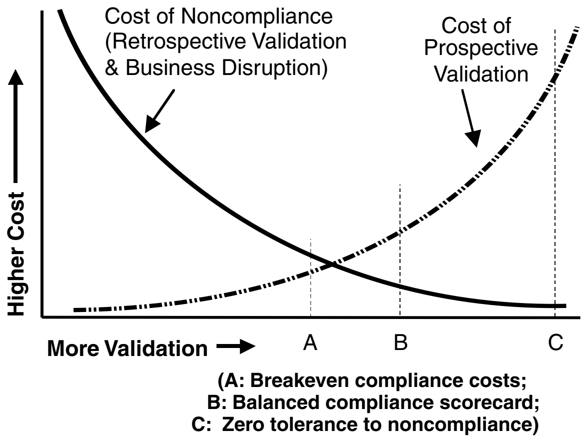


FIGURE 3.1 Compliance Strategy. A: Breakeven Compliance Costs; B: Balanced Compliance Scorecard; C: Zero Tolerance to Noncompliance.

of any regulatory criticism, however minor. The cost that this point implies is exorbitant and adds little value to the business or indeed to the likelihood of regulatory compliance — most of it is unnecessary overhead. It is useful, then, to consider a third compliance strategy, the *middle way*, whereby a more balanced approach to compliance is adopted. It has been claimed that this type of approach can lead to 40 to 50% cost savings when compared to those incurred at point C, while still maintaining a sustainable level of regulatory compliance.[4] Point B can be viewed as a *common-sense approach*, erring on the side of caution by being more conservative than that represented by point A. There is a broad consensus of approval within the industry over the wisdom of setting validation effort at this point. Its precise location can only be determined by surveying industry practice and monitoring regulatory expectations. Benchmark exercises are not readily available, so a more *ad hoc* collation of information derived from consultants, new recruits, industry associations, and informal regulatory contacts is often used. As long as the quality of information is adequate, not tainted by hidden political agendas, it should be possible to arrive at point B relatively easily.

Members of the working group establishing a compliance strategy must appreciate the implications of their policy on working practices. For instance, point B should not be determined solely on the basis of the cost of validation but rather on its effectiveness, by examining the standards and practices to be employed. Inefficient validation practice can inflate project overhead costs by up to 30%. This could entirely undermine the potential cost savings associated with adopting a compliance strategy based on point B.

Organizational Considerations

Computer validation should not be undertaken unless fundamental validation controls have been fully understood and implemented within the pharmaceutical or healthcare company's organization. Here we allude to properly qualified personnel, effective document management and change control systems, internal audit procedures, methods of managing the deviations from standard practice thereby exposed, and a culture of continuous improvement (see Chapter 4 for more details). Senior management must not fall into the trap of assuming through complacency or idleness that these controls have been fully instituted! In most firms there is usually much that still needs to be done in these areas. Let us examine these controls a little more closely.

Personnel

A critical factor in the successful development of a sustainable validation capability is knowledge retention. Many pharmaceutical and healthcare companies are highly dependent on contract resources. Contractors must be trained just like permanent personnel. However, the problem here is that by the very nature of their employment they are temporary employees. Many pharmaceutical and healthcare companies are developing knowledge retention tools, but this does not avoid the need to carefully address the development and retention of a critical mass of permanent staff. It is only in this way that the continuity of the organization's validation capability can be secured.

Document Management

Regulators are dependent on documentation for evidence of validation, but it is surprising, for example, how often documents are temporarily mislaid or permanently lost. Poor document management will nearly always attract regulatory criticism. Signatures applied to documents should be appended in a timely fashion. Signatures should never be applied retrospectively. Such an act is fraudulent, however good the intention, since the implication of a signature is that it has been applied at the right time unless specifically stated otherwise. In the same way, altering pages of an approved document without seeking a renewal of the approval is deceitful. Staff can find themselves under enormous pressure to infringe these rules in the white heat of fierce production demands. Pharmaceutical and healthcare companies must ensure that staff are protected by outlawing such practices, and invoke professional misconduct or disciplinary procedures with potential dismissal as the outcome should they discover such activities. There are two other issues that often arise. The first is whether superseded documents should be archived. The second is whether review comments from different individuals should be retained for each document issued, even though not all the suggested comments have been accepted and included. The firm must have procedures or standards to clarify these questions.

Change Management

Formal change control is essential for data, documents, hardware, and software. The principles involved are not complex and can be readily implemented. However, organizations often fall into the trap of implementing needlessly diverse change control systems. This usually happens when different functions or projects claim to have unique requirements, or when the existing change management process does not fit organizational structure and role responsibilities. On occasion these arguments may be quite valid, but they need to be critically explored and challenged. A multiplicity of change control systems contributes complexity within the scope of the project and to the interfaces between individual systems. Consequently, users become confused over which system is appropriate to use to manage their change. Such confusion is dispiriting. Something that is even worse is that hidden gaps between change control systems may develop, leading to systemic noncompliance. The lesson then is clear; minimize the number of change control systems in use and be vigilant.

Self-Inspections, Managing Deviations and Continuous Improvement

It is a basic expectation of the regulators and the regulations that a program of self-inspections (internal audits) with effective follow-up action has been established. Unfortunately, too often the organizations and projects that need this most claim that they do not have enough time to conduct such self-inspections. Without these, an organization has no sure means to identify noncompliances early and correct them. Managing deviations rather than eliminating their causes becomes the norm rather than the exception. Such organizations often complain that validation does not add value and begin to treat it negatively as an overhead. On the contrary, within a culture of continuous improvement, advances to make validation more efficient and effective are constantly being sought. Such advances should be managed from a continuity perspective. Discontinuity is often associated with a high compliance risk.

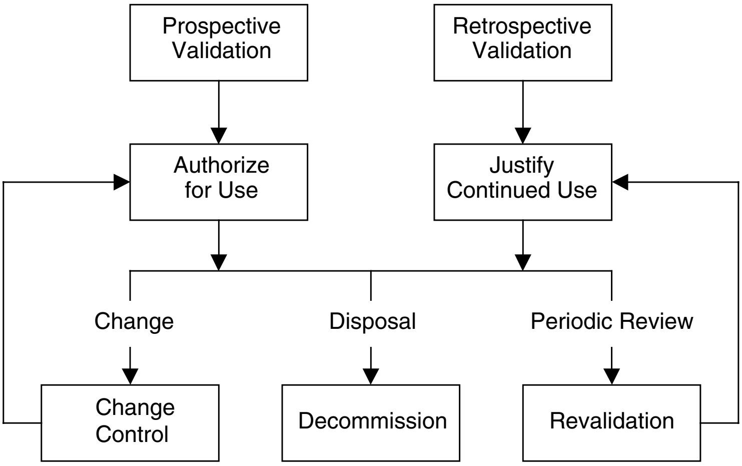


FIGURE 3.2 Relationship between Prospective and Retrospective Validation Management.

VALIDATION POLICY

Validation Policies vary greatly between different companies. There are no set rules governing their content or structure. A Validation Policy should define the corporate intent toward GxP and may refer to a specific policy for validating computer system systems. It is suggested here that the policy should cover the following:

- A definition of the overall principle of validation
- The scope of its application
- A statement of commitment
- An outline of how validation will be achieved and maintained
- A definition of who is to be responsible
- A glossary defining the terminology to be used

Both prospective and retrospective validation must be considered for new and existing computer systems, respectively. Figure 3.2 outlines the relationship between prospective and retrospective validation. New systems must be authorized for use, while the continuing use of existing systems must be justified. Once validation is achieved, it must be maintained despite any changes that are made to it. All such changes must be scrutinized through change control and the revised system specifically authorized for use. Periodic reviews must also be performed to ascertain whether an overall revalidation is required, either as a result of cumulative changes over time, regulatory developments, or due to organizational changes that relate to validation practice. If the use of a computer system cannot be justified, it should be decommissioned (a premature retirement). A computer system will require decommissioning at some time anyway, once it reaches the end of its useful life.

Key principles for computer systems validation that might be included in the Validation Policy based on the GAMP 4 Guide[5] are presented in Appendix 3A. These principles have been prepared to address the basic requirements of the following regulatory authorities:

- U.S. Food and Drug Administration (FDA)
- U.K. Medicines and Healthcare products Regulatory Agency (MHRA)
- Japanese Ministry of Health, Labor, and Welfare (MHLW)
- Australian Therapeutics Goods Administration (TGA)

Experience has shown that a clear, concise computer validation policy can be achieved in a 5- to 10-page document and produced in a couple of man-months. Typically, most of the effort is spent in consultative meetings to secure a consensus on the content of the policy that should, in any event, confine itself to high-level statements of principle. It should by its nature be relatively stable. Nevertheless, it should be periodically reviewed and kept up to date.

Recent Inspection Findings

- No evidence that the Quality Policy has been implemented, understood, and maintained by all levels of the organization. [FDA Warning Letter, 2001]

VALIDATION PROCEDURES

Standard operational procedures (SOPs) should be drafted by experienced validation practitioners who have experience in developing such procedures. The number of SOPs that are required will depend on organizational complexity and the magnitude of the systems being validated. It may be necessary to hire an external consultant to fulfill this role. He or she should join a team of end users as a ghostwriter to aid the development of the validation procedures. The aim here is not to impose a set of generic validation procedures that might be in some way foreign to the organization. Rather, the goal is to tailor the end user's current working practices into compliant validation procedures with a minimum of change. The end user's personal involvement should ensure that they and their colleagues will readily adopt the procedures without resentment.

About 20 to 25 generic procedures will be needed to cover the validation life cycle for a computer system. The following list is based on GAMP 4:[5]

Management

- Validation Planning
- Supplier Audit
- Risk Management
- Design Review and Traceability Analysis
- Quality Planning
- Validation Reporting
- Change Control (Project and Operational)
- Configuration Management
- Document Management

Project

- User Requirements Specification
- Functional Specification
- Hardware Design Specification
- Software Design Specification
- Software Controls
- Testing and Qualification

Operation

- Periodic Review
- Service Level Agreements

- Security
- Performance Monitoring
- Record Retention, Archive, and Retrieval
- Backup and Recovery
- Business Continuity Planning
- Decommissioning

Managing electronic records and electronic signatures may be handled with separate SOPs or integrated into the above. An abridged set of procedures will be appropriate for small systems, but supplementary procedures will be needed for larger ones.

Management must approve procedures and any subsequent changes made to them. Approval signatures will be required from at least two individuals representing a quality and technical perspective. Management must then ensure that any deviation from these procedures is properly authorized and documented. Deviations will often be associated with corrective actions arising from internal audit findings; these must also be documented together with the evidence that demonstrates their resolution.

Individual procedures can easily consume 10 to 15 days' effort to produce, even with experienced staff. Use should be made of industry guidance when developing procedures, e.g., GAMP example procedures and IEEE standards. Where existing procedures are being revised to secure validation compliance, this estimate of effort could be reduced by about half. As recommended above, this should be supplemented with about 20 days' effort across all the procedures, shared among a team of end users. The individuals should contain the core users who are involved in all the procedures in order to ensure consistency. Other end users on the team, however, can be seconded for the development of particular procedures in which they have a specific interest, or can contribute a particular skill or competence. For instance, an end user quality representative may wish to be seconded for the development of the Supplier Audit procedure.

To make the use of these validation procedures easier, many organizations are developing document templates and tools to assist practitioners to prepare, review, and approve documents in a rapid, quality-conscious fashion.

Pharmaceutical and healthcare companies may also find it beneficial to tailor particular sets of procedures to different types of systems. This may help ownership and adoption since different types of systems are usually supported by QA/laboratory, engineering, and IT departmental functions. From a technical standpoint, too, it is very difficult to make a single set of procedures easy to use while providing a practical level of detail to address the various technical characteristics of different types of computer systems. One size does not readily fit all. As a consequence, typically there might be four sets of validation procedures:

- Laboratory applications (e.g., analytical, measurement)
- Control systems (e.g., PLC, SCADA, DCS)
- IT systems (e.g., ERP, MRP II, LIMS, EDMS)
- Computer Network Infrastructure (e.g., servers, networks, clients)

An additional set for desktop applications (e.g., spreadsheets, databases, and Web applications) may be needed, but more typically these are included within the general scope of IT systems.

There are inevitable interfaces between the application areas of the various sets of procedures and the computer systems to which they apply, as indicated in Figure 3.3. Client–server technology is typically the deciding factor in determining whether control system and laboratory application projects would be better served by IT system procedures. Another example might be that robotic systems used to automate laboratories would be better served by control system procedures.

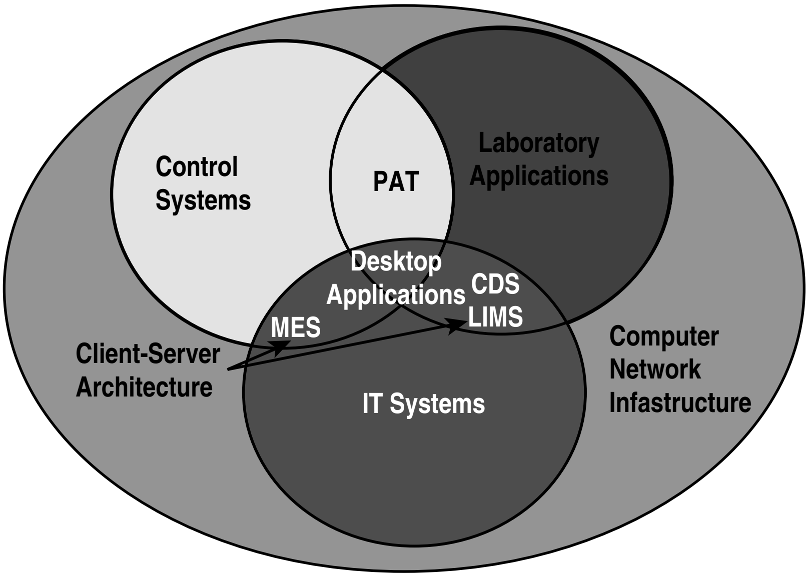


FIGURE 3.3 Mapping Procedures to Computer Systems.

Recent Inspection Findings

- The Quality Control Unit failed to ensure that adequate procedures were in place to define and control computerized production operations, failure investigations, equipment, qualifications, and laboratory operations. [FDA Warning Letter]
- Quality system procedures not implemented. [FDA 483, 2002]
- No SOPs for system validation. [FDA 483]
- There were no written standard operating procedures for, but not limited to, system validation, hardware and software change control, revalidation, … [FDA Warning Letter]

COMPUTER SYSTEMS INVENTORY

Computer systems used by pharmaceutical and healthcare companies should appear on an inventory. The determination of whether individual systems impact GxP needs to be indicated on the inventory, together with the status of its validation. The assigned priorities given to the validation of different computer systems can also be shown here. In this way the inventory can be presented to a GxP regulator as evidence of the commitment of an organization to validate its computer systems and to present an overall report on progress to date. The amount of information to be disclosed to a regulator should be carefully considered. Inventory fields will normally include:

- System Name
- System Reference
- System Description
- System Type
- Site at which the system is used (unless site-specific inventory)
- GxP Impact

The System Name field describes the common name by which the system is known within the operating company. A synonym or "also known as" column may prove useful if different groups refer to the same system under a variety of names. The System Reference field, providing a unique reference number for the computer system, can be used to identify the bounds of the system and its associated documentation set. The System Description field gives a brief overview of the system's functionality. The System Type field might group business systems, production systems, laboratory systems, and infrastructure systems from which sub-lists can be easily generated if an inspector is interested in a particular type of system. The Site field is useful when the inventory covers multiple sites. Inspectors are normally only interested in the systems used at the site they are visiting, and it is wise not to swamp the inspector with additional lists that detail computer systems outside the scope of his or her interest. The GxP Impact field will need to identify whether or not the computer system has been used to measure, monitor, control, record, or report those critical processing parameters that characterize substandard product (based on the *Baseline Pharmaceutical Engineering Guide*[6]). When determining whether or not a computer system has a GxP Impact, it is useful to reflect on the advice given by Sam Clark, a former FDA investigator: "If it looks like a duck, flies like a duck, and sounds like a duck, then it's probably a duck"! In other words, use your common sense.

Other fields that can prove useful in the inventory include a Validation Status field to indicate whether validation is complete, in progress, or yet to be started. A high-level decision tree for determining a validation requirement is shown in Figure 3.4. An ERES field could be used to monitor compliance and fulfillment of the electronic record and electronic signature regulations. Another field might be used to show the current status of validation on a project (e.g., Validation Plan or Qualification). The list might conclude with a document reference to the system's Validation Report that will give details of the completed validation. Identification of inspection frontiers, business owners, and contacts for support may also prove very useful when preparing for inspections. A field giving last or next review dates can be used to schedule periodic reviews and follow-up Supplier Audits.

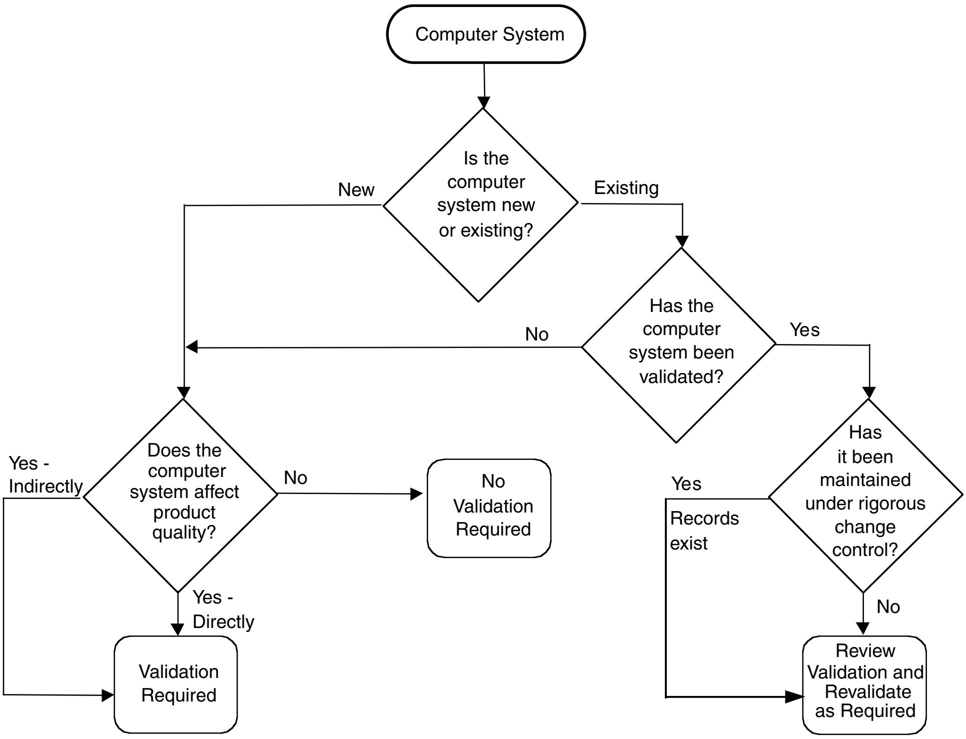

Computer System
Is the computer system new or existing?
New
Existing
Has the computer system been validated?
No
Yes
Does the computer system affect product quality?
Yes - Indirectly
No
No Validation Required
Yes - Directly
Validation Required
Has it been maintained under rigorous change control?
Yes
Records exist
No
Review Validation and Revalidate as Required

FIGURE 3.4 Determining a Validation Requirement.

It is important to consider the effort to maintain increasing numbers of fields when designing the system inventory. The greater the number of fields the more maintenance will be required and the harder it will prove to keep it up to date. Remember that there may be many thousands of computer systems in use on larger pharmaceutical and healthcare manufacturing sites.

Some firms have adapted Y2K inventories that were originally established to manage millennium compliance. If the inventory is held on a spreadsheet or a database application, then it should be validated in its own right. However the inventory, paper based or automated, is managed, it is vital that the copy that is the master is established: this typically means just a single list exists.

The inventory should be reviewed and approved by a QA representative. Reapprovals of the whole inventory will be required periodically, perhaps every 2 years. In the meantime, individual system changes that affect the degree of GxP impact of a system or its validation status should be approved by QA. Changes to information in the inventory that do not affect details about GxP systems require review and approval but not necessarily involving a QA representative.

An example inventory is shown in Table 3.1. The table can be used to generate management status data such as:

- Number and percentage of systems requiring validation
- Number and percentage of systems with validation planned but outstanding
- Number and percentage of systems with validation in progress
- Number and percentage of systems with validation complete

The regulatory agencies will be interested in the continued commitment of senior management to monitor and fund progress. Summary reports should be periodically prepared for this purpose. When doing this it is important to remember that the validation status will also have to be periodically reviewed to determine whether revalidation is required. This will be needed to address the cumulative effect of changes made to the computer system and to address any impact resulting from changed regulatory requirements.

VALIDATION MANAGEMENT

The management of validation can be considered a cyclical process, as shown in Figure 3.5. The cycle begins with GxP Assessments surveying the validation requirements of computer systems in readiness for preparing Validation (Master) Plans. Supplier Audits assess the capabilities of suppliers in providing computer systems and associated services. Validation is then conducted according to any prevailing priorities. Once validation is completed for individual computer systems, its operational compliance must be maintained.

Throughout the validation management process there should be formal opportunities to review validation practices. The status of validation work is likely to change; some existing systems may be decommissioned, new systems may be planned, and the priorities of the current inventory work may vary due to changing company needs. Validation Policy and Validation Procedures may change as a result of new regulatory requirements or feedback from project experience.

GETTING STARTED

There are two main obstacles to be surmounted here: a lack of compliance experience and an absence of focus and determination. Weak compliance experience is often characterized by validation questions like “Why are we doing this anyway?” and “What are the fundamental principles we should follow?” Education and training, and managing the learning curve are both key issues. Managers and sponsors, as well as practitioners, need a practical appreciation of validation (trends, constraints, areas of flexibility, and benefits). It must be understood that initial work adopting a validation approach will require more effort because practitioners will not be familiar with the new

TABLE 3.1
Example Inventory

System								
Name	Reference	System Description	System Type	Site Used	GxP Impact	Status	Priority	Next Review
CAPCON	NB121	Capsule Filling Weight Control (SCADA)	Production	Brighton	Yes	Validated	N/A	April 2002
ERP	CPY02	SAP R/3 MRPII system with SD, FI, CO, MM, and PP Modules	Business	Bordeau, Brighton, Darwin, Trenton	Yes	Not Validated	High	N/A
CART2	NB081	Cartonner	Production	Brighton	Yes	Validated	N/A	April 2002
N/A	B279	Product Market Costings Spreadsheeet	Production	Brighton	No	Not Validated	N/A	Jan. 2001
HPLC30	NB233	HLPC	Laboratory	Brighton	Yes	Validated	N/A	Dec. 2002
LAN1	NB351	LAN — Site Network	Infrastructure	Brighton	Yes	Not Validated	Low	N/A
N/A	NB401, B205	SOP Distribution Spreadsheet	Business	Bordeau, Brighton	Yes	Not Validated	Medium	N/A
APRAISE	NB424, B196	Statistical Package for Annual Product Reviews	Laboratory	Bordeau, Brighton	Yes	Not Validated	Medium	N/A

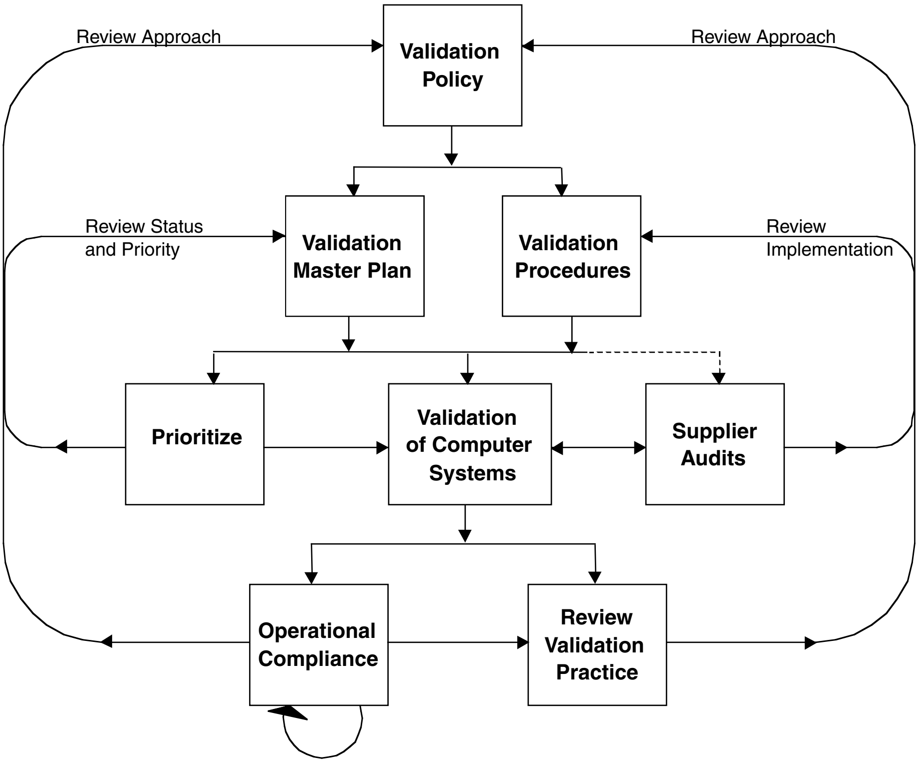


FIGURE 3.5 Validation Management.

way of working. It is wise to avoid large work packages as the starting point for adopting the new quality-assured mode of working since the scale of inefficiency might shock an organization, tempting it to abandon the quality-assured approach altogether. It is better to begin with smaller work packages and watch the cost of validation reduce as practitioners improve their understanding and efficiency. The learning curve will eventually flatten out to a plateau. However, herein lies another danger. If key staff move on and learning has not been captured in the corporate memory (policies, procedures, guidance, training, and succession planning), learning and efficiency will be lost. Even if these pitfalls are avoided, there is still a need to pace the introduction of the new way of working in case progress is not forthcoming. The best course is to build on success and extend the new ways of working based on a proven track record.

Risk Management

Risk management is very important if appropriate resources are to be deployed in a timely fashion to mitigate or reduce the potential effect of identified risks. It is recommended that a risk map be produced showing where computer systems are used to support the various process streams of operational activity.

Determining which operational aspects are most critical requires an understanding of the potential impact on drug or healthcare product safety, quality, and efficacy. The Canadian Health Products and Food Branch Inspectorate has already identified a number of high-risk issues that are likely to result in noncompliant drug product and present an immediate or latent public health risk.[7] A similar identification of high-risk issues has been proposed by the former U.K. MCA (now MHRA).[8] These high-risk issues are applied here to computer systems and aligned to the following six operational areas.

Quality Systems

- Document management
- SOP administration
- Security access controls (e.g., user profiles and password management)
- Change control records
- Customer complaints
- Adverse event reporting
- Review/audit/corrective actions management
- Training records

Facilities and Equipment Systems

- HVAC controls and alarm handling
- Critical equipment and instrumentation (calibration and maintenance)
- Change control records
- Validation records

Materials Systems

- Traceability of material handling
- Raw material inspection/testing/quarantine management
- Storage conditions
- Containers usage and cleaning management
- Distribution records and recall management

Production Systems

- Recipe/formulation management
- Batch manufacturing instruction and records
- In-process testing
- Yield calculation
- Purified water
- Aseptic filling

Packaging and Labeling Systems

- Labeling information

Laboratory Control Systems

- QC raw data
- Stability testing
- Sterility testing
- QC analytical results
- Quality disposition
- Out-of-specification investigations

Workflow analysis is an effective way of pictorially mapping risks. The example given in Figure 3.6 is very simplistic. A balance has to be struck so that the process can be mapped in a manageable number of pages. Typically, the process flowchart will include decision trees and there may be more information given on computer systems. The information for each computer system should include a determination for each system; identifying any validation requirement should be clearly marked on the risk map (see "GMP Impact" in Figure 3.6). In addition, the relative risk (e.g., high, medium, low) to the process at the points where computer systems are used should be included. Other relevant supplementary information can be added as deemed appropriate. Producing the risk map on A3 size paper can help give a better overview of larger processes.

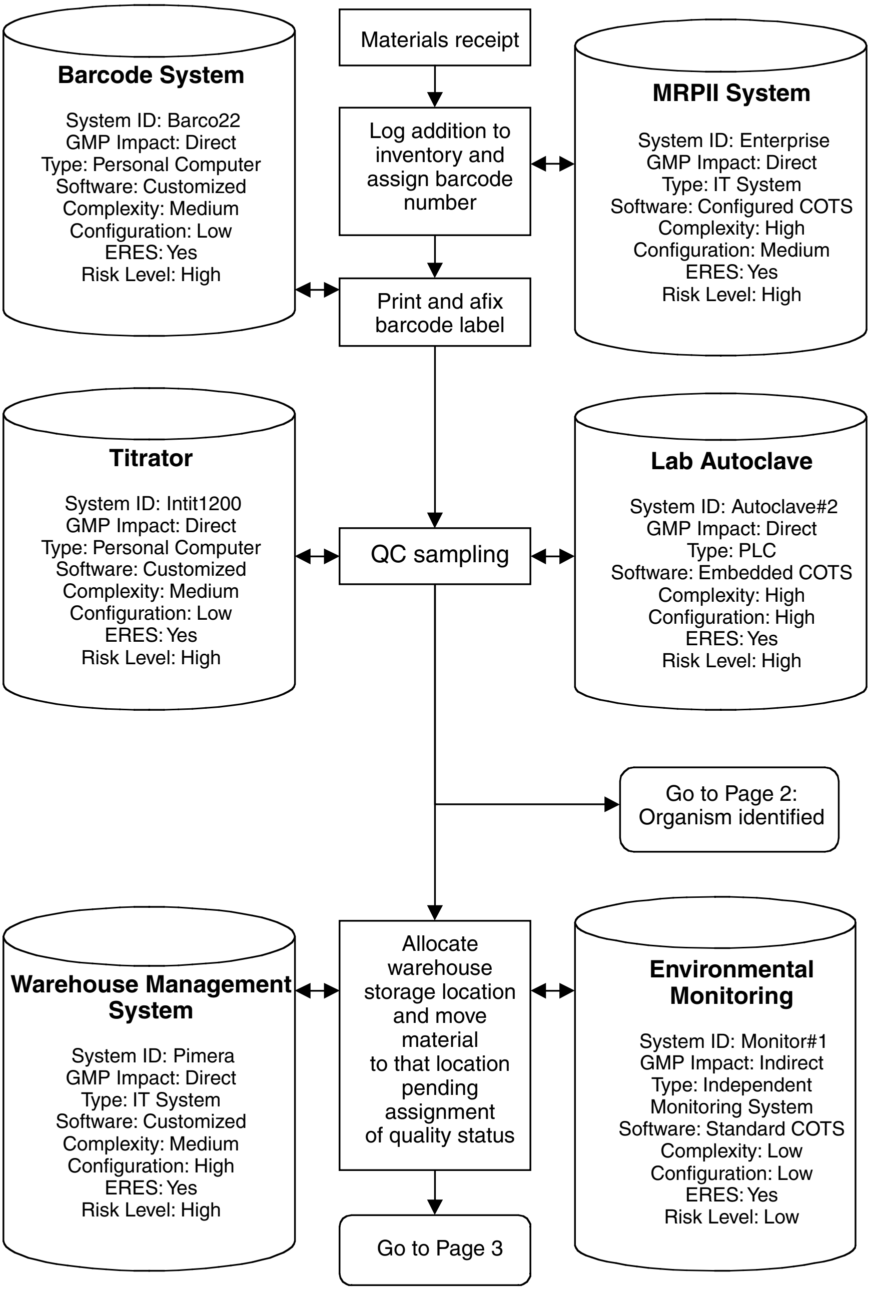


FIGURE 3.6 Example Risk Map.

The rigor of validation for computer systems supporting these critical operational aspects of the processes should take account of their composite custom (bespoke) software, COTS software, and supporting computer network infrastructure. The risk map and supporting rationales will form the basis of Validation (Master) Plans that are discussed in more detail later in this book.

Life-Cycle Approach

The life-cycle approach has attracted broad acceptance across the pharmaceutical and healthcare industries and can be refined to meet the needs of particular applications. Different organizations use variants of the life cycle, but the methodology of dividing a life cycle into phases remains the same. For instance some companies develop the subphases that are indicated in the phase descriptions above as distinct phases in their own right. The specific life-cycle model chosen does not really matter. Its constituent phases must, however, be clearly defined in advance, with entry and exit criteria for each phase and appropriate verification procedures to ensure the controlled completion of constituent phases.

Figure 3.7 presents a set of life-cycle phases that summarize the validation approach typically used within the pharmaceutical and healthcare industries. Life-cycle phases may be known by alternative names in different organizations. There is no standard glossary throughout the industry relating to naming conventions or groupings of phases. It is important, however, that all the activities covered by this chapter are included in any alternative scheme.

The validation life cycle is primarily for new computer system systems. The principles, nevertheless, also apply to older computer system systems. The life cycle is consistent with guidance provided by the Australian, European, Japanese, and U.S. GxP regulatory authorities.[9–13] It is also consistent with guidance provided by the German GMA-NAMUR, the U.S. PDA, and the U.K. GAMP Forum industry initiatives.[5,14,15]

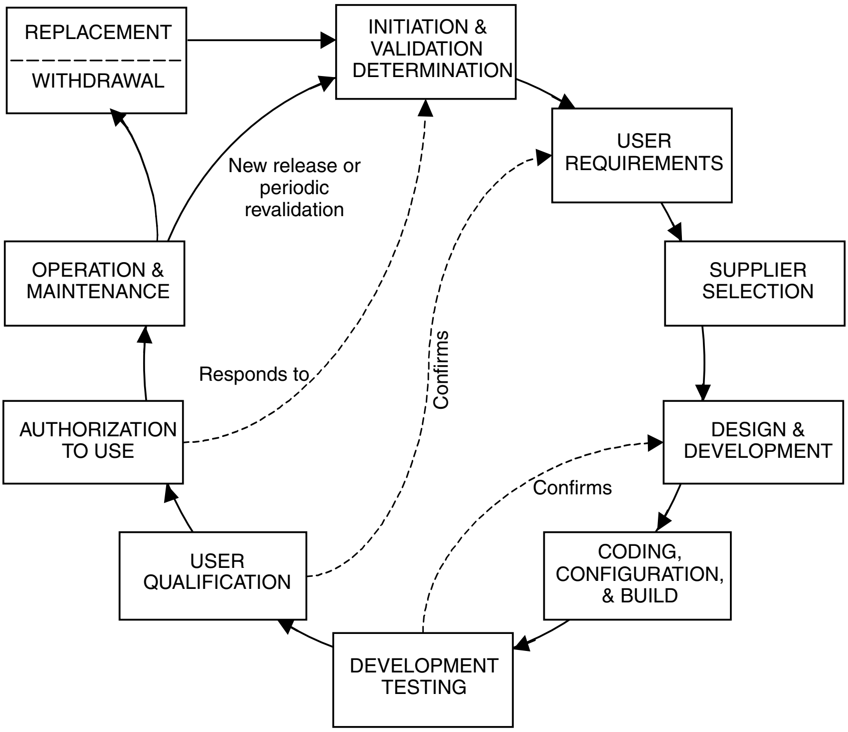


FIGURE 3.7 Validation Life Cycle.

The steps in the validation life cycle are not necessarily executed in the order indicated. Rather, the steps are usually executed as an iterative process in which various functions may be carried out concurrently. If necessary, steps may be repeated. For instance, the Validation Master Plan may be developed after, or concurrently with, the User Requirements Specification (URS) rather than before, as indicated. Equally, a Supplier Audit often involves a series of steps that may not be complete until well into the validation project.

Some of the steps in the validation life cycle will not be needed in some validation projects. For instance, the use of preferred suppliers for software and hardware products or services removes the need for repeated supplier appraisal and selection. Life-cycle steps, however, cannot be eliminated for packaged systems with embedded computer system systems purchased from, or subcontracted to, a vendor. The degree of redundancy in the life-cycle model used for validating existing computer system systems will be specified in the Validation Master Plan.

For systems that have been in use for some time, a compilation and review of documentation and a review of historical data supplemented by a series of functional tests may be adequate to demonstrate that the system performs its intended function. Evidence that the software is structurally sound may be provided by a formal evaluation of the supplier's software development and testing procedures and by an analysis of historical, system-related data. Where historical, system-related data is not available, for whatever reason, additional functional testing may be required.

Recent Inspection Findings

- Lack of documentation demonstrating an adequately validated system. [FDA Warning Letter, 2001]
- Risk assessment revealed numerous unanticipated risks that have not been addressed. For example, one such risk is that the computer unit may acquire the wrong patients' data. [FDA Warning Letter, 2001]
- There were three possible causes attributed to this failure in the System Risk Assessment document, yet there is no implemented strategy to reduce the risk of these failures. [FDA Warning Letter, 2001]
- Your response fails to trace back to source code, and the related software development cycle which establish evidence that all software requirements have been implemented correctly and reliably and has been validated. Software is validated in its controlled development and in control of ongoing maintenance of the software and its documentation throughout its lifetime. [FDA Warning Letter, 2001]
- The XXXX computer system … is not validated to the current corporate standards. [FDA Warning Letter, 2002]

MANAGEMENT REVIEW

A management review will usually be conducted periodically, with one or more validation reports used as feedback on the overall validation program. Although shown as the last phase of management cycle, reviews can and should take place throughout validation. The review endeavors to draw out lessons from the validation conducted to date, to consider the impact of any regulation developments, and to report any recommendations. The overall aim is the continuous improvement of the company's validation capability.

Projects can provide a central focus for applying and refining policies and procedures. Feedback from practical experience is vital if a cost-effective validation approach is to be established. An external validation consultant may be seconded to provide an independent perspective, to comment on current industry practices, and to provide updates on topical regulatory issues.

Management reviews may make changes to the corporate validation strategy, SOPs, and the Computer System Inventory (adding new systems or removing decommissioned systems). Validation rework (additional testing) may also be required. All these matters can have an impact on the Validation Master Plan, so the management cycle continues. It must effectively address the inspection issues raised by GxP regulatory authorities concerning computer system systems. Validation does not have to be unduly expensive if the issues involved are managed in a timely manner.

It is recognized that senior management in pharmaceutical and healthcare companies are faced with multiple and changing priorities, e.g., customer service, quality, financial performance. Nevertheless, it is very important that the level of support given to compliance is sustainable. Should it falter, the organization will face pendulum swings of compliance investment and compliance underfunding. Such feast and famine nearly always leads to a serious noncompliance sooner or later as validation practices try to adapt to the level of prevailing financial support. Senior management should beware of making arbitrary cuts and instead work on cost-effectiveness improvements. Equally, throwing money at compliance does not necessarily result in "solving the problem." Senior management must avoid the notion that compliance is a one-off project type activity. Inspection readiness results from maintaining the ongoing compliance of legacy systems that must be properly supported; otherwise their compliance will be compromised over time.

Recent Inspection Findings

- No management review procedures and no documented management reviews. [FDA Warning Letter, 2001]
- Quality audits did not verify effectiveness in fulfilling quality system objectives. [FDA 483, 2002]

RESOURCE PLANNING

Pharmaceutical and healthcare companies should have adequate numbers of competent personnel to implement the GAMP guidance and the standards so implied. Individuals should not be intimidated by their responsibilities but, on the other hand, the principles of validation should not be compromised either. GxP regulatory authorities will not accept staff shortages or deficient training as excuses for noncompliant validation.

Senior managers are often asked by their companies to ensure a successful GxP inspection with minimum resources. It is a difficult task, and senior managers themselves need guidance. ISO 9001 gives the following advice:

> *... The responsibility, authority, and interrelation of all personnel who manage, perform and verify work affecting quality shall be defined.*[2]

The role of senior management is not limited to validation policies and procedures. Senior managers should also sponsor validation projects and ensure that the necessary supporting organization is established. Project roles will normally include the following:

System Owner/User: Responsible for defining system requirements, ensuring that validation is conducted in a compliant manner, that appropriate user procedures are in place, that users are trained, and that validation is maintained once the system is in use, right on through to decommissioning.

Developer: May be internal or external to pharmaceutical or healthcare company's organization. He or she is responsible for the technical development, implementation, and handover of the system. This includes the quality assurance attributes of the work.

Operational Support: May be internal or external to pharmaceutical or healthcare company's organization. He or she is responsible for the technical support during operation and maintenance, including decommissioning. This includes the quality assurance attributes of the work.

Quality and Compliance: Fulfill regulatory expectations of "QA/QC Unit" — responsible for interpretation of regulatory requirements into policy and procedures. He or she is responsible for validation oversight (compliance audits and approval of key validation documentation). This role is typically also responsible for inspection-related liaison with regulatory authorities.

Most pharmaceutical and healthcare companies will split the roles mentioned above into several specific jobs as appropriate to their organizational structures. For instance, the Quality and Compliance role may be split into an Operations Quality Representative and a Validation Expert. Equally, the Developer role may consist of Project Manager, Project Engineers, and Supplier Representative. Similarly, System Owners may sometimes delegate responsibilities such as specific maintenance activities to a system administrator function that may be internal or external to the company. The Quality and Compliance role, however, must not be combined with other roles; otherwise its independence will be compromised.

GxP regulatory authorities expect to see an organizational chart with the specific duties and job descriptions of individuals recorded. Individuals must be given sufficient authority to fulfill their duties. Duties may be delegated to designated deputies with satisfactory levels of competence. There should be no unexplained gaps or overlaps in duties affecting validation. Senior management is responsible for ensuring personnel assigned to validation work are competent to fulfill their roles, and for arranging any supplementary training requirements. It is not acceptable for senior managers to rely on individuals to fulfill their roles without management support. The quality representative should be independent of the project management to ensure independence and impartiality. This will prove crucial if things go wrong later on.

A validation consultancy firm has conducted validation work for several North American and European pharmaceutical manufacturers and has developed a staffing life-cycle model.[16] This firm's experience suggests that this model is rather generic, in that it seems to fit most computer system validation programs. It can be used to assess the total validation workload. Figure 3.8 shows that the staffing profile consists of three phases: preparation, implementation, and maintenance.

The preparation phase is a period for developing the validation master plan, the Computer System Inventory, the SOPs, and sanctioning the Computer System Validation Program. This phase will often require some external consultancy.

The implementation phase is for the validation of existing computer systems, which is usually allocated a 2- to 3-year time frame for completion. The staffing requirements for this phase are usually met using a combination of in-house and contractor resources. The in-house staff members control the implementation of the validation program for individual computer systems. Contractors provide an on-site engineering resource for the duration of the validation program. Contractors can be particularly useful if they are already familiar with computer system validation, since this accelerates their understanding of a new set of validation procedures. Some elements of validation work can be partitioned off into work packages that can be conducted by contractor staff based off-site. It should be the objective of both pharmaceutical and healthcare companies and suppliers to develop a successful partnership. Contractors are usually deployed because the extra effort required is temporary and the pharmaceutical or healthcare company does not wish to increase its headcount of permanent staff.

The final staffing life-cycle phase is entitled "maintenance" and covers the prospective validation of new computer system systems and periodic revalidation of current computer systems. This phase is typically managed in-house, using work packages contracted to suppliers. At this stage, a set of

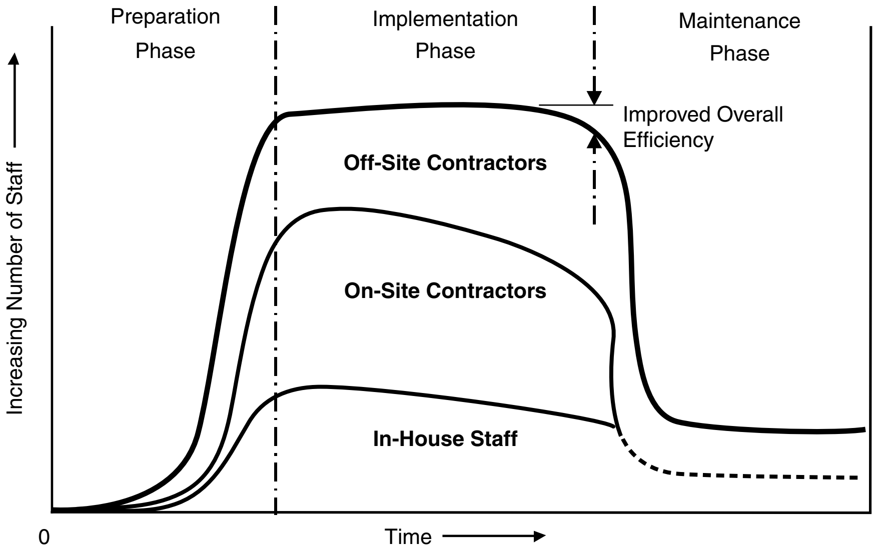


FIGURE 3.8 A Staffing Profile.

preferred suppliers will have been established who have a successful track record with pharmaceutical and healthcare companies.

REFERENCES

1. OECD (1995), *The Application of the Principles of GLP to Computerised Systems*, No. 10 OECD Series on Principles of Good Laboratory Practice and Compliance Monitoring, GLP Consensus Document, Environmental Monograph No. 116, Organisation for Economic Co-operation and Development Environmental Directorate, Paris.
2. ISO (2000), ISO 9001: *International Standard: Quality Systems — Model for Quality Assurance in Design/Development, Production, Installation and Servicing*, International Organization for Standardization, Geneva.
3. Stokes, T. (1994), The Role of Senior Management in Computer Systems Validation, in *Good Computer Validation Practices: Common Sense Implementation* (edited by T. Stokes, R.C. Branning, K.G. Chapman, H. Hambloch, and A.J. Trill), Interpharm Press, Buffalo Grove, IL.
4. Bruttin, F. and Dean, D. (1999), A Risk-Based Approach to Reducing the Cost of Compliance in Pharmaceutical Manufacturing, *Pharmaceutical Technology Europe*, pp 36–44.
5. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (www.ispe.org).
6. ISPE (2001), *Baseline Pharmaceutical Engineering Guide: Qualification & Commissioning*, International Society of Pharmaceutical Engineering, Tampa, FL.
7. Canadian Health Products and Food Branch Inspectorate (2000), *Good Manufacturing Practices — Risk Classification for GMP Observations*.
8. Taylor, J., Turner, J., and Munro, G. (1998), *Good Manufacturing Practice and Good Distribution Practice: An Analysis of Regulatory Inspection Findings*, *The Pharmaceutical Journal*, November 7, 1998. The Pharmaceutical Press, The Royal Pharmaceutical Society, London.
9. TGA (1990), *Australian Code of Good Manufacturing for Therapeutic Goods*, Medicinal Products — Part 1, Therapeutic Goods Administration, Woden, Australia.
10. European Union (2001), *Annex 11 — Computerised Systems*, European Union Guide to Directive 91/356/EEC.
11. Japanese Ministry of Public Welfare (1993), Guideline on Computer Systems in Drug Manufacturing, in *Manual on Computer Systems in Drug Manufacturing,* Tokyo.

12. FDA (1983), *Guide to Inspection of Computerized Systems in Drug Processing*, Technical Report, Reference Materials and Training Aids for Investigators, U.S. Food and Drug Administration, Rockville, MD.
13. Pharmaceutical Inspection Co-operation Scheme (2003), *Good Practices for Computerised Systems in Regulated GxP Environments*, Pharmaceutical Inspection Convention, PI 011-1, Geneva, August.
14. NE 58: Abwicklung von qualifizierungspflichtigen PLT-Projekten (version 04.06.96); [including …] Dannapel, B., Hensel, H., Mockel, B., Muhlenkamp, J., Muller-Heinzerling, T., Otto, A., and Teuchert, V. (1995), Qualifizierung von Leitsystemen: Ein Gemeinschaftsprojekt von GMA und NAMUR zur Validierung, *ATP — Automatisierungstechnische Praxis*, 37 (10): 64–76.
15. PDA (1995), Validation of Computer Related Systems, *PDA Journal of Pharmaceutical Science and Technology*, Technical Report No. 18.
16. Wingate, G.A.S. (1997), *Validating Automated Manufacturing and Laboratory Applications*, Interpharm Press, Buffalo Grove, IL.

APPENDIX 3A
KEY PRINCIPLES FOR COMPUTER SYSTEM VALIDATION

These principles apply to computer systems that can affect the quality of drug and healthcare products. Such computer systems include laboratory systems, process control systems, spreadsheet and database applications, business systems, and associated computer network infrastructure.

FOUNDATION PRINCIPLE

1. Validation is an ongoing process of establishing documented evidence that provides a high degree of assurance that a computer system will consistently perform according to its predetermined specifications. A key consideration is the protection of the integrity, authenticity, and security of data relating to the quality of drug and healthcare products and/or supporting regulatory submissions.

MANAGEMENT REQUIREMENTS

2. All new systems requiring validation must be validated prospectively. Existing systems that require validation, but have not already been validated, must be retrospectively validated.
3. Validation must be planned, executed, and reported. Validation encompasses the entire life of the computer system, from planning through development and implementation, use and operational support, to decommissioning. Responsibilities and accountabilities must be defined and documented.
4. Computerized systems must have documented authorization to be used in their operating environments. Any restrictions in the use of a computerized system resulting from its validated status must be recorded at the time of its authorization.
5. Validation must employ predefined procedures and plans designed to build in quality during all stages of the computer system life cycle. The effectiveness of these procedures must be assessed periodically and improvements made as required.
6. System requirements must be traceable throughout validation records.
7. Suppliers of computer systems and associated services must be managed to assure that the software, hardware, and/or related services they supply are fit for purpose.
8. Enhancements and modifications to computer systems and associated documentation must be implemented under change control and, where appropriate, configuration management. Changes affecting the validated status of a computer system must be approved before they are implemented.
9. Those involved in the development and implementation, use and operational support, and decommissioning of computer systems must have the documented education, training, and experience to fulfill their duties.
10. Rationales must be developed and documented to justify validation decisions not supported elsewhere.

PROJECT REQUIREMENTS

11. User requirements and design must be specified, documented, and approved.
12. Software controls must be used to manage programming.
13. Development testing of computerized systems must be documented, and must cover structural and functional attributes.

14. Design reviews (also known as design qualification) must be conducted and documented to verify that user and regulatory requirements are satisfied in the wider system context including equipment, processes, and manual interaction.
15. User/site acceptance testing of computerized systems (known as qualification) must cover installation, operation, and performance in the wider system context including equipment, processes, and operator interaction (i.e., Installation Qualification, Operation Qualification, and Performance Qualification).
16. Data migration must preserve the integrity and security of original data. Processes and methods used to load data (manually and automatically) must be defined and validated with supporting documentation before they are used.

Operation Requirements

17. The performance of validated computer systems must be monitored against predefined requirements to demonstrate acceptable operational service.
18. Preventative maintenance and calibration, where required, must be planned, conducted, and documented.
19. Software changes and upgrades must be conducted according to defined procedures and documented.
20. Records must be established to demonstrate data integrity being maintained.
21. Backups of software, configuration, and data must be planned, conducted, and documented. Backup data must be readable and therefore retrievable.
22. Archiving of software, configuration, data, and associated documentation must be planned, conducted, and documented. Storage media must be retained for a predefined period at a separate and secure location under suitable environmental conditions, be protected against willful or accidental damage, and be periodically checked for durability and restoration. Archived materials must not be destroyed until this retention period has expired.
23. The procedures to be followed if the computerized system breaks down and is unavailable must be documented and periodically verified.
24. Access to computerized systems and associated functionality must be restricted to authorized persons and documented.
25. Formal agreements regarding operational support (e.g., contracts and service level agreements) must be established defining responsibilities and accountabilities.
26. User procedures must be established and trained out to ensure that computer systems are consistently used.
27. The validation status of computerized systems and the cumulative effect of change must be periodically reviewed and any required revalidation conducted.
28. Decommissioning of computer systems must be planned and conducted in accordance with defined procedures.

Responsibilities and Accountabilities

System Owners/Users are accountable for assuring that computer systems used by them, or on their behalf by other organizations, in support of pharmaceutical regulatory requirements and pharmaceutical regulated areas or processes are compliant with pharmaceutical agency regulations, are properly validated, and are used in a compliant manner.

Developer and Operational Support organizations are responsible for the technical delivery and quality of work of their associated activities supporting validation. These activities must be conducted in compliance with regulatory requirements. Developer and Operational Support organizations have a mutual responsibility to ensure that project hand-over activities are appropriate and completed.

Quality and Compliance are responsible for establishing necessary policies and procedures to manage validation and for approving validation work. They must be able to demonstrate their independence to the System Owner/User, and to the Developer and Operational Support organizations.

4 Supporting Processes

CONTENTS

Successful validation depends on the satisfactory operation of a number of underlying supporting processes. Among these are training, document management, change control, configuration management, self-inspections, and managing deviations. Validation is fundamentally flawed without them, and so they are discussed here.

TRAINING

All personnel (permanent, contractors, consultants, and temporary staff) developing, supporting, or using a GxP computer system must be trained so that they acquire the necessary level of competency before they may be allowed to perform their designated duties. To this end, all personnel involved in any aspect of validation should have:[1]

- A role description
- Appropriate qualifications that have been documented
- Training plans and completed training records

Organizational Roles

The organizational structures in the enterprise, whatever its size, must be defined and documented. Organizational charts must be maintained. Of critical importance in these is the Quality and Compliance group (or QA), whose role and reporting relationships must be explained. It is essential to recognize that the regulatory authorities will hold the QA organization accountable for the firm's compliance with regulatory requirements, including those for validation.

Role descriptions for individuals should be prepared and kept up to date. Evidence that individuals have sufficient education and experience to enable them to undertake their assigned functions must be collected and kept ready for presentation when required. The delegation of duties to qualified individuals should include a definition of deputies so that working practices are not hamstrung when key staff are absent.

Inspectors expect senior management to appreciate and understand the implications of regulatory requirements on their business. Senior management must ensure that an adequate number of personnel are available with the necessary qualifications and practical experiences appropriate to their responsibilities. The discovery that such numbers of personnel are insufficient will attract criticism, since this is likely to lead to individuals being burdened with excessive responsibilities and workloads, and subjected to inordinate pressure. The consequential risk is that quality will then be compromised in some way. In the worst case situation, senior company executives are subject to potential prosecution if they fail to meet such regulatory expectations.[2]

Qualifications

When recruiting, pharmaceutical and healthcare companies should try to verify the details of the education, training, and experience claimed by the candidates. Copies of their certificates should be requested and retained. Personal references should also be taken up although their value should be weighed with care, remembering that the commendations given can be presented in a politically correct fashion that carries a hidden, subliminal, and rather less favorable implication! For example, consider the following, embarrassingly flattering accolade:

> *You write to ask me for my opinion of XXXX, who has applied for a position in your department. I cannot recommend him too highly, nor say enough good things about him. The validation he conducts is the sort of work you don't expect to see nowadays. His documentation clearly demonstrates his complete capabilities. His understanding and appreciation of regulatory requirements will surprise you. You will indeed be fortunate if you can get him to work for you.*

Curriculum Vitaes (CVs) for permanent staff are usually kept by the Human Resources Department. This is not necessarily the case for contractors, consultants, and temporary staff. Their CVs, typically retained by the responsible manager, can be easily lost when individuals move on to new contracts. One way to systematically capture such training records for contractors, consultants, and temporary staff is to attach them as appendices to Validation Plans or Validation Reports.

The profile suggested for a computer validation practitioner is given in Wingate[3] and comprises:

- Technical background involving computer systems
- Technical qualifications associated with computer systems
- Two or more years of GxP experience, not necessarily involving computer systems

Validation practitioners must have a measure of tenacity and self-discipline so that they stay the course, progressing work through to completion without constant supervision. This is not to say that they should be discouraged from seeking advice, but rather that they should have sufficient judgment and validation knowledge to make some basic decisions themselves.

The profile suggested for a computer validation expert is also given in Wingate[3] and comprises:

- Graduate in science discipline
- Four or more years of GxP experience with computer systems
- Very good communicator
- Validation experience with more than one organization
- Journal publications and conference presentations

Validation experts must take care not to assume that there is only one way to validate. Rather they should be flexible in their approach, so that when the inevitable problems arise they do not instinctively resist exploring new solutions. Validation experts do not have to be leaders, provided they have management support and can articulate their views to management in order that they can receive appropriate direction.

Technical personnel such as those providing computer system development and support skills should also have the necessary qualifications to fulfill their roles. This requirement is specifically stated in the U.S. 21 CFR Part 11 regulation covering electronic records and electronic signatures.

Managers are likely to be qualified by experience, perhaps supplemented by training. Personal attributes, and especially attitudes, are of critical importance, but this theme is not developed here. Educational attainment alone does not ensure that a manager has the competencies required to manage effectively.

Training Plans and Records

Training Plans should be used to manage the development of staff, and subsequent training should be conducted in accordance with approved procedures. All personnel should be aware of the principles of GxP affecting them and receiving initial and continuing training relevant to their job responsibilities. This includes those personnel developing, supporting, and using computer systems. It is important to recognize that necessary training be provided prior to the need for the use of the associated competency arising, rather than when the lack of such competency has already been painfully demonstrated!

Training records must be maintained. Some pharmaceutical and healthcare companies make use of questionnaires to try to verify in a formal way that personnel have understood their training and really acquired the intended competence. Authorized assessors should be engaged to mark such questionnaires. If personnel fail to pass such a competency test, some supplementary training is required. Care should be taken not to simply repeat the original training and the examination. Perhaps the training materials or delivery were at fault, and they may require improvement. There may be a systematic reason why individuals have not understood what they have been taught.

The performance of personnel should be periodically reviewed to identify any refresher training requirements. Some pharmaceutical and healthcare companies achieve this through an audit or a Periodic Review of training records, which must be updated to reflect the training received. Marked competency questionnaires or test papers should be attached to training records where possible.

Recent Inspection Findings

- There are no records to document that the Information Technology (IT) service provider staff personnel have received training that includes current good manufacturing practice regulations and written procedures referred by the regulations. [FDA 483, 2000]
- There is no documentation to indicate that (users) are trained in the software and its applications. [FDA Warning Letter, 2000]
- There is no assurance that adequate training was given to all analysts on how to use the *[computer]* system software [FDA 483, 2002]
- There is no documentation that a qualified person reviewed the training records [FDA Warning Letter, 2000]
- Corrective action to noncompliance with SOPs consists of retraining in the same manner as initially trained. There is no limit to the frequency of retraining. [FDA 483, 2000]
- There is no evidence that the training provided during 2000 and 2001 to your analysts is adequate as evidenced in the following events. The efficiency and adequacy of the training program is questionable in that numerous training sessions are performed during the same day (a specific list of 8 training session a particular employee received on the same day was then listed). [FDA 483, 2002]
- Failure to document that personnel employed in drug manufacturing operations are trained on a continuing basis and with sufficient frequency to assure they remain familiar with current Good Manufacturing Practice requirements applicable to their assigned function. [FDA Warning Letter, 2000]
- Your response to this letter should include [your] plan for establishing a system of training and evaluation to ensure that personnel have the capabilities commensurate with their assigned function. [FDA Warning Letter, 2000]
- The current training procedure for employees does not determine the proficiency or comprehension at the end of training. [FDA Warning Letter, 2000]

DOCUMENT MANAGEMENT

Documentation of research, development, and manufacturing practice is vital to pharmaceutical and healthcare companies because, unless they do this, they have no way of demonstrating validation to the various GxP regulatory authorities. Examples of documents include policies, procedures, plans, reports, and operational data. Regulatory inspectors will expect to see document management procedures established covering preparation, review, approval, issue, change, withdrawal, and storage. This is especially important for contractual documents and those documents endorsed by the pharmaceutical or healthcare company's QA organization.

Document Preparation

Documentation standards should be defined so that there is consistent document layout, style, and reference numbering. Documents should be clearly marked as draft until they are formally released. Version control should be apparent. The version identifiers should distinguish documents under development (drafts) from those that have been issued formally. Documents should include a document history section to log the changes made in successive issued versions of the document.

Individual documents should have the following controls:

- Document Title
- Document Number
- Version Number

- Page x of y
- Date of Issue
- Copy Number

Some organizations include a date for the next routine review of the document. This ensures that even if no changes have occurred, the document will still be examined to verify that it is still relevant and accurate.

Documentation has progressed enormously from the days when word processing represented the apex of efficiency and automation in this arena. Special considerations for more sophisticated document types are given in Table 4.1. In some instances it is recommended that the document type should be stored with or within the document to make ongoing document maintenance easier. This also applies to some types of embedded documents.

Document Review

Documentation should be subject to review, prior to its formal release. Such a review might assume a number of forms ranging from the evaluation of the document and collection of comments through to its inspection within formally convened review meetings. The reviewers should be identified in advance within the Validation Plan, the Project and Quality Plan, or the document management procedures. Many firms give staff guidance on how to decide the most appropriate reviewers for different documents.

The review can be recorded using a template or simply recording meeting minutes in the traditional manner. In either case, the date of the review and the names of the reviewers should be noted. It is recommended that a multidisciplinary team that includes technical and quality representatives review documents. Each reviewer does not necessarily have to be an authorizing signatory for the document under inspection, as long as his or her comments are included in the review records.

For the review process to be effective, reviewers must come prepared. Copies of the documents under review should be distributed and scrutinized prior to the meeting. A chairperson for the review meeting should be nominated beforehand. Similarly, remote reviews will require the appointment of an individual to coordinate and collate the review feedback.

The review must systematically cover each section of the document. If a section attracts no comments, this should be indicated in the records. Any corrective actions identified in the review must be assigned to a named individual with a completion date. The progress of individual actions should be tracked through to closure. Care must be taken to ensure that associated documents are also reviewed and updated as necessary.

Sometimes in the absence of any consensus, a compromise on the content of a document will have to be reached. Such compromise positions should be agreed upon before approval is sought, in order to avoid delaying the approval process. Normally the document author has the responsibility for resolving these issues. Often this is not simple, especially when reviewers have entrenched and opposing views! An escalation route to resolve any impasse should therefore also be defined and agreed upon beforehand.

At a recent conference, practitioners expressed the opinion that, in their experience, 90% of review comments related only to format and style. In spite of this, over 90% of the problems arising from poor documentation could be attributed to omissions and inaccuracies! Something is wrong here. Reviewers need to bear these statistics in mind and strive to make their reviews as effective as possible in identifying the defects in documents — defects that will cost time and money later on.

Once the agreed changes have been incorporated into the document, it is ready for approval. The document history should be created to record the changes made. There is no need for the document history to affirm what remained the same. Document histories are usually written in the form of a summary at the beginning or end of a document.

TABLE 4.1
Characteristics of Various Document Types

Document Type	Characteristics	Examples of Application Areas	Special Precautions
Portable Format Document	A homogeneous document type created from many other types of document but stored in a standard or proprietary file format. The international standard is SGML. Proprietary formats include Adobe's PDF format. A semi-portable file format is HTML, which is used on the Internet World Wide Web. See also the compound file formats. Files cannot usually be edited. Editing tools normally require an operating system with a graphical user interface (GUI).	All document types including more complex ones such as integrated batch documentation, illustrated SOPs. All document types must be stored in a "neutral" file format.	Specify file format, application, and version. Special printer drivers may be necessary.
ASCII Text Document	The simplest document type to manage. Typically created in word processor, it consists only of characters belonging to the ASCII or ANSI sets (text and some symbols). Such documents can be viewed by the word processor program itself or a file viewer program in most systems. Many proprietary formats exist but the most popular have become de facto standard formats.	Memos, master production and control records, SOPs, deviation reports, validation protocols, manual batch documentation, and many more.	Specify file format, application, version, and language.
Graphical Document	A homogenous document type, stored in a standard graphical file format. Includes scanned paper documents. Many file formats exist, from raw bit-mapped pictures to highly complex vectored drawings in a CAD environment. Simplest file formats are bit-mapped formats (e.g., TIFF, PCX, GIF, JPEG) or generic vectored formats (e.g., WMF, CGM, DXF). Many proprietary formats exist. Some CAD formats include product database information.	CAD drawings, SOP illustrations, scanned paper documents, label pictures for batch documentation.	Specify file format, application, version, and language.

Review records should not be destroyed, at least until the document is approved and formally issued. Many projects have a policy of retaining all document review minutes and records until the computer system is handed over and commissioned into use. Even then the project files containing the review records may be retained for a few years just in case some question or defect arises in the future.

Document Approval

Just like reviewers, document approvers should be identified in advance within the Validation Plan, the Project and Quality Plan, or the document management procedures. Again, many firms guide staff on the most appropriate choices of personnel for the sensitive and important task of document approval.

The number of signatures on individual documents should be monitored. There are usually four principal signature roles (not all required for each document):

- Technical approval
- Regulatory compliance
- Compliance with corporate procedures (including format)
- Authorization to proceed

There is no reason why one individual cannot fulfill more than one role, provided he or she has appropriate competencies. There is one prohibition, however; no single individual should represent both quality and technical roles. The minimum requirement is for two signatories, representing quality and technical aspects, respectively.

Documents with up to 10 signatures are common where the number of signatories is not controlled. During a survey at one European pharmaceutical company, a document was found bearing no less than 18 signatures! Was this really necessary? Indeed, too many signatories will retard the release of documentation, while some practitioners have argued that many signatures lead to less effective document scrutiny rather than more. It is not hard to see why — human nature being what it is. The temptation to believe that the effort of an effective review is pointless because so many others have already endorsed it becomes almost irresistible (a phenomenon also known as the *rubber stamp effect*). Furthermore, what personal price will be exacted for questioning the combined wisdom of so many colleagues? There is thus some truth in the cynic's maxim that the quality of a document is inversely proportional to the number of approval signatures.

The rationale for the presence of each approval signature should be unambiguous and documented — signatories should know why they are signing the document! Example approval signatures include technical authority, QA compliance, and user acceptance. Regulatory authorities normally require key validation documents to have formal signatures from two or more authorized persons. Signatures should be written in black or blue ink as the pigments and dyes in them render the signatures more resistant to fading. Approval signatures should be dated and accompanied by the name of the person signing, as the identities of some signatures are indecipherable (a weakness normally associated with the medical profession and their prescriptions, but also all too common in the validation world). Interestingly, in some countries such as Japan it is legally admissible as a signature to use a mark or stamp that does not identify the person's spelled name.

Document Issue

Approved documents should be distributed in accordance with a defined distribution list. By minimizing the number of copies, the task of retrieving and updating distributed documents will become much easier. Some organizations print documents on colored paper so that a black and white photocopy can be easily recognized as a copy of the master. It may be necessary for pharmaceutical and healthcare companies to agree with suppliers who receive copies of their documents the nature of any confidentiality agreements that must be established in advance.

The identity of custodians of controlled copies of documents should be defined in the Validation Plan, the Project and Quality Plan, or the document management procedure. The allocation of released documents to these individuals should be controlled through managed distribution lists. Superseded versions of controlled documents must be replaced in a comprehensive and timely fashion. Obsolete versions of documents must be clearly marked as *superseded*.

Uncontrolled copies should be identified on the document as such, and users notified that they are responsible for checking before use whether the document has been superseded. If an electronic document management system is in place, it should ensure that printed paper copies are endorsed to the effect that the paper copy is not an authoritative document but simply a copy of an electronic master at an instance in time.

In some organizations the approval step is associated with setting an "effective date" that must be reached before the document may be used. The effective date is usually defined to allow a period for dependent activities, for instance the distribution and training implied in the associated SOPs. Where effective dates are deployed, they must be prominently displayed on the document's frontispiece.

Safeguards should be instituted to prevent the unintended use of unapproved, superseded, or withdrawn documents. Many firms use audits for this purpose. Management must ensure that personnel do not retain unauthorized copies of documents (e.g., photocopies), as these could not be relied upon after they have been revised. Where paperless systems distribute copies of documents, these should be clearly marked as having a limited shelf life. Beyond this date they lose all validity *whether or not they have been superseded*. A shelf life of one week is often recommended.

Document Changes

All changes to released approved documents must be subject to change control. The revised document should be clearly marked as a draft and managed accordingly as described above. Modifications to approved documents should be reviewed and approved by the same functions/organizations that performed the original review and approval, unless specifically designated otherwise. Despite changes in individual signatories there should be a consistent allocation of responsible review and approval roles.

Document Withdrawal

From time to time documents must be withdrawn from use, for all kinds of reasons. In this situation, document keepers can be asked either to notify the central distribution group that they have destroyed their copy, or to return their copy to the central distribution group for disposal. Dealing with uncontrolled copies is much more difficult. Some firms send e-mails to relevant parts of their organizations notifying them of withdrawn documents. Alternatively, many firms rely on audits to pinpoint the continued availability and use of withdrawn documents.

Document Administration and Storage

It is wise to keep the organization and administration of documentation as simple as can be conceived. Complex systems are much harder to manage successfully. Centralized vs. distributed administration of documentation has advantages and disadvantages. Centralized administration offers economies of scale and easier control of master documents, as the latter are held in one location. Distributed administration meanwhile offers more "ownership" because it is closer to its users and it is easier to plan for busier periods. Most organizations tend to centralize administration on a site basis, but either approach can be adopted as long as it is controlled. In theory, the best of both worlds is available through the implementation of an electronic document management system (EDMS). Such systems must of course be validated!

Master copies of documentation should be stored in a safe and secure location, according to defined procedures. These master copies should be stored with:

- Approval signatures
- Document history
- Change control records
- Document distribution records where applicable
- Superseded versions, clearly marked as such
- Withdrawn documents, clearly marked as such

Stored documents should be protected against accidental and malicious damage. They must be retrievable in a legible format throughout their predefined retention period. This usually means that a minimum of two copies is retained, each in a separate place, just in case of accidents. Once the retention period has expired, a decision can be taken whether or not to destroy the master copies. A record of destruction, evidence that the document once existed but has since been destroyed, should be made and retained for a further period.

A document index should be maintained to log documentation by reference/title, version, and physical storage location. As the status of a document changes, the document index will need to be updated.

Quality of Documentation

The quality of documentation must be assured. Poor documentation is often marred by rambling, unfocused, and verbose text, with omissions in some areas and excessive detail in others. This impedes its use as well as undermines the goal of achieving GMP compliance. Those preparing documentation and records should therefore ensure that documents meet the *six virtuous Cs*, i.e.:

- Concise
- Complete
- Consistent
- Comprehensible
- Correct
- Controlled

Wherever possible, keep documents short (preferably fewer than 20 pages) and avoid the duplication of information.

Basic regulatory expectations include:

- Mistakes being altered correctly: single strike, initialled and dated, with a brief reason for the correction (or a reference to a change control number if appropriate)
- Avoiding use of dittos or arrows, as regulatory authorities consider them insufficiently descriptive where actual values with corresponding signatures are needed
- Avoiding transcriptions, even if the original document/record looks messy
- Ensuring that white opaque correction fluid is *never* used, as it hides the original information

Regulatory authorities will search documentation for certain vague words that, in their experience, are often associated with imprecise documentation:

- Calculate — is the actual calculation intended to be used specified?
- Automatic — is the degree of manual intervention specified?
- Typically, usually — exactly how often is meant here?
- Normally — what is normal, what is abnormal?
- Appropriate — what is appropriate, what is not appropriate?

A simple word search can be used on word processors when a document is being written to identify the use of these words, which should then either be replaced with alternative phrases or be clarified to alleviate the uncertainty.

Recent Inspection Findings

- Lack of appropriate documentation procedures. [FDA Warning Letter, 2001]
- Lack of procedures to ensure records are included with validation documentation, maintained, and updated when changes are made. [FDA Warning Letter, 2001]
- Significant deficiencies regarding documentation controls were reported. Documents were either not dated, lacked a documentation control number, were missing, were reported in pencil on uncontrolled pages, or dates were crossed out without initials, dates, or explanation. [FDA Warning Letter, 2001]
- Errors on batch production, control and lab records must not be erased or overwritten (interpret as no whiteout). A line must be drawn through an incorrect entry and the corrected figure or word written neatly and initialled. Significant data must not be discarded without explanation. To discard significant data, the data must be crossed out, initialled, and a valid reason for discarding the data explained. [FDA Warning Letter, 2001]
- SOPs do not clearly describe who must approve documents or what each type of approval represents. [FDA 483, 2002]
- Numerous instances were observed of lack of control of official controlled documents: use of incorrect version of testing forms, incorrect data sheet used because old sheets not replaced with new, incorrect log sheets were used. [FDA 483, 2002]
- Several pages were missing in printouts. [FDA Warning Letter, 2000]
- Two pages of a laboratory notebook were written in pencil and erased. Your abbreviation for … could be read on one of the erased pages. [FDA Warning Letter, 2000]
- Values in at least two *[laboratory records]* were altered. Altered values were written under computer generated values … and used in potency calculations. Review of the electronic data confirmed the incorrect values, which were part of your submission to the Drug Master File. [FDA Warning Letter, 2000]
- Typewritten dates (21/10/1999) were pasted over computer generated dates (04/01/1980). You stated that these … were generated on 04/01/2000 (day/month/year) and that the year printed out was the result of a Y2K glitch. But, the date pasted on the … was 21/10/1999. Either this explanation or the date … generated was wrong. [FDA Warning Letter, 2000]

CHANGE CONTROL

The following maxims of change are based on work by Lehman and Belady.

- *First Maxim of Change — Change will happen*
  Computer systems do not have static requirements. A system that is being used will almost certainly undergo continuing change either because its requirements were not fully understood in the first place or because the use of the system is changing. Change will only stop when the system's functionality becomes obsolete or it is judged more cost-effective to reengineer the system or replace it by a completely new version.

- *Second Maxim of Change — Change breeds change*
  Programmers often find it difficult to resist adding unsolicited functionality.
- *Third Maxim of Change — Change increases complexity*
  Software subject to change becomes less and less structured and thus becomes more complex. Extra effort is required when implementing changes to avoid increasing complexity.
- *Fourth Maxim of Change — Documentation eases change*
  The quality of documentation associated with computer systems is a limiting factor to the ease of implementing change over its operational life. Faster rates of change typically indicate the dominance of developing functionality over documenting the change. Slower rates of change may indicate system modifications being hindered by previous changes not being fully documented, or in recognition that developing functionality is being fully documented.
- *Fifth Maxim of Change — More resources do not imply faster change*
  There is an optimum level of resource for change. Applying more people to implement a change does not imply the change will be achieved faster. Indeed it can quite often add management complexity and slow change down.

These maxims should raise awareness of the need for effective change management. Quality and Compliance have to be preserved.

All changes to validated computerized systems must be reviewed, authorized, documented, tested (if applicable), and approved before implementation. Software cannot be partially validated. When a change, even a small change, is made to a software program, the validation status of the entire software system should be reconsidered, not just the validation of the individual change.[5] Retrospective validation and reverse engineering of existing software is very difficult but may be necessary in order to properly document and validate changes.

The procedure for Change Control can be divided in four phases[4] (see Figure 4.1):

- Request for change
- Change evaluation (impact analysis) and authorization
- Testing and implementation of the change
- Change completion and approval

Request for Change

A system owner should be (should have been) appointed for every system. This should be laid down in the system documentation (or validation plan). A proposal for a change should be directed first to the system owner, who shall be responsible for ensuring that all changes to the system are reviewed, authorized, documented, tested (if applicable), approved, and implemented in a controlled manner. The system owner may delegate this responsibility if permitted to do so in the validation documentation.

Any proposed change should be requested and recorded by submitting a *Change Request Form*. An example change control form is given in Figure 4.2. The Change Request part should include at least the following items:

- Requester name
- Origination date
- Identification of component or software module to be changed
- Description of the change
- Reason for the change
- Unique reference number, to be assigned by the system owner or his delegate using a logging mechanism

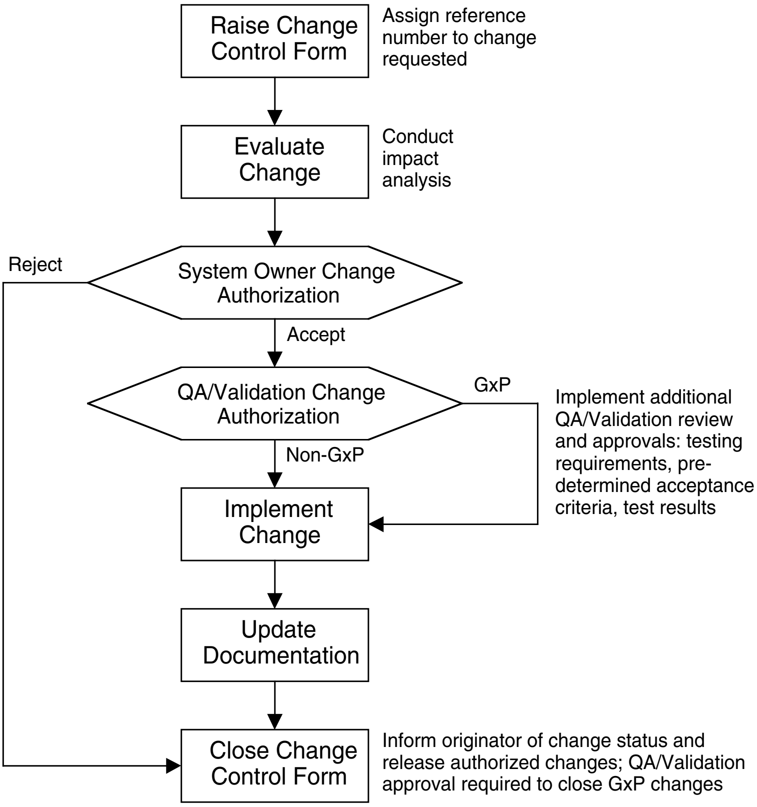

Raise Change Control Form
Assign reference number to change requested
Evaluate Change
Conduct impact analysis
Reject
System Owner Change Authorization
Accept
QA/Validation Change Authorization
GxP
Implement additional QA/Validation review and approvals: testing requirements, pre-determined acceptance criteria, test results
Non-GxP
Implement Change
Update Documentation
Close Change Control Form
Inform originator of change status and release authorized changes; QA/Validation approval required to close GxP changes

FIGURE 4.1 Change Control Process.

Change Evaluation (Impact Analysis) and Authorization

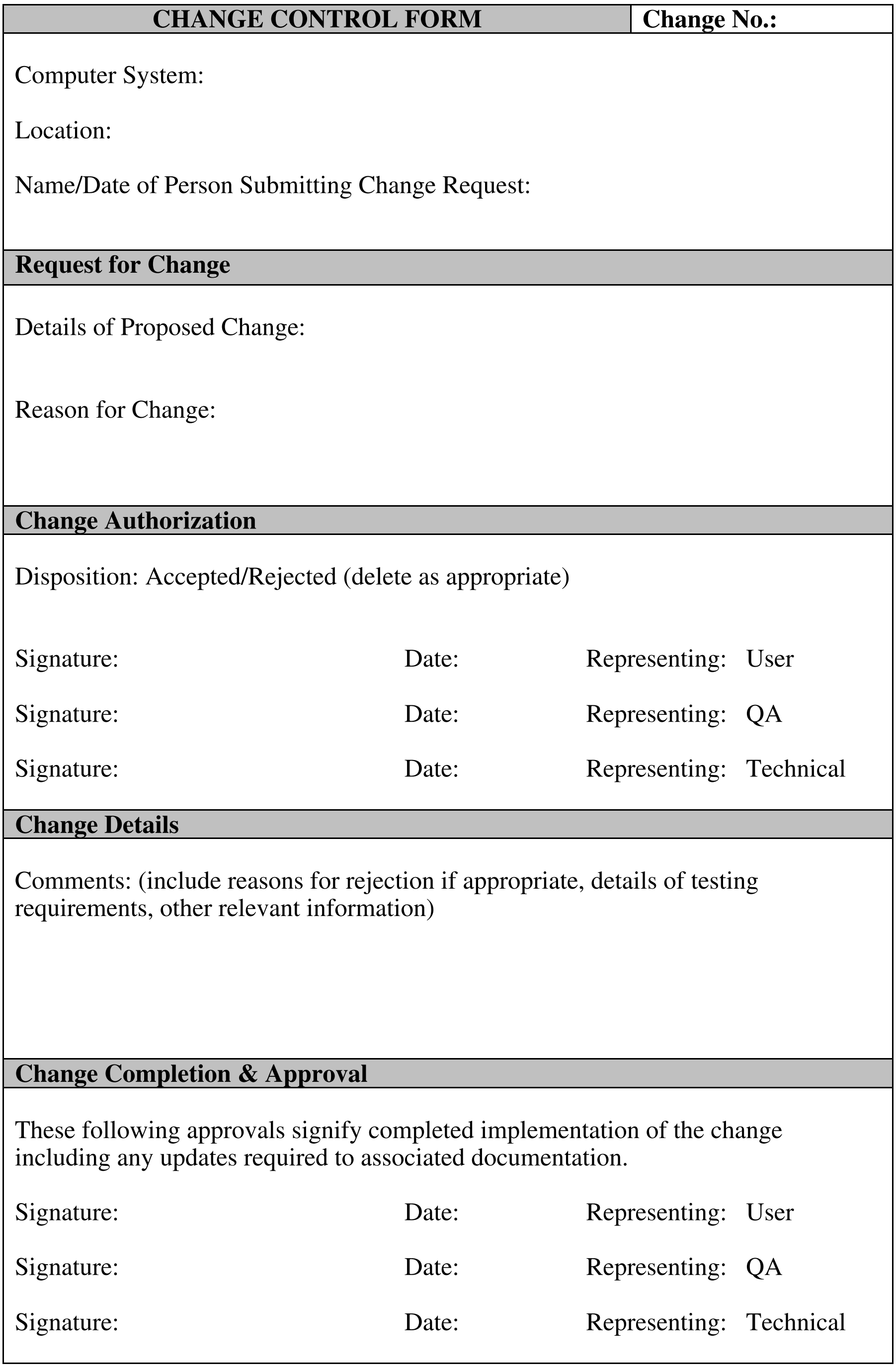
Each Change Request raised must be reviewed and a judgment made (accepted or rejected). In principle, for systems used for GxP-related activities, QA should be involved in the change control process.

For changes to small systems such as like stand-alone analytical systems (e.g., HPLC), it is usually quite clear which departments are affected by the change. Where changes to large systems such as Electronic Document Management Systems (EDMS) and Laboratory Information Management Systems (LIMS) are required, or where upgrades to central hardware (e.g., server) or software (e.g., operating system) associated with the network are needed, the impact on other applications is difficult to define accurately. For these kinds of changes a good *impact analysis* is very important. This exercise takes into account questions such as the urgency of the change, risk, schedule, cost (time and manpower), safety, and performance.

For minor changes that can be confidently expected to have no effect on the business process involved, the procedure can be accelerated. A list of change types may be specified that may be implemented on that system without a QA sanction for each. The prerequisites here are that:

- The simplified procedure must deliver equally good documentation
- It should be laid down in an SOP approved by QA
- Adherence to it should be subject to QA audits

For system changes with a scope wider than that solely of the department that owns it, the Change Request should be circulated to all the departments affected by the change. These should



CHANGE CONTROL FORM	**Change No.:**

Computer System:

Location:

Name/Date of Person Submitting Change Request:

Request for Change

Details of Proposed Change:

Reason for Change:

Change Authorization

Disposition: Accepted/Rejected (delete as appropriate)

Signature: Date: Representing: User

Signature: Date: Representing: QA

Signature: Date: Representing: Technical

Change Details

Comments: (include reasons for rejection if appropriate, details of testing requirements, other relevant information)

Change Completion & Approval

These following approvals signify completed implementation of the change including any updates required to associated documentation.

Signature: Date: Representing: User

Signature: Date: Representing: QA

Signature: Date: Representing: Technical

FIGURE 4.2 Example Change Control Form.

be identified by the system owner or the owner's delegate, and they must be obliged to review the change request with their appropriate technical, management, Quality Assurance, and user personnel.

An impact analysis should be documented on or attached to the Change Request Form. It should list the alternative solutions, potential impact on other systems or applications, and the required changes to the system documentation. The affected departments should give a recommendation to the system owner for the acceptance or rejection of the change. Once the impact of the proposed change has been assessed, the system owner or the owner's delegate must then decide whether to accept or reject the proposed change.

After acceptance of a change by the system manager, QA should be informed about the change. QA will review the change for its relevance to GxP. At this point QA can determine whether it would be appropriate for it to announce its own, separate endorsement to accept or reject. This will hinge on the GxP relevance of the system or the impact of the change on the validation status of the system. Future QA involvement in the rest of the change process depends greatly on this decision. Authorization by QA is required at several stages when the change is regarded as GxP relevant.

Changes may be implemented separately or collected into bundles for implementation. Consideration should be given in either case as to whether or not revalidation of the whole system is required. As more and more changes are applied, revalidation becomes increasingly appropriate.

Testing and Implementation of the Change

After evaluation and acceptance, the change can be effected, tested (if applicable), and formally commissioned into use. This principle applies equally to hardware and software; in the case of the latter, code redevelopment and testing should follow the same procedure as newly developed software. It is wise, if possible, to develop and test such changes in an isolated test/development environment before applying the change to the operational system.

Testing is necessary to determine whether the change works properly and has not compromised the system's functionality. The scope of testing should be based on the impact analysis. Where potential impact on other system functionality or other applications is identified, testing must be extended to include affected areas. This is sometimes referred to as regression testing.

Testing should be performed according to a test plan, and all testing should be fully documented (e.g., test description, test items, acceptance criteria, results, date of test, and names and signatures of persons who performed the test). While testing is of course necessary, it is vitally important to understand a critical principle where software changes are concerned: that the assurance of the safety of the change should rest far most heavily on a review of the change to the *design* of the software. If reliance is confined to test results alone, serious new flaws consequential to the change but quite unanticipated may be overlooked.

After implementation of the change (in the operation environment) the system owner should formally accept the change. This formal approval can be made based upon the test results, or the system owner might decide to perform some separate acceptance test.

Change Completion and Approval

In this phase, all the documentation concerning the change and all documents required for operation with the change need to be completed. It is important to identify and satisfy any training needs. The Change Request Form shall be completed and passed to the system owner for final review and approval. Depending on the GxP relevance of the system, or the impact of the change, QA should review and endorse the implementation of the change. QA should always be informed about the change by being sent a copy of the completed Change Request Form. The users should be informed (and trained if applicable) about the change. The system owner gives the final approval of the change and releases the system.

Recent Inspection Findings

- Firm's change control procedure does not include software changes. [FDA 483, 2003]
- Program for XXXX was changed but the change did not go through change control procedure. [FDA 483, 2001]

- Lack of change control documentation approving change in software. [FDA Warning Letter, 2001]
- No change control form was initiated or completed as part of the XXXX change. [FDA 483, 1999]
- Failure to establish test plan/protocol for approved hardware changes. [FDA 483, reported 2001]
- Lack of system checks before each program modification or correction becomes operational. [FDA Warning Letter, 1999].
- Change was not validated. [FDA 483, 2001]
- Inadequate standard operating procedures to ensure that records are included with validation documentation, are maintained and updated when changes were made. [FDA Warning Letter, 2001]
- Change control records were found signed off by the Quality Unit that had not been properly annotated in the code. [FDA 483, 2001]
- Supporting documentation requirements must be defined for corrective actions. [FDA Warning Letter, 1999]
- The firm failed to document review and approval of test records supporting program modification. [FDA 483, 2001]
- In managing change, personnel will receive their appropriate XXXX via an e-mail that has been sent from an e-mail distribution list. The firm has failed to implement controls to document that these distribution lists are maintained updated with the current approved listing of users. [FDA 483, 2001]
- There is no system in place to insure that parameter adjustments, which are executed during production runs, are made by authorized personnel. [FDA Warning Letter, 2002]
- Software "bug" that could result in erroneous release not scheduled for correction. [FDA 483, 2002]
- Computer program change requested to prevent shipping error has not been addressed. [FDA 483, 2003]
- Computer enhancement was identified as needed to correct labeling deviation but not implemented over one year later. [FDA 483, 2002]
- No record of review of software fix and correction of incorrect electronic records. [FDA 483, 2002]
- There was no evaluation of impact of software changes on other parts of the program. [FDA 483, 2003]

CONFIGURATION MANAGEMENT

Configuration management refers to the overall task of managing the use of varying versions of the various components (hardware, software, and documentation) that comprise a complex computer system. Both ISO 9001-3 (TickIT) and GAMP promote configuration management as a recommended and necessary discipline. The level of formality needed is greater for an operating system compared to, say, a system in its early development, but the principles are the same. The use of configuration management tools can considerably ease the effort required here, especially in the case of larger systems where the level of complexity grows exponentially rather than in a linear fashion.

Configuration management consists of the following activities:

- Configuration identification (what to keep under control)
- Configuration control (how to perform the control)
- Configuration status accounting (how to document the control)
- Configuration evaluation (how to verify that control)

Configuration management should be planned and conducted in accordance with defined procedures. This should include specified roles and responsibilities. Configuration management activities are normally specified with the Validation Plan or Project and Quality Plan, although the complexity of larger projects implies the desirability of a separate Configuration Management Plan.

Configuration Identification

Configuration Management begins with the system assembly. The task here is to identify and document the build configuration by ensuring that the mix of software, hardware, and documentation, all in their various versions, are unambiguously known and coordinated. Clearly, if this is not done, chaos rapidly ensues. It is important to be able to establish the exact composition of a particular system build that can act as a baseline or known reference point against which any subsequent changes or behavior can be referred. Key configuration management records include:

- Document index (approved documents including key documents provided by suppliers such as user manuals)
- Hardware unit index (clients, servers, communication interfaces, printers, etc.)
- Software program index (source code, executables, configuration files, data files, and third-party software such as operating system, library files, and drivers)

Configuration Control

All documents, hardware units, and software programs must be uniquely identified. It is not necessary to violate warranty seals in order to uniquely identify subcomponents. However, in situations where hardware units and software programs do not have a unique identification, physical labels and software header information can be added retrospectively. Unique identification should include the model number for hardware and the version number for software (e.g., MS Windows 2000 Professional Service Pack 2). Current approved versions of source code must correspond to current approved versions of software documents, object code, and test suites. All source code should have associated documentation.

Configuration Status Accounting

Documentation showing the status and history of configuration items should be maintained. Such documentation may include details of changes made, with the latest version and release identifiers.

Configuration Evaluation

A disciplined approach must be sustained to maintain the integrity of configuration management. It can be tempting to relax configuration management just to release resources and reduce costs. Consequential problems that often arise include:

- Only partial backups/archives of data made in a rush, or on insufficient storage media
- Items labeled unclearly or ambiguously labeled with poorly handwritten labels that cannot be understood by anyone other than the author
- Ambiguous version numbering nomenclatures, particularly for software where media might be labeled by date rather than by the version of the software carried
- Supplier's notification of serial numbers that do not match the actual serial numbers delivered
- Documentation not fully synchronized with system changes, often because documentation was accorded a lower priority than the physical implementation of a change

It is therefore recommended that the configuration status and practices should be regularly checked. Periodic reviews should include configuration management. It is important to demonstrate that:

- The Configuration Management Plan is up to date
- Recorded configuration is consistent with physical status
- Naming and labeling conventions are being followed
- Software version controls are being applied
- System is in the intended baseline state in accordance with defined milestones (e.g., for supplier testing, at installation, for user acceptance testing, and for use)
- Change Management is effective

The fundamental challenge to configuration management record keeping is whether it can make a full system rebuild possible by relying solely on these records. A good test is to assess the measure of confidence the responsible person has that the system could be successfully restored on a first attempt. At a practical level, this capability is the foundation of disaster recovery, directed at the effective support of business continuity plans.

Recent Inspection Findings

- The firm has failed to establish an overall revision control system for the program throughout its software life cycle. [FDA 483, 2001]
- The *[computer system]* is not validated in that … configuration management: The firm failed to document all sites, departments or connections on the network … The firm has failed to document external program interfaces … The firm has failed to define or describe the various uses for the development, test, and production environments. [FDA 483, 2001]
- Control over the XXXX is via configuration management of the customizable functions available as part of the proprietary base software application. Evaluation of this configuration management found
  1. Original baseline configuration documentation generated on June xx, 1997 as part of the configuration management plan was not reviewed/approved by Quality Assurance.
  2. An audit report dated June xx, 1997 evaluating the original baseline configuration documentation was found not reviewed/approved by Quality Assurance.
  3. Changes to the configuration of XXXX were found being made with no oversight or review by the Quality Assurance unit. [FDA 483, 1999]

SELF-INSPECTIONS (INTERNAL AUDITS)

Self-inspections, also known as internal audits, are a fundamental activity of competent quality assurance. They typically focus on reviewing validation documents and the SOPs used to generate them. Guidance is not usually audited unless it is effectively being used as the procedure for work.

The PIC/S harmonized computer inspection guide contains aide memoires that can be used as input to develop a self-inspection checklist.[6] Such checklists can be used to examine any aspect of working practices, assess the level of GxP compliance of processes, including computer systems, with a view to identifying poor practices and opportunities for improvement. They can also be used with equal beneficial effect in non-GxP areas.

Personnel independent of the work practices being examined should conduct these audits. Independence implies the ability to demonstrate a measure of impartiality. It does not necessarily mean that the person conducting the self-inspection is from a separate department or function. A peer review is perfectly acceptable.

A report describing the self-inspection and its observations should be produced, unless a dispensation has been specifically not required and given within the terms of reference. Observations from internal audits should be precise and objective. Documenting subjective opinions should be avoided. Closure of actions should be tracked.

Internal audits are not usually subject to regulatory scrutiny without due cause, and reports of such should not be presented during a regulatory inspection without good reason. If an inspector does ask to see evidence of self-assessments being conducted, the pharmaceutical or healthcare company should consider sharing with the inspector the schedule of self-inspections recently completed, and leave it at that.

MANAGING DEVIATIONS

A nonconformance may be discovered during an internal audit, a regulatory inspection, through a customer query, or by chance. It may be caused by failure to follow procedural controls, or by a failure of the procedural controls themselves, or by an ambiguity or lack of detail in the documents and records supporting the procedural controls. Examples of nonconformances include:

- Computer systems that do not behave in the necessary manner (e.g., electronic record and electronic signature requirements)
- Validation activities that do not conform to defined procedures
- Validation documents that do not conform to defined procedures
- Defined procedures that do not fulfill regulatory requirements

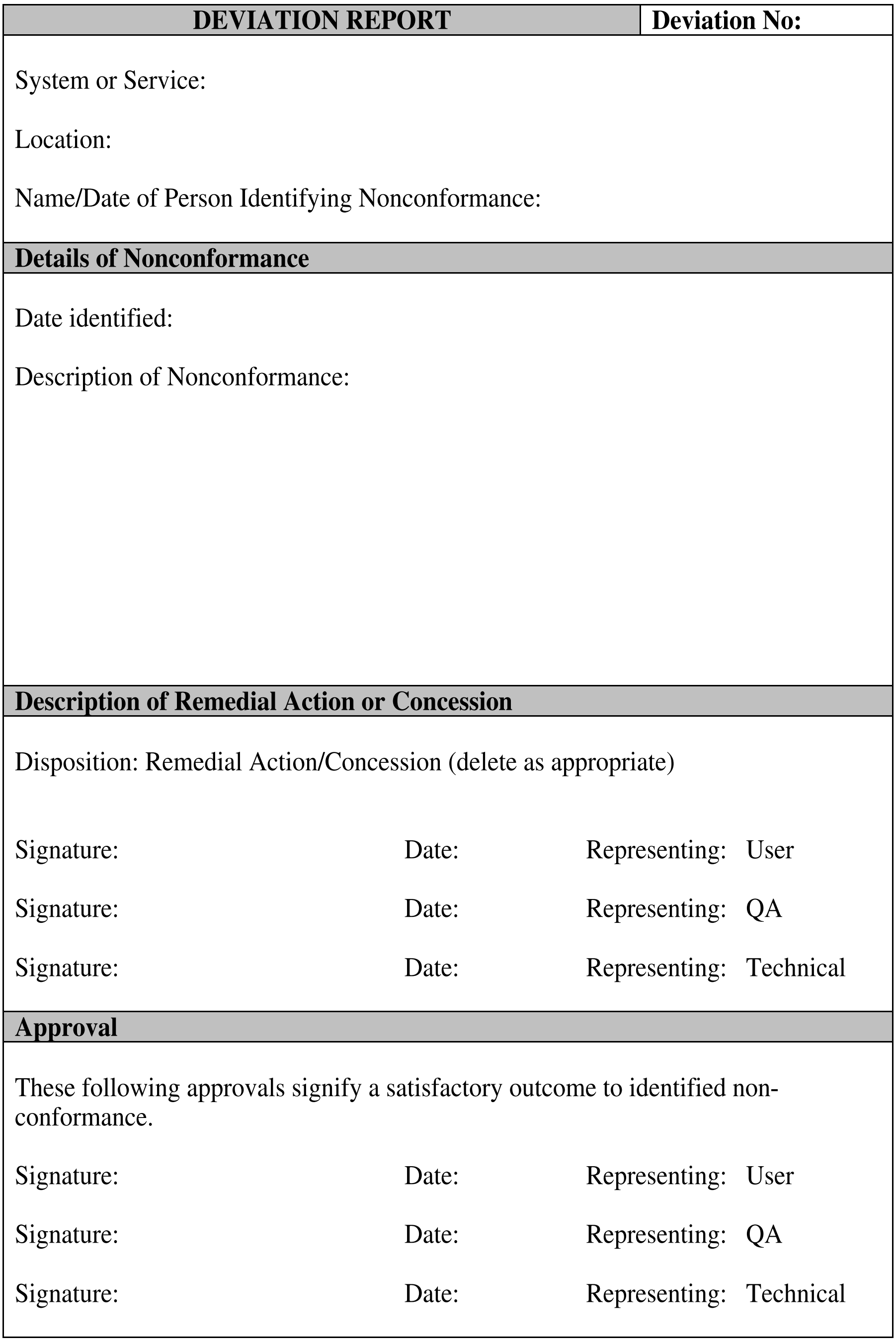
When a nonconformance is discovered, it must be reported to the manager responsible for the computer system, service, or document. Depending on the significance of the nonconformance, QA management may also need to be informed.

Deviation Reports should be prepared to describe the nonconformance, analyze the nature of the deviation, and define how the deviation is being addressed. The criticality of the deviation will determine appropriate controls:

- Deviations that directly impact GxP processes, i.e., those that affect the quality, efficacy, or safety of pharmaceutical and healthcare products, will require root cause remediation.
- Indirect impact of GxP processes can be addressed through process change (sometimes referred to as work-arounds) involving modifications to SOPs, or by avoiding the impact in the first place through modifying the GxP process.
- Those deviations that do not impact GxP processes, i.e., those that cannot impact pharmaceutical and healthcare products, do not necessarily require remediation and can be accepted without corrective action so long as there is no other key operational deficiency.

Deviation Reports must document the approval of the remedial actions (or concession to accept the deviation without remedial action) and justify closure of the deviation, explaining where appropriate ongoing controls to stop it from happening again. An example Deviation Report is shown in Figure 4.3. Deviation Reports associated with GxP impacting nonconformance require the signature and approval of the QA organization. The reports with supporting evidence should be retained as part of the computer system documentation. How deviations are handled is a good indicator of how well an organization understands validation, and pharmaceutical and healthcare companies should not be surprised if deviation records are requested for review during regulatory inspections.

The management of deviations and compliance issues is discussed further in Chapter 6 where the use of Project Compliance Issue Logs and RAID Log (Risk, Actions, Issues, Decisions) are introduced.



DEVIATION REPORT	Deviation No:

System or Service:

Location:

Name/Date of Person Identifying Nonconformance:

Details of Nonconformance

Date identified:

Description of Nonconformance:

Description of Remedial Action or Concession

Disposition: Remedial Action/Concession (delete as appropriate)

Signature:	Date:	Representing:	User
Signature:	Date:	Representing:	QA
Signature:	Date:	Representing:	Technical

Approval

These following approvals signify a satisfactory outcome to identified nonconformance.

Signature:	Date:	Representing:	User
Signature:	Date:	Representing:	QA
Signature:	Date:	Representing:	Technical

FIGURE 4.3 Example of Deviation Report.

REFERENCES

1. ACDM/PSI (1998), *Computer Systems Validation in Clinical Research: A Practical Guide*, Version 1.1, December.
2. Mikkola, C. and Rios, M. (2002), Regulatory Compliance Training: Who Needs It? *IT Innovations* (supplement to *Pharmaceutical Technology*).
3. Wingate, G.A.S. (1997), *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice*, Interpharm Press, Buffalo Grove, IL.
4. Koelman, E., De Jong, K., and Piket. K. (2000), Maintenance and Support of Validated IT Systems, in *Validating Corporate Computer Systems: Good IT Practice for Pharmaceutical Manufacturers* (Ed. G. Wingate), Interpharm Press, Buffalo Grove, IL.
5. FDA (1997), *General Principles of Software Validation: Guidance for Industry*, Draft Guidance Version 1.1., June.
6. Pharmaceutical Inspection Co-operation Scheme (2003), *Good Practices for Computerised Systems in Regulated GxP Environments*, Pharmaceutical Inspection Co-operation Scheme (PIC/S), Pharmaceutical Inspection Convention, PI 011-1, Geneva, August.

APPENDIX 4A
EXAMPLES OF DEFICIENT DOCUMENTARY EVIDENCE

- Failure to use standardized document formats
- Incomplete definitions
- Constraints not cited
- Formulae inconsistencies
- Inappropriate or inconsistent "<" and " >"
- Legends with inconsistent or misleading scales
- Excessive changes
- Different corrections not distinguished
- Not reporting significant changes
- Standard reports and forms not used (raw data recorded informally)
- Illegible writing
- Raw data records not available
- Inconsistent dates
- Inconsistent units of measure
- Lack of double checking for accuracy
- Not reporting all adverse reactions
- Problems and deviations not fully reported
- Unusual or unexpected recorded results
- Records retained informally
- Inappropriate procedures
- Deficient procedures
- Staff responsibilities not defined
- Training records not up to date
- Poorly organized documents and records
- Hard-to-follow documents and records
- Report conclusions that seem too good to be true
- Reports with exaggerated claims
- Reports with political half-truths
- Reports with incorrect absolute terms (all, every, none, never, etc.)

APPENDIX 4B
EXAMPLE SELF-INSPECTION CHECKLIST

- Determine the critical control points (base investigation on FMEA or other hazard analysis technique). Examples would be:
  - Pasteurization
  - Sterilization
  - pH control
  - Temperature control
  - Cycle timing
  - Control of microbiological growth
  - Quality status of materials and products
  - Record keeping
- For those critical control points controlled by computerized systems determine if failure of the computerized system may cause drug adulteration.
- Identify computerized system components including:
- Hardware inventory
  - Input devices
  - Output devices
  - Signal converters
  - Central Processing Unit
  - Distribution system
  - Peripheral devices
- Hardware
- Obtain a simplified drawing of the computerized system (covering major computer components, interface, and associated system/equipment). For computer hardware determine the manufacturer, make, and model number.
- Software inventory
  - Inventory of files (program and data)
  - Documentation
  - Manuals
  - Operating procedures
- Software

  For all critical software determine:
  - Name
  - Function
  - Inputs
  - Outputs
  - Set-points
  - Edits
  - Input Manipulation of Date
  - Program overrides
  - Version control
- Who developed the software (standard, configured, customized, bespoke)?
  - Software security to prevent unauthorized changes
  - Computerized systems input/outputs are checked
  - Obtain simplified drawing of overall functionality of collective software within computerized systems
- Data
  - What data are stored and where?

  - Is data distributed over a network — how is it controlled?
  - How is compliance to electronic record regulations achieved?
  - How is data integrity verified?
- Personnel
  - Type (developer, user, owner)
  - Training records
- Observe the system as it operates to determine if:
  - Critical processing limits are met
  - Records are accurate
  - Input is accurate (sensor or manual input)
  - Time keeping is accurate
  - Personnel are trained in systems operations and functions
- Determine if the operator or management can override computer functions. How is this controlled?
- How does the system handle deviations from set or expected results?
- Check all alarms, calculations, algorithms, and messages.
- Alarms
  - Types (visual, audible, etc.)
  - Functions
  - Records
- Messages
  - Types (mandate action?)
  - Functions
  - Records
- Determine the validation steps used to insure that the computerized system is functioning as designed.
- Was the computerized system validated upon installation?
  - Under worst case conditions?
  - Minimum of three test runs?
- Are there procedures for routine maintenance?
  - User manual
  - Vendor-supplied manual
  - Third-party support manual
  - Management manual
- Does the equipment here meet the original specifications?
- Is validation of the computerized system documented?
- How often is system:
  - Maintenance performed
  - Calibrated
  - Revalidated
- Check scope and records of any service level agreements.
- Are there procedures for revalidation? How often is revalidation conducted?
- Are system components located in a hostile environment that may affect their operation (ESD, RFI, EMI, humidity, dust, water, power fluctuations)? Are system components reasonably accessible for maintenance purposes?
- Determine if the computerized system can be operated manually. How is this controlled?
- Automated CIP (cleaning in place)
  - How is automation verified?
  - Documentation of CIP steps
- Automated SIP (sterilization in place)
  - How is automated sterilization verified?

  - Documentation of SIP steps
- Shutdown procedures
  Does firm use a battery backup system?
  Is computer program retained in control system?
- What is the procedure in the event that power is lost to computer control system?
  Have backup and restore procedures been tested?
  Is there a documented system for making changes to the computerized system?
  Is there more than one change control system (hardware, software, infrastructure, networks)? For each of these challenge as follows:
  - The reason for the change
  - The date of the change
  - The changes made to the system
  - Who made the changes
- How do they interface? Challenge change history, verify audit trail?
  What are the auditor's impressions of:
  - Presentation of validation
  - State of documentation
  - State of compliance
  - Maintaining validation
  - Requirements for revalidation

5 Prospective Validation Project Delivery

CONTENTS

The various validation approaches promoted within the pharmaceutical and healthcare industries by GxP regulators and industry groups adopt the same basic approach:

- Define what is to be done (plan).
- Define how to do it (specification, procedures, and resources).
- Do it, controlling any changes (Change Control).
- Establish that the end result was what was originally intended (verification).
- Provide evidence demonstrating this (audit trail).

This chapter presents the set of life-cycle phases summarizing the project approach typically adopted within the pharmaceutical and healthcare industries. These life-cycle phases may be known by alternative names within different organizations, as there are no generally accepted naming conventions or groupings of phases yet throughout the industry. It is important, however, that all the activities covered in this chapter are included in any alternative scheme.

CHARACTER OF APPLICATION

The features of a computer system, and hence its validation requirements, can be described in terms of its hardware and software. The GAMP Forum has defined five categories of software found in computer systems. These categories are intended to be comprehensive, so it should not be possible for any software to fall entirely outside of them. They are as follows:

- **GAMP Category 1 Software: Operating Systems**
  This category defines established commercial operating systems. Examples include OS/2 and Microsoft Windows. Regrettably, upgrades can be a mixed blessing, for while correcting defects and delivering enhancements they can at the same time have a serious impact on overall system performance and security. Because of this, regression testing of the application using the new version of the operating system cannot be skipped when the system is upgraded.
- **GAMP Category 2 Software: Firmware**
  This category defines the firmware embedded on chips in instruments, controllers, and computer peripherals such as printers. Because of its form, firmware is not usually accessible by users, although for some kinds of firmware there is a trend nowadays to make such code upgradeable using proprietary so-called *flash upgrade* installation tools together with a new release of code. Examples of pharmaceutical and healthcare firmware include those in mass balances, pH meters, spectrophotometers, bar code readers, weigh

scales, bar code scanners, and three-term controllers. Upgrades to nonconfigurable commercial firmware can rely on IQ and calibration. Upgrades to configurable commercial firmware will require regression testing in addition to IT and calibration. Custom-built firmware should be managed as Category 5 software.

- **GAMP Category 3 Software: Standard COTS Software Packages**
  This category defines Commercial Off-The-Shelf (COTS) software packages. Any configuration to which such software would be subjected in a pharmaceutical and healthcare operation is normally limited to operating parameters and configuration of the system environment parameters (e.g., file names, directory/folder structures). Examples include statistics packages and software for the use and control of laboratory instruments such as HPLC. Regression testing is required when such software is upgraded. Commercial software packages that lack wide exposure in the market must not be recognized as market-tested, and should be managed as Category 5 software.
- **GAMP Category 4 Software: User Custom-Configurable COTS Software Packages**
  This category defines commercial software packages whose configuration makes use of elements of custom code such as application macros and database scripts. Configuration of the system environment is usually required. Examples of commercial configurable software packages of this kind include SCADA, DCS, MES, LIMS, MRPII, and some test equipment. The core software package and its configuration should be managed as Category 3 and Category 5 software, respectively.
- **GAMP Category 5 Software: Custom (Bespoke) Application Software**
  This category defines software written entirely to meet the exclusive requirements of a single user/company or a small group of users. Because of this, it is likely to be updated frequently. Examples of custom software development include code for data migration, code for reports, and code for interfaces. Software from another category that has been customized should also be managed as Category 5 software.

Most computer systems will have a number of software components falling into several of these categories, as illustrated in Table 5.1.

Hardware can also be divided into categories representing different validation requirements. The GAMP 4 Guide introduced two basic categories of hardware: standard hardware and custom (bespoke) hardware.

COTS Products

Nowadays most software and hardware is based on purchased COTS products rather than being bespoke/custom built. Pharmaceutical and healthcare companies (the COTS product users) are accountable to the regulatory authorities for ensuring that the product development methodologies used by the COTS developer are of a sufficient degree of capability maturity and adequate for the intended use of the COTS product. If the supplier can provide information about the development of its COTS product, then this can be used as the basis of user validation.

COTS products have the same basic validation requirements as any other type of software. Typically, supplier audits are expected for critical and complex applications as discussed in Chapter 7. If an audit is conducted and progress with any significant corrective actions cannot be proven, then suitable alternative suppliers or sources of software should be sought, including bespoke/custom developments.

A common problem with COTS products is that access to product development documentation may not be available. The supplier may refuse to share its proprietary information, or it may be unavailable for some other reason. In such circumstances pharmaceutical and healthcare companies are expected by the regulatory authorities to compensate with additional black box testing to establish with sufficient confidence that the COTS products meet user needs. (Black box testing is

TABLE 5.1
Applying GAMP Software Categories to Typical Computer Systems

		GAMP Software Categories[a]				
Type of System	**Typical Applications**	**1**	**2**	**3**	**4**	**5**
Computerized Analytical Laboratory Equipment	pH Meter		✓			
	High Performance Liquid Chromatography (HPLC)	✓	✓		✓	
	Chromatography Data System (CDS)	✓			✓	✓[e]
Process Control and Monitoring Systems	Field Instrumentation		✓		✓[c]	
	Programmable Logic Controller (PLC)	✓			✓	✓
	Supervisory Control and Data Acquisition (SCADA)	✓			✓	✓
Spreadsheet and Database[b]	Spreadsheet Application				✓[d]	✓[e]
	Database Application				✓[d]	✓[e]
Corporate Computer Systems	Laboratory Information Management System (LIMS)	✓			✓	✓[e]
	Manufacturing Resource Planning (MRPII)	✓			✓	✓[e]
IT Infrastructure and Services	Desktop Environment	✓		✓	✓	
	Communication Network	✓		✓	✓	

[a] GAMP 4 Guide.
[b] Excludes resident Personal Computer or workstation.
[c] "Fieldbus" instrumentation is on-line user configurable.
[d] Nearly all standard software products all require some form of configuration, at least in the form of set-up parameters or application macros.
[e] Typically interfaces and bespoke reports.

functional testing of software often based on verifying expected outputs from defined inputs without having any real knowledge of how the software works. The imaginary box is black because it cannot be opened to inspect its inner workings.) Commercial software products may have "bug-lists," user manuals, and product specifications that can be compared to the user requirements to structure such black box testing effort.

The FDA recognizes that user black box testing may be impractical for some COTS products like compilers, linkers, editors, software development tools, and operating systems. The proper operation of these COTS products may be satisfactorily inferred by other means such as their independent certification by accreditation bodies.[1] The FDA has also suggested that COTS operating systems need not be validated by a separate exercise as long as the validation of the application software addresses operating system characteristics upon which the application is dependent. Such might include maximum loading conditions, file operations, handling of system error conditions, and memory constraints for the operating system.[1]

Open Source Software

Open source software (also known as freeware or shareware) is increasingly being incorporated into COTS products. Open source software is developed by informal communities who claim no ownership and refute any accountability for the code. Features and bug fixes emerge out of uncoordinated custom developments of freely available, uncontrolled copies of the source code. There is usually no formal quality umbrella for software development and support; hence it is very difficult to demonstrate that such software is fit for purpose. Reverse engineering development documentation from the code and conducting comprehensive testing are likely to be extremely expensive. The use of open source software should therefore be avoided. If it cannot be avoided (e.g., embedded in COTS products), intensive black box testing should be undertaken commensurate with the criticality of the application.

APPROACH TO VALIDATION

The specific project life-cycle model adopted does not matter as long as it covers planning, requirements and design, implementation, and testing.[2] The constituent phases making up the life cycle must be clearly defined with entry and exit criteria for each phase and appropriate verification procedures to ensure the controlled completion of constituent phases. Figure 5.1 illustrates the project life cycle used in this book. Although represented as a sequence of phases, many activities can be conducted in parallel without compromising compliance. Indeed, where

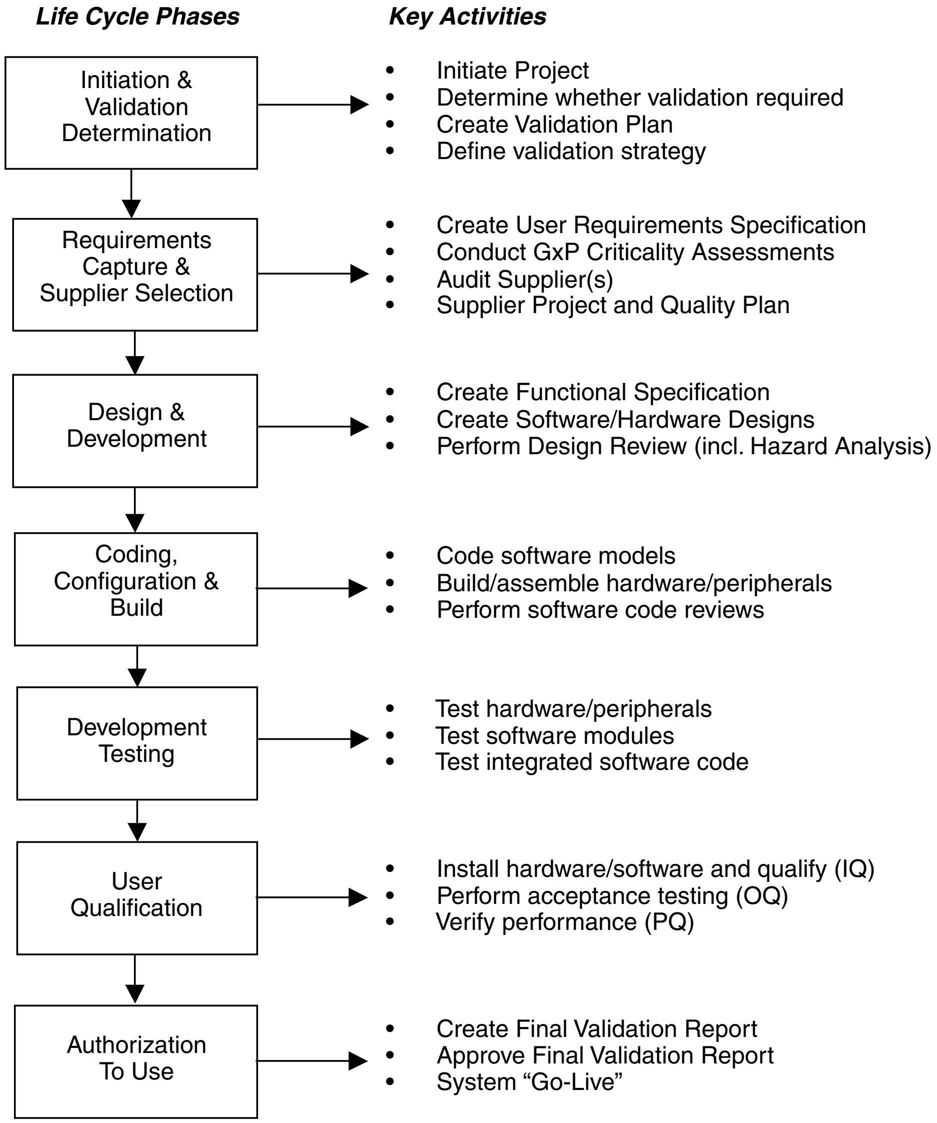

Life Cycle Phases
Key Activities
Initiation & Validation Determination
• Initiate Project
• Determine whether validation required
• Create Validation Plan
• Define validation strategy
Requirements Capture & Supplier Selection
• Create User Requirements Specification
• Conduct GxP Criticality Assessments
• Audit Supplier(s)
• Supplier Project and Quality Plan
Design & Development
• Create Functional Specification
• Create Software/Hardware Designs
• Perform Design Review (incl. Hazard Analysis)
Coding, Configuration & Build
• Code software models
• Build/assemble hardware/peripherals
• Perform software code reviews
Development Testing
• Test hardware/peripherals
• Test software modules
• Test integrated software code
User Qualification
• Install hardware/software and qualify (IQ)
• Perform acceptance testing (OQ)
• Verify performance (PQ)
Authorization To Use
• Create Final Validation Report
• Approve Final Validation Report
• System "Go-Live"

FIGURE 5.1 Key Validation Activities.

TABLE 5.2
Software Categories (Based on GAMP 4)

Category	Software Type	Validation Approach
1	Operating System	Record version (including any service pack). The Operating System will be challenged indirectly by the functional testing of the application.
2	Firmware	Record version of nonconfigurable COTS firmware and calibrate as necessary. Record version and configuration of configurable COTS firmware. Calibrate as required and verify operation against user requirements. Manage custom-built firmware as Category 5 software.
3	Standard COTS Software Packages	Record version and any configuration of environment. Verify operation against user requirements. Consider auditing the supplier for critical and complex applications.
4	User Custom-Configurable COTS Software Packages	Record version, any parameter configuration, and any configuration of environment. Verify operation against user requirements. Normally assess (audit) software development capability maturity of the package supplier for complex and critical applications. Manage any bespoke programming (e.g., macros) as Category 5 software.
5	Custom (Bespoke) Application Software	Assess (audit) software development capability maturity of supplier and validate complete computer system.

TABLE 5.3
Hardware Categories (Based on GAMP 4)

Category	Hardware Type	Validation Approach
1	Standard Hardware	Record the model, version number, and serial number where available of preassembled hardware. Retain hardware data sheets and other supplier specification material. Document hardware configuration details. Verify installation and performance of hardware components.
2	Custom (Bespoke) Hardware	Manage standard hardware components as Category 1 hardware. Prepare design specification including any hardware configuration. Verify installation and performance of hardware components. Assess (audit) hardware development capability maturity of supplier.

appropriate, the parallel execution of activities should be encouraged to help de-bottleneck project critical paths.

The validation life-cycle approach applies to both “in-house” developed and purchased computer systems. Supplier responsibilities are indicated later in this chapter. Suppliers include internal development or support groups, external vendors, and outsource organizations.

Emphasis within the life cycle will change depending on whether computer hardware is bespoke or standard and also on the mix of software categories in which the application software falls. Table 5.2 and Table 5.3 outline the preferred validation approach toward different categories of software and hardware based on the risk the various categories of software pose.

While this approach may seem simple, it becomes more complex when validating an application containing software in multiple categories. Most computer systems will contain software in multiple software categories as it is very unusual to find a system that is made up of software falling into only one category. It is not usually practical or desirable to validate each item of software independently. Rather, validation typically validates the software collectively as a complete application. A well-organized and streamlined approach is necessary.

TABLE 5.4
Tools Validation

Type of Software Tool	Compliance Requirement	Examples
Computer Aided Software Engineering used in context of supporting creation, ongoing maintenance, and/or retrieval of validation records	Validation required	Configuration management supporting IQ records; automated testing tools supporting OQ/PQ records; tools applying approvals to validation evidence; change management tools supporting change control records
System Software	Validation by inference	Compilers, communications interface driver software, Acrobat Writer
Incidental Use	Validation not required	Word processors, Microsoft Project, Network Performance, Computer Aided Software Engineering not requiring validation as above (e.g., debuggers, static code analysis, process modelling, and PLC programming)

Software Tools

Validation requirements for software tools are summarized in Table 5.4. Software tools may require validation even though they might be considered supplementary to the application software.

Software used to automate any part of the software development process, its maintenance, or any part of the quality system must be validated.[3] Specifically, software tools that generate code, provide quality diagnostics, automate testing, or provide software maintenance facilities such as configuration management need to be assessed in line with their software categories. The use of software tools must be described in the application documentation. Tools that create electronic records and manage them also need to be assessed for compliance with applicable regulatory requirements for electronic records and electronic signatures.

Software tools supporting the functionality of applications should also be validated. Specific validation is not required because validation is inferred by validating the correct operation of the application itself. Examples include operating systems (GAMP Category 1 Software) that should be validated as GAMP Category 1 software.

Software tools do not require validation if they are incidental to the creation of regulated records that are subsequently maintained in traditional paper-based systems. Examples include word processors that essentially act as typewriters, and project scheduling tools that do not have a regulatory dimension. Software that provides record storage and retrieval facilities for incidental records would be treated like traditional “file cabinets.” Overall reliability and trustworthiness would derive primarily from well-established and generally accepted procedures and control for paper records.

Managing Change

An important matter when choosing a life cycle is the relative cost of making changes to software as the cycle progresses. The relative cost of correcting errors is shown in Figure 5.2. It is very important to clarify incomplete or ambiguous information as soon as reasonably practical because the longer it is left the more expensive it will be to correct if meantime a wrong assumption has crept in. The later the changes are made the more they are likely to cost.

Experience suggests that poor design and programming errors account for up to one third of system malfunctions.[5] This emphasizes the importance of not only avoiding these problems in the first place but also discovering these problems as early as possible. In particular, the use of design reviews is promoted to catch errors early and thereby reduce the cost of their correction when compared to only discovering the errors during testing. The use of software inspections (either

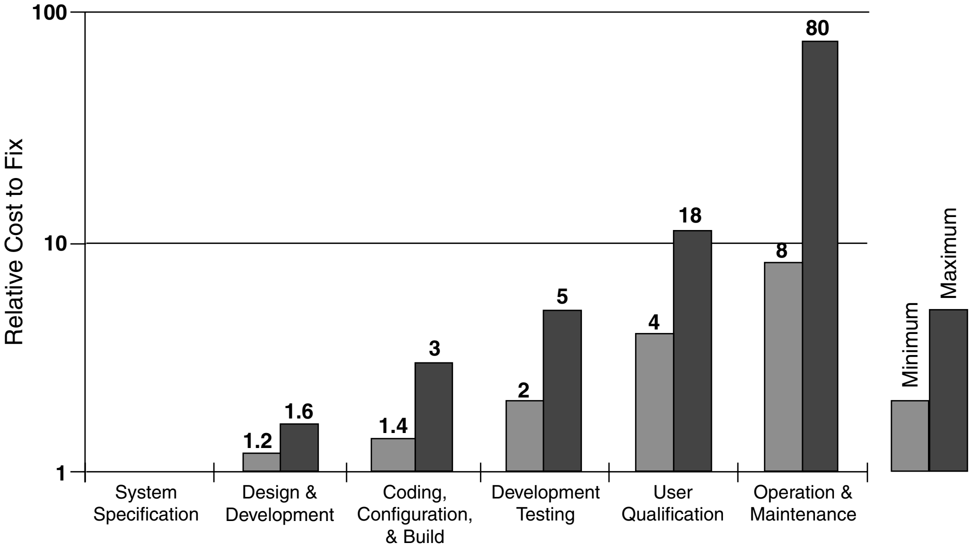


FIGURE 5.2 Relative Cost of Error Correction. (Based on David Begg Associates (2001), *Computer Systems and Automation Quality and Compliance,* York (U.K.), June 5–7.)

conducted formally, or via less formal code walkthroughs) could also save money by the early exposure and correction of errors. It is estimated that projects implementing effective design reviews and software inspections can reasonably expect the overall project effort to shrink by about 10% when compared to projects paying little or ineffective attention to these questions.[6] Not only is the project cheaper to run but the outcome (the computer system) is more predictable and has higher quality with all the benefits that this implies for validation and compliance.

Another important feature of change is the changing influence between the pharmaceutical and healthcare companies and the supplier during the project. At the beginning, the company has enormous influence, which is right and proper as this is where the definition and the direction of the project originate. However, the pharmaceutical or healthcare company's influence quickly declines as the project progresses because of the increasing relative cost of change. The supplier soon has a significant influence because the supplier largely dictates whether or not changes can be implemented within the constraints of the project (time, functionality, and cost). This transition of influence is illustrated in Figure 5.3. Practitioners need to ensure that a project does not inadvertently fall into the position of becoming resistant to the implementation of critical changes because time and budgets were frittered away earlier in the project in the pursuit of less important changes.

Validation Roles

The diagram for the life cycle presented in Figure 5.1 is complemented in Figure 5.4 with one showing the main roles of System Owner/User, Developer, and Quality and Compliance defined in Chapter 3. Each role is described below.

The System Owner/User role is responsible for defining and approving requirements. It should be possible to describe in overview the system to be implemented from this information. The System Owner/User should then agree with Quality and Compliance what functionality within the computer system is GxP critical. This is used to help select the supplier to develop the system. The System Owner/User should lead a Design Review to verify that what is being developed meets the requirements, with feedback to the design and development group(s) as required. The User Qualification Process should take account of GxP Assessment of critical functions, the capability

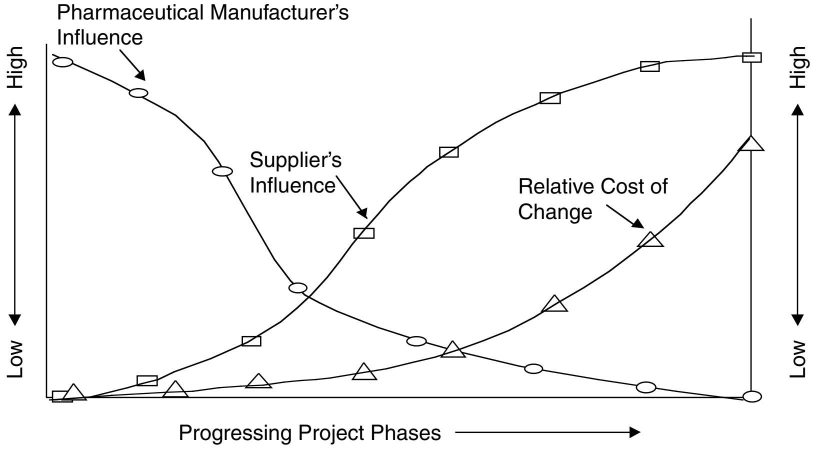


FIGURE 5.3 Changing Project Influences.

of the supplier from the supplier selection process, and any recommendations from the Design Review. The System Owner/User should approve User Qualification and the final Validation (Summary) Report.

The Developer role should start with the drafting of an agreed contract of supply. The scope of supply may alter as a result of the supplier selection process. Design and development, system build, coding, and configuration (including Software Inspection/Source Code Review) follow. The computer system is functionally tested (with traceability) to confirm that the design intentions (and in turn the user requirements) are achieved. Satisfactory testing is used finally to authorize release of the system for distribution. The System Owner/User will then qualify the system, using as much evidence from Development Testing Process as possible to reduce the User Qualification effort required.

The Quality and Compliance role traditionally starts with the preparation and approval of a Validation Plan and associated specification of a validation strategy. Both will be influenced by the User Requirements Specification and the output from the GxP Assessment. A Supplier Audit may be required depending on the level of criticality and degree of custom software involved (discussed further below). Predelivery Inspection should be conducted as needed to follow up on any Supplier Audit issues and to confirm that the supplier is implementing its own quality management system as required on the development of the computer system. Quality and Compliance should further participate in the Design Review to verify that what is being developed meets validation requirements. User Qualification should also be reviewed and approved together with authorization of the Validation (Summary) Report concluding project validation.

CHOOSING AN APPROPRIATE LIFE-CYCLE METHODOLOGY

The validation activities presented in this book can be implemented in accordance with a number of life-cycle methodologies. The "waterfall model" life cycle is the most rudimentary approach and basically cascades the activities presented in Chapter 6 through Chapter 12 inclusive. The ordered sequence of the "waterfall" life cycle works well with tightly defined and understood requirements. Unfortunately, in the real world, although we might like to think otherwise, most projects are not tightly defined and understood.

The pharmaceutical and healthcare industry has adopted the "V-Model" life cycle for computer system projects.[7] The validation activities are presented as a V life cycle in Figure 5.5. The V-Model was developed to promote planning and designing in anticipation of testing. Phases of the life cycle are conducted in a controlled sequence. In theory the current phase must be completed

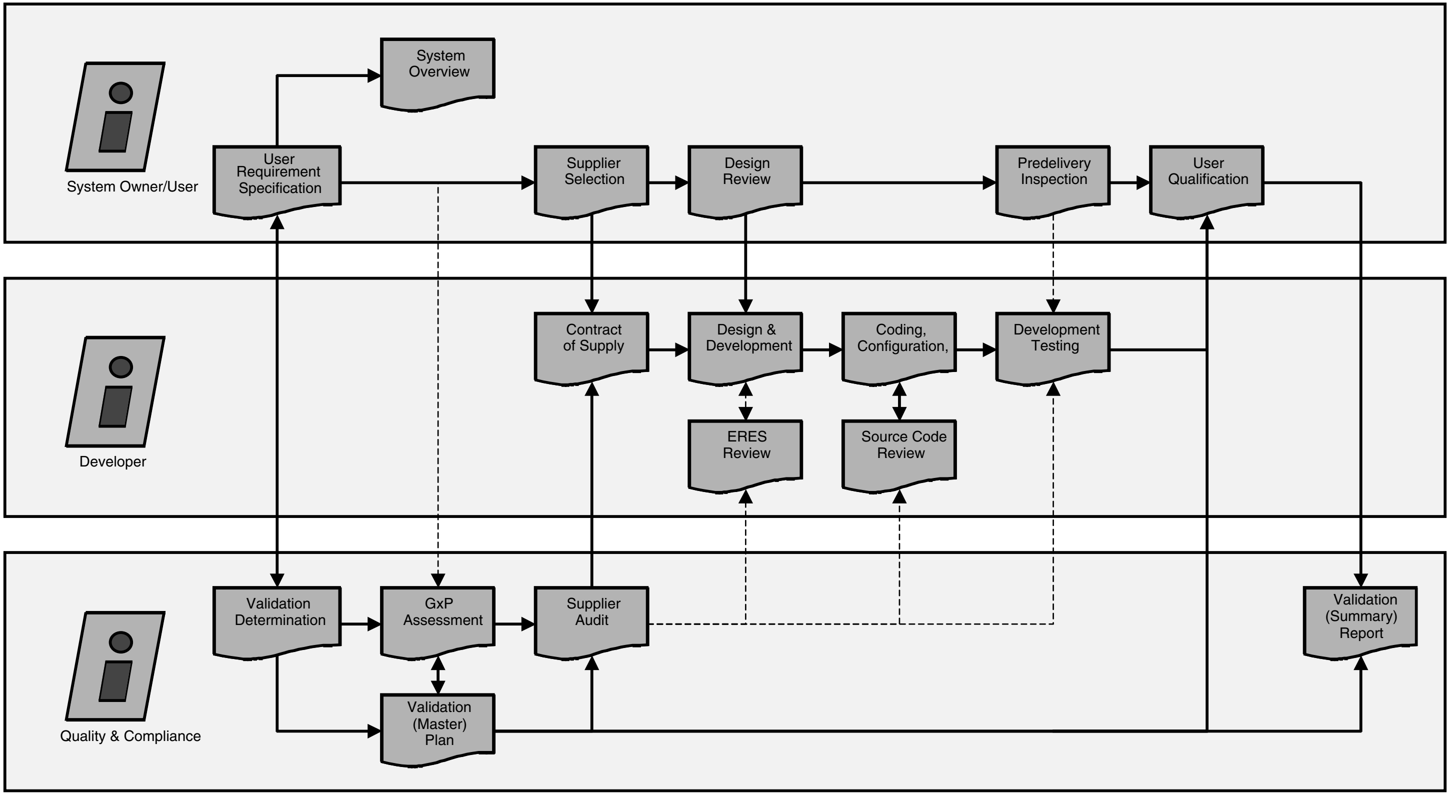


FIGURE 5.4 Role Activity Diagram for V-Model.

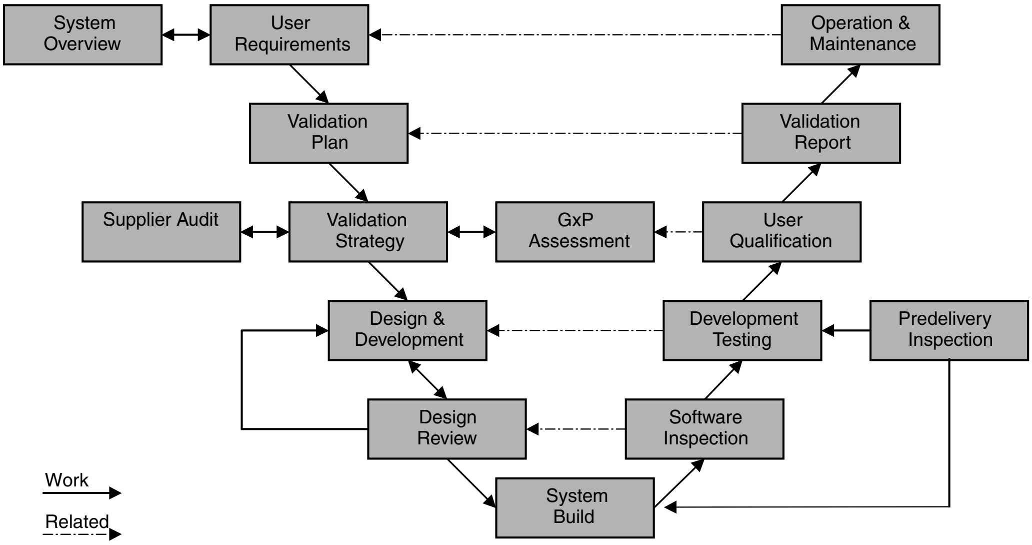


FIGURE 5.5 V-Model Life Cycle for Computer Validation.

before progressing to the next phase. In practice, while the edges may blur a little between phases, the order of finalizing key documents cannot be broken. User qualification (acceptance testing) should only be conducted after coding, configuration, and build. Design should not be concluded without a finalized user requirements specification.

The strengths of the V-Model have been summarized as follows:[8]

- The model emphasizes planning for verification of the computer system in the early stages of the project.
- The model encourages verification of all deliverables, not just the system.
- The V-Model encourages definition of the requirements before designing the system, and it encourages designing software before building it.
- It defines the computer system that the project development process should deliver; each deliverable must be testable.
- The V-Model enables project management to track progress accurately; the progress of the project follows a timeline, and the completion of each phase is a milestone.
- It is easy to use (when applied to a project to which it is suited).

Weaknesses of the V-Model have also been summarized as follows:[8]

- It does not handle concurrent events very well.
- It does not handle iterations of phases very well.
- The model is not equipped to handle dynamic changes in requirements throughout the life cycle.
- The requirements are tested too late in the cycle to make changes without affecting the schedule of the project.
- The model does not contain project risk analysis activities.

The structure and control brought by the V-Model can be quite frustrating to project managers up against challenging deadlines. Accelerated development is often thought of as the solution but there is no "silver bullet." Rapid Application Development (RAD) and Prototyping allow management and users to "see" the system requirements as they are developed. It is very useful when requirements for a new system are not fully defined at the beginning of a project. Clarifying requirements is especially important for "proof of concept" projects. It has been proven well suited for highly interactive systems, first-of-a-kind systems, decision support systems, and medical diagnosis. Such accelerated development can be habit forming and may go on too long or be applied inappropriately to projects that do not warrant such an approach. Undisciplined programmers can fall into a "code and fix" cycle (akin to relying on breadboards for hardware "build and fix") that does not embody the quality assurance principles expected for GxP applications.

PROJECT INITIATION AND VALIDATION DETERMINATION

Inputs:
  Project Scope (Outline Specification)
Outputs:
  Validation Determination
  Validation (Master) Plan
  Project and Quality Plan

PROJECT SCOPE (OUTLINE SPECIFICATION)

The Project Scope provides an executive summary justifying the purpose and benefits of the proposed computer system. Particular project management risks will normally be identified at this stage so that key controls to help ensure project success can be planned.

An Outline Specification is typically produced as part of a Project Scope to sanction a project. The Outline Specification should not be specific to hardware and software products but should provide sufficient information for a Validation Determination Document to be prepared.

The Project Scope is not maintained after the project is completed. The Outline Specification, meanwhile, must not be lost. It should be carried forward to form the basis of a System Overview that is maintained during the operational life of the computer system.

VALIDATION DETERMINATION

All computer systems should be assessed as early as possible to determine whether or not validation is required. Validation Determinations should be periodically reviewed against changes in pharmaceutical and healthcare regulatory requirements, or use of the system, and updated as necessary. Some pharmaceutical and healthcare companies include the Validation Determination within the Validation Plan. The benefit of keeping this as a separate document is that it can be used to justify why some computer systems are not validated.

During regulatory inspections the Validation Determination can be presented with a Validation Certificate to demonstrate that a particular computer system has been validated without delving into detail.

VALIDATION (MASTER) PLANNING

Validation Plans record standards, methods, and personnel involved to assure quality through the system development life cycle and to establish the adequacy of the performance of the computer system. The term Validation Master Plan is typically used for large or multiple computer system validation projects. Validation planning should be initiated at the earliest practicable opportunity and may be reviewed and updated through subsequent stages of the project.

The size of the Validation Plan should be commensurate with the project complexity. Any planned amalgamation or split of documentation to fit the size and complexity of a computer system should be defined here. Review and approvals should be defined either within referenced procedures or specifically within the plan.

The information contained in User Requirements Specifications is often used to help determine the basic approach to validation to be taken for individual projects. Guidance on the scope of validation required for different types of computer systems is defined at the beginning of this chapter. A rationale must always be given supporting any validation decisions made during a validation project in the form of a separate document or included within the Validation Plan.

During regulatory inspections, the Validation Plan can be presented with a Validation Report or Validation Summary Report to demonstrate that a particular computer system has been validated. These documents provide a regulator with additional detail to the Validation Determination document and Validation Certificate but without digressing into the supporting validation evidence.

If the computer system is relatively small and contained entirely within a stand-alone piece of equipment, the Validation Plan for the computer system may be embodied within the equipment's overall Validation Plan.

Project and Quality Plans

Project Plans are typically based on work schedules. The project life-cycle stages should be specified with the project control documentation to be delivered. These disciplines, although perhaps sometimes regarded as irksome, should soon become recognized as enormously helpful in mitigating project risk. Of all the project documentation, the Gantt charts are of the highest importance, defining deliverables and timetables. They are often created with project management tools under version control to monitor project progress against defined milestones and critical paths. Project phases can overlap as long as the validation principles are not compromised. Project milestones are usually also included in the Validation Plan to indicate a commitment to planned project implementation and to stop Project Plans from becoming a regulatory document.

Separate Quality Plans are sometimes established to define the procedures, documentation, roles, and responsibilities that will collectively assure the quality of a computer system. However, Validation Plans are often considered to supersede the need for user Quality Plans.

REQUIREMENTS CAPTURE AND SUPPLIER (VENDOR) SELECTION

Inputs:
  Project Scope (Outline Specification)
Outputs:
  User Requirement Specification
  GxP Assessment
  Request for Proposal (for external rather than internal suppliers)
  Supplier Proposal (for external rather than internal suppliers)
  Supplier Audit (for external rather than internal suppliers)
  Evaluation of Proposal
  Purchase Orders and Contracts of Supply (for external rather than internal suppliers)

User Requirements Specification (URS)

User Requirement Specifications describe the user's functionality requirements, level of user interaction, interfaces with other systems and equipment, the operating environment, and any constraints. Specific regulatory requirements should be included, e.g., requirements regarding use of electronic records and electronic signatures. The documentation making up the URS should:

- Allow the developer to understand the user's requirements
- Clearly define any design constraints
- Provide sufficient detail to facilitate acceptance testing

- Support operation and maintenance of the computer system
- Anticipate and ease phase-out and withdrawal of the computer system

Emphasis is on the user requirements, not on the method of implementation, and should therefore not be product specific. Requirements should be defined such that individual requirements can be identified (with acceptance criteria) for traceability through development and testing. Diagrams should be used to enhance readability.

Because the URS is written in conversational English rather than a stricter, more formal notation, it is notoriously prone to imprecise expression and ambiguity. While this is a feature of everyday conversation where it is often no handicap, it undermines the very objective for which the URS in intended. Care should be taken through review to try to minimize such source of error wherever possible.

GxP Assessment

Examination of the URS is necessary to identify GxP functionality, processes, and/or components implemented by the computer system. These aspects of the computer system should be the focus of attention during validation, especially during Design Review and User Qualification. Some GxP Assessments may be delayed until the Functional Specification has been issued. The GxP Assessment can then be conducted on the Functional Specification. It is often useful for processes to be mapped showing critical points in the process, which computer system(s) support these critical process points, and in what way.

Request for, Receipt of, and Evaluation of Proposals

Requests to potential suppliers for a proposal should include a copy of the URS with specific reference to regulatory compliance requirements. For instance, such references might be to the GAMP Guide and specific paragraphs of the U.S. Electronic Record and Electronic Signature Rule. The proposals received in response to requests need to be evaluated against the request to ascertain any deviations. Once a preferred supplier, or more than one, has been short-listed, then a decision whether or not to conduct a Supplier Audit can be made.

Supplier Selection

The maturity of suppliers in developing software needs to be assessed to determine the level of assurance that they can provide contracted computer systems, services, and/or documents to meet the pharmaceutical and healthcare companies' requirements. These requirements include compliance with regulatory expectations. Regardless of origin, computer systems and their software should

- Be developed according to a defined, documented life cycle of adequate scope and depth that ensures the structural integrity of the delivered software and facilitates its testing and maintenance
- Undergo appropriate testing and be released according to approved procedures
- Be maintained under Configuration Management (if in the form of multiple source code files) and Change Control

User project controls may need to be introduced to compensate for any supplier deficiencies identified.

Supplier Audits are conducted to determine at first hand the supplier's capability. Audits are not normally required for COTS software because they are market-tested. The performance of a supplier with the pharmaceutical or healthcare company and/or other users can be used to determine whether an audit is appropriate.

Supplier Audits are more beneficial if they are conducted as part of the supplier selection and procurement process so that any actions arising can be progressed within a project's implementation. More than one Supplier Audit may be appropriate or necessary for a system where multiple subsystem suppliers or subcontractors are used. The Validation Plan will document which suppliers require and do not require auditing and when these audits should take place. Table 5.2 indicates when Supplier Audits are required.

Purchase Orders and Contracts of Supply

Copies of purchase orders and contracts of supply should be retained in accordance with pharmaceutical and healthcare regulatory requirements in support of validation.

DESIGN AND DEVELOPMENT

Inputs:
- User Requirements Specification
- GxP Assessment
- Supplier Proposal (for external rather than internal suppliers)

Outputs:
- Supplier Project and Quality Plans (for external rather than internal suppliers)
- System Overview
- Functional Specification
- Architectural Design
- Software Design and Program Specification
- Hardware Design
- Data Definition (incl. Configuration)
- Operating Manuals (for external rather than internal suppliers)
- Design Review (incl. Hazard Study)

Supplier Project and Quality Plans

This is essentially the same document as described earlier for the Project Initiation phase but defining the scope of the supplier's role and responsibility. The Supplier Quality Plan should act as an extension to the Supplier Proposal and Contract of Supply. The Supplier Quality Plan should be approved before the Functional Specification is approved. If the supplier is an internal function of the pharmaceutical or healthcare company, these plans do not need to exist as separate documents but rather can be integrated within overall project Validation Plans, Project Plans, and Quality Plans.

System Overview

This is a single document (clear, concise, accurate, and complete) describing the purpose and function of the system. System Overviews are therefore not necessary for less complex systems where there is a single URS or Functional Specification document.

The System Overview is a useful document to present during an inspection and should be written in nontechnical language so that it is meaningful to an inspector without the need for technical expertise. It should include:

- A diagram indicating the physical layout of the system hardware
- Any automated and manual interfaces
- The key functions of indicating inputs, outputs, and main data processing

This information may be contained within the Validation Plan or Validation Determination, in which case it need not be duplicated.

Functional Specification

This describes the functionality of the chosen or developed system and how it will fulfill the user requirements. This document is the specification against which the system operability will be tested and maintained. Specific hardware and software products may be referenced. The Functional Specification should not duplicate information available in standard prepublished supplier documentation if commercial software/hardware is being used, as long as it is referenced and managed under change control.

Contents of the Functional Specification should be cross-referenced to the URS to demonstrate coverage. The Functional Specification must not be approved until its corresponding URS has first been approved. After the Functional Specification is issued, a GxP Assessment may be conducted or updated if it has already been prepared in association with the URS.

Architectural Design

Larger systems will typically benefit from a separate Architectural Design to explain the structure of the computer system and help link Functional Specification to the Hardware and Software Design. For smaller and simpler computer systems this information is more readily included in the Functional Specification. The Architectural Design will be required for the Design Review.

Hardware and Software Design

Design Specifications define the equipment hardware and/or software system in sufficient detail to enable it to be built. Design considerations include inputs and outputs, error handling, alarm messages, range limits, defaults, and algorithms and calculations. The Design Specification may comprise or make use of formal supplier documentation such as data sheets for COTS software products. Design information will be used to support installation, commissioning, and testing. Software Design may itself be supplemented by Program Specifications, depending on the appropriate level of granularity. Diagrams should be used where appropriate to assist readability.

The Design Specification needs to cross-reference relevant sections of the Functional Specification. The Design Specification must therefore not be approved prior to the approval of the Functional Specification. Detailed design information can be included within the Functional Specification, in which case it should be called a Functional Design Specification.

Data Definition (Including Configuration)

Data structures and content must be defined. Actual data to be loaded into tables, files, and databases must be specified by reference to their sources. Data dictionaries should be used to describe different data types. Specific data requirements include:

- Electronic record and electronic signature compliance
- Built-in checks for valid data entry
- That data is only entered or amended by authorized persons
- That GxP data is independently checked

The Data Definition may stand on its own as a separate document or documents or be incorporated within Functional Specification or Design Specification. Data Definition needs to be approved before Data Load can begin.

Operating Manuals

Operating Manuals need to be formally reviewed as fit for purpose by the supplier as they form the basis of User Procedures and User Qualification. Operating Manuals must be kept up to date with developments to the computer systems to which they relate, and they must refer to specific hardware models and software versions making up the computer system being supplied. Recommended ways of working defined by the supplier should be verified as part of the development testing.

Design Review (Including Hazard Study)

A Design Review is undertaken to ensure that all requirements, functional and design specifications, and drawings and manuals have been produced and updated appropriately in readiness for testing. Design Reviews conducted for new and existing systems are often referred to as Design Qualifications. Documentation must be reviewed for the following principal attributes:

- *Clear and Concise:* Documentation should conform to document standards and should be readily understandable.
- *Complete:* A Requirements Traceability Matrix (RTM) should be developed to confirm that all relevant aspects of the URS have been brought forward into the Functional Specification and Design Specifications. This includes both positive and negative requirements (the latter often overlooked), i.e., what the software is supposed to do and what it is NOT supposed to do!
- *Current:* Verify that the documentation is current and that the necessary Change Control measures have been applied.
- *Testable:* Criteria to be used for user acceptance testing must be specific, measurable, achievable, realistic, and traceable to the Functional Specification and Design Specification.
- *Fit for Purpose:* A Hazard Study should be used to identify potential operational problems with the computer system, and how they are or will be managed.
- *Use of Electronic Records/Signatures:* Confirm whether or not the software falls within the scope of the Electronic Records and Electronic Signatures Rule, and if so that the required functionality (where appropriate) has been provided.

The Design Review provides a feedback mechanism into the Design and Development phase to refine the design of the computer system before construction begins in the System Build phase. The Design Review can also be used to carry forward issues to be tested during Development Testing and User Qualification.

Hardware Platform

Computer hardware needs to be assembled in accordance with good practice and any manufacturer's instructions. Hardware components should be itemized in the Hardware Design and checked as part of the Installation Qualification.

CODING, CONFIGURATION, AND BUILD

Inputs:
Hardware Design

Software Design and Program Specification
Configuration Definition
Software Programming Standards
Outputs:
Hardware Platform
Application Software
Source Code Review

APPLICATION SOFTWARE

Application software includes all of the software required to implement a computer system. An inventory of application software should be itemized as part of the Functional Specification, the Architectural Design, or the Software Design as appropriate, and checked as installed as part of the Installation Qualification.

Initial and subsequent commercial software product releases should only be used once they have been proven over a period of time by a user community to be fit for purpose. It is recommended that software not be used until it has been commercially available for at least 6 months. Software products released by a supplier that are pending final evaluation (so-called *beta testing* of software) must NOT be used to support the manufacture of drug and healthcare products!

SOURCE CODE REVIEW

A Source Code Review must be performed on application software unless there is evidence from the Supplier Audit that the source code has been, or will be, developed in a quality assured manner and subjected to review as part of its development life cycle. The decision and justification not to perform a review must be documented within the Validation Plan.

The Source Code Review aims to provide confidence in the operability of the system and, as such, has five basic objectives:

- To verify the adoption of good programming practices (e.g., headers, version control, Change Control) and compliance with documented, audited programming standards
- To determine the level of assurance by which the code fulfills design specifications including process sequencing, affirming I/O handling, formulae, algorithms, message, and alarm handling
- To detect possible coding errors
- To identify evidence of dead code
- To check that superfluous options are deselected or disabled and cannot accidentally be enabled

DEVELOPMENT TESTING

Inputs:
Functional Specification
Hardware Design
Software Design and Program Specification
Design Review (incl. Hazard Study)
Outputs:
Unit/Module Testing
Integration/System Testing
Predelivery Inspection

Unit/Module Testing

Unit testing focuses on verifying the smallest unit of Software Design — the unit/module or program. Using the Software Design and Program Specification, important control paths with worst case conditions are exercised. These activities include data entry operations. Also, the limits of internal boundaries need to be checked by exercising the ranges of acceptable values. All these tests are intended to expose errors within the unit/module or program. The relative complexity of tests and the level of residual concealed errors are limited by the constrained scope established for unit testing. Unit/module testing is normally *white-box* orientated; in other words, tested by someone who understands exactly how the code works (so the *box* is *white,* i.e., you can look inside it). The work can be conducted in parallel for multiple units/modules or programs. Supplier Audits and Source Code Reviews may recommend specific unit/module testing activities. Unit/module testing may not be appropriate for instrumentation and control systems which tend to be more self-contained computer systems.

System/Integration Testing

Integration testing is a systematic technique for constructing a system while conducting tests to reveal errors associated with interfacing. It may be tempting to adopt a nonincremental approach to integration testing — i.e., to test the system as a whole without incrementally testing each assembly. The problem with this approach is that when errors are discovered it is difficult to precisely identify the cause and hence specify effective and error-free corrective action. When corrections are thus implemented, they can introduce new system errors that themselves are difficult to isolate for the same reason. Such uncontrolled testing can become extended, frustrating, and dangerous for ultimate product quality.

Incremental integration testing is the antithesis of the nonincremental approach. The system is tested as subsystems before being tested as collections of subsystems; finally, the complete system is tested. Errors should be easier to isolate and correct predictably, interfaces should be easier to test more comprehensively, and it should be easier to develop a systematic test plan.

System testing should be comprehensive and should include:

- Normal system operation including failure mode operation
  - Processing sequences
  - I/O handling
  - Formulae, calculations, and algorithms
- All error conditions (contrivance of error conditions may be needed) and associated error message intelligibility, relevance, and treatment
- All range limits with alarm conditions
- Service continuity throughout defined operating environment
- Recommend testing from Source Code Review

Predelivery Inspection

Pharmaceutical and healthcare companies may consider conducting predelivery checks on their suppliers to verify that Supplier Project/Quality Plans have been implemented. Computer systems should not be accepted at their user sites if outstanding and agreed issues from the Supplier Audit have not been resolved to the satisfaction of the pharmaceutical or healthcare company.

Predelivery checks can significantly reduce overall project timelines if conducted properly. For example, some or all of the documentation destined for delivery with the computer system may be used in lieu of on-site inspection if the appropriate approvals are obtained prior to the predelivery checks. It is recognized that there may be situations where the cost of attending to these checks may outweigh the benefits. The application of predelivery checks should be contractually specified with suppliers and the outcome of Predelivery Inspections documented.

USER QUALIFICATION AND AUTHORIZATION TO USE

Inputs:
User Requirements Specification
GxP Assessments
Functional Specification
Hardware Design
Data Definition (incl. Configuration)
Operating Manual
Design Review (incl. Hazard Study)
Outputs:
Prequalification Commissioning and Calibration
Data Load (incl. Configuration)
Installation Qualification
Operational Qualification
Performance Qualification
Operation and Maintenance Prerequisites
Validation (Summary) Report

PREQUALIFICATION ACTIVITIES

The installation site must be prepared (commissioned) as required and necessary calibration of equipment and instruments conducted. This activity is usually separated from Installation Qualification.

DATA LOAD (INCLUDING CONFIGURATION)

Procedures and protocols must define the data load (entry) process to fulfill the Data Definition requirements identified during the Design and Development phase. All GxP data, including configuration, should be double-checked to verify that it is correct. Statistical sampling can be used to check other data commensurate with the business need to verify data integrity. Rationales justifying sampling regimes must be defined. Automated data load tools should be validated.

INSTALLATION QUALIFICATION (IQ)

This records the checks to establish that the installation has been completed in accordance with system specifications. It may contain

- Inventory Checks (hardware, software, data, user manuals, and SOPs)
- Operating Environment Checks (e.g., power supply, RFI, EMI, RH, temperature)
- Diagnostic Checks (installation diagnostics and software launch)

The boundary of the system and hence the scope of the IQ must be defined in the Validation Plan. The RTM should be updated with IQ cross-references.

OPERATIONAL QUALIFICATION (OQ)

Tests must be designed to demonstrate that the installed computer system functions as specified under normal operating conditions and, where appropriate, under realistic range conditions. Destructive testing is not required.

OQ testing should only be conducted after the IQ has been successfully concluded. The scope of the OQ should be defined in the Validation Plan. System Testing can be repeated or referenced to reduce the amount of OQ testing required, provided supplier documentation standards fulfill user qualification requirements. The OQ should cover:

- Confirmation of user functionality
- Audit trails for electronic records
- Application of electronic signatures
- Verification of operation and maintenance of SOPs
- Verification of business continuity plans

OQ protocols should define any ordering between individual tests. Specific tests may be recommended by Supplier Audit, GxP Assessments, and Design Reviews. The RTM should be updated with OQ cross-references. The RTM should specifically identify where GxP functionality identified by the GxP Assessment is tested.

Performance Qualification (PQ)

PQ has often been the least understood phase of User Qualification. This is probably because the character of PQ testing can vary considerably between different computer systems.

PQ testing should be designed to demonstrate that the installed computer system's functionality is consistent and reproducible. The PQ may be conducted, in whole or in part, immediately after the computer system is brought into use, but it cannot be brought into routine use until the PQ has been successfully completed.

PQ testing should only be conducted after the OQ has been successfully concluded. The scope and timing of the PQ should be defined in the Validation Plan. The scope of the PQ should cover (as appropriate):

- Product performance qualification to verify that critical items/records are consistent (e.g., batch records, labeling variants)
- Process performance qualification to verify that critical functionality is reproducible (e.g., coping with operating environment variations, demonstrating integrity and accuracy of data processing, and managing service metrics such as availability, reliability, and probability of failure on demand)

Parallel ways of working are expected to complement PQ in case the validation fails or a catastrophic failure occurs. A back-out strategy should be developed involving manual ways of working or switching over to an alternative validated system, or a hybrid combination of both. Maintaining data integrity is the most important objective here. The strategy is likely to make use of current business continuity plans.

PQ protocols should define any ordering between individual tests. The RTM should be updated with PQ cross-references.

Operation and Maintenance Prerequisites

It is necessary to ensure that arrangements for operation and maintenance of the computer system are either already established, or that documented plans have been prepared to ensure that these arrangements are in place by the time the system is authorized for use.

- ***Performance Monitoring***
  SOPs for collection and analysis of performance data must be approved before data is collected. Statistical analysis should be conducted under the supervision of a professional

statistician where results are used to support GxP decisions. Performance monitoring may be linked to preventative maintenance, e.g., as part of a Reliability Centered Maintenance (RCM) program.

- ***Repair and Preventative Maintenance***
  The Validation Plan may cover the requirements for maintenance planning for the system, in which case they should be verified by the IQ/OQ. Where this is not the case, the following areas will be addressed under the Validation Report:
  - Recommended Spares Holding
  - Frequency of Routine Testing/Calibration

  SOPs covering maintenance activities must be approved before the system is used.
- ***Upgrades, Bug Fixes, and Patches***
  SOPs for software upgrades, bug fixes, and patches must be approved before such changes can be made.
- ***Data Maintenance***
  SOPs supporting the management and control of data integrity must be approved before the computer system is used. Procedures will include data change control.
- ***Backups and Recovery***
  SOPs for backup and restoration of software and data files must be approved and verified before the computer system is used. It is not unknown for some projects to verify the backup process but to forget to verify the restoration process with the adverse results that can be had if it does not work when called upon.
- ***Archiving and Retrieval***
  SOPs for archiving, retention, and retrieval of software, data, documentation, and electronic records must be specified, tested, and approved before the system is approved for use.
- ***Business Continuity Planning***
  Procedures and plans supporting business continuity (Disaster Recovery Plans and Contingency Plans) must be specified, tested, and approved before the system is approved for use. Business Continuity Plans will normally be prepared for a business or an operational area rather than for individual computer systems. It is likely that the only way to verify the plan is to walk through a variety of disaster scenarios. Topics for consideration should include catastrophic hardware and software failures, fire/flood/lightning strikes, and security breaches. Alternative means of operation must be available in case of failure if critical data is required at short notice (e.g., in case of drug product recalls). Reference to verification of the Business Continuity Plans is appropriate during OQ/PQ.
- ***Security Management***
  SOPs for managing security access (including adding and removing authorized users, virus management, and physical security measures) must be specified, tested, and approved before the system is approved for use.
- ***Contracts and Service Level Agreements***
  Commercial contracts and service level agreements for operation and support of computer systems should be established prior to their use.
- ***User Procedures***
  User Procedures for operating and maintaining the computer systems must be specified, approved, and, where possible, tested before the systems are approved for use. Projects should aim to use and thereby test User Procedures within User Qualification activities. This approach offers the opportunity to use end-users to help conduct testing with the User Procedures. This can often be coordinated as a training exercise. User Procedures can be refined by end-users themselves in readiness for handover of the computer system.

- ***Availability of Software and Reference Documentation***
  Application software and relevant development documentation must be available for inspection. Formal access agreements should be established if access to software is restricted, e.g., escrow accounts.

The RTM should be updated to link operation and maintenance activities that address system requirements. Operation and maintenance requirements are discussed in more detail in Chapter 11.

Validation (Summary) Report

A Validation Report must be prepared at the conclusion of the activities prescribed in the Validation Plan. Where there are deviations from the Validation Plan or unresolved incidents, these should be documented and justified. Where critical unresolved issues remain, the computer system cannot be considered validated or fit for purpose.

The Validation Report for a system must not be approved until all the relevant documents defined within its Validation Plan have been approved. Approval of the Validation Report marks the completion of the validation process. The Validation Report must, therefore, include a clear statement confirming whether or not all computer systems are validated and fit for purpose.

Some pharmaceutical and healthcare companies prepare what is referred to as a Validation Summary Report. This is really the Validation Report described above, but at a very high level with little detail.

The concept of a Validation Summary Report can be taken a step further in the form of a Validation Certificate. This merely states that a computer system is validated, and it specifies a review date against this validated status. While a Validation Certificate could be presented to a regulatory inspector as evidence of validation, this would probably just prompt the inspector to request more detailed information. The effort to produce and maintain Validation Certificates must be carefully weighed.

PROJECT DELIVERY SUPPORTING PROCESSES

The following should be established to support the project and its hand-over to ongoing use. With the exception of requirements traceability these topics are discussed in more detail in Chapter 4.

Training

Project staff should be trained as required for the project. Users should also receive training in advance of using a computer system. Indeed, all personnel using or maintaining a computer system are expected to be trained to the correct level of competency before they are allowed to operate that system.

Training should be conducted in a timely manner. A Training Plan should be developed to achieve this objective. Training must be conducted against approved procedures and training records updated to reflect training given on the system. Any requirements for refresher training should be audited through Periodic Reviews.

Contractors and temporary staff should have a CV stating all relevant details retained by the responsible manager or included in the staff training records. Staff should be selected to fully exploit their skills, education, and experience.

Document Management

Documentation and records must be reviewed prior to approval and maintained under Change Control. Amendments to documentation and records must not obscure the original content. All such amendments must be signed and dated. Attachments to documents and records must be marked as such (e.g., test results).

Raw data attached to documents and records or existing in their own right should be clearly itemized, signed, and dated. Sets of raw data should be physically coupled together (e.g., bound or stapled). Sets of raw data should have each page marked as belonging to the set, and the front page signed and dated.

Change Management

Change Control must be established for software, hardware, data, and documentation. Procedures for Change Control may be project specific and vary from the procedure used after hand-over of the project. Any transition in Change Control procedures needs to be managed. Software should further be checked for unauthorized changes. Although they should not occur and must be strictly forbidden, regrettably unauthorized changes do occur in practice.

Configuration Management

A system should be established to document and control the versions of computer system software and hardware through the development, testing, and use of the system. Configuration Management needs to link software and hardware versions to particular data sets and document versions. Changes to software, hardware, data, and documentation may be interdependent.

Requirements Traceability

In the words of ex-FDA investigator Martin Browning "traceability is the absolute key [to documentation]" through design, development, deployment, and all the way through to decommissioning. Specifically, requirements traceability will improve:

- Test coverage
- Impact assessments of change
- Inspection support

A Requirements Traceability Matrix (RTM) or equivalent mechanism for establishing and maintaining requirement traceability should be put in place for projects for use during operation and maintenance. This mechanism should provide a method of tracing a requirement from the URS through the Design and Development and User Qualification phases. It provides assurance that all system requirements have been included in the design and tested to verify correct operation.

During the operational life of the computer system, requirements traceability will enable impact assessments to be conducted in support of regression testing of changes. It is quite normal for a particular aspect of a computer system to require more than one test. When making a change, it is important to appreciate if this one-to-many situation exists and ensure that all appropriate tests are executed.

Requirements traceability should also ease inspections by facilitating the rapid location of any specification-to-test information requested. For many organizations, this forward tracking of requirements can be quite cumbersome if they rely solely on reverse tracking offered by test documentation referencing requirements and specifications.

Deviation Management

Validation deviations during the project must be managed. Details of deviations must be recorded with a description of the circumstances under which the deviation was noted (e.g., reference to design review or test case) and the name of the person noting the deviation. A record is also required for the remedial action taken to a deviation (including any testing required) or the justification for not taking action. An index of all deviations should be maintained. Deviations can be prioritized for resolution.

VALIDATION PACKAGE

Throughout the validation process a package of documentation should be maintained that contains the working documents of the validation process. The contents of a Validation Package (to be defined in the Validation Plan) are indicated in Table 5.5. Key validation documents are indicated in Figure 5.6 and must include any Change Control records.

The Validation Package (both formal documents and raw data) must be securely stored in accordance with site procedures and be readily retrievable. The Validation Package will normally be stored in a centralized site archive/repository.

REVIEWS AND APPROVALS

Review and approval requirements for documentation can be defined in procedures or specified in the Validation Plan. Roles and responsibilities will vary between organizations; there are no definitive role models.

Recommended minimum approval levels for validation documentation generated at each stage and the associated responsibilities for each signatory are suggested in Table 5.6. The “Developer” role introduced in Chapter 3 is split into the User Project Manager and Supplier. The User Project Manager is the pharmaceutical or healthcare company’s own project manager. The Supplier can be an internal group within the pharmaceutical or healthcare company’s organization or an external company. Similarly, the “Quality and Compliance” role from Chapter 3 is split into Compliance Oversight and Operational Quality. Compliance Oversight represents QA staff with specialist computer compliance knowledge. Operational Quality represents QC department staff supporting general GxP operations. Other roles may be more appropriate depending on the organizational structure.

Table 5.6 also defines responsibilities. The “Originator” responsibility is preparing a document or record, “Review” responsibility is confirming technical content and consistency with other validation activities, and “Approve” responsibility is authorizing the activity as complete and correctly documented. The Quality Unit may at its discretion, as part of its approval process, review or audit supporting and referenced deliverables. Due account must also be taken where a Supplier Audit determines that the pharmaceutical or healthcare company should take over a lead responsibility for a supplier’s documentation responsibilities. The review and approval roles presented in Table 5.6 are consistent with the review of regulatory expectations presented in Chapter 14.

TABLE 5.5
Validation Package Contents

Validation Package:	Software Category				
Documents and Records	2	3	4	5	Comments
System Overview	X	X	X	X	Executive introduction to overall system
Validation Determination Statement	X	X	X	X	May initiate a detailed electronic record/signature assessment
Validation (Master) Plan	X	X	X	X	Scope of a Validation Plan need not be limited to one system; for exact replicas of a computer system the Validation Plan can be proceduralized; reference may be made to the Project Quality Plans and Supplier Quality Plans if they are separate documents
User Requirements Specification	X	X	X	X	Typically a single document covering whole system
GxP Assessment	X	X	X	X	For overall system
Supplier Audit Report	—	—	X	X	For bespoke and critical COTS-based applications
Functional Specification	X	X	X	X	For standard COTS packages reference to product specification is sufficient
Architectural Design	—	—	X	X	Only consider for large or complex systems; may be combined with Functional Specification
Software Design and Program Specification	—	—	X	X	Configuration only for Category 4 software
Hardware Design	—	—	—	—	For bespoke hardware; otherwise reference COTS hardware documentation
Data Definition (incl. Configuration)	X	X	X	X	To cover electronic records and any data configured/bespoke data structures
Design Review (incl. Hazard Study)	—	—	X	X	Typically covering whole system (sometimes known as Design Qualification)
Source Code Review	—	—	X	X	Configuration only for Category 4 software
Unit/Module Testing	—	—	—	X	May be combined as appropriate with Integration/System Testing
Integration/System Testing	—	—	—	X	
Predelivery Inspection	—	—	X	X	For bespoke and critical COTS-based applications, follow on from Supplier Audit
Prequalification Activities	X	X	X	X	Site preparations, commissioning, and calibration
Data Load (incl. Configuration)	X	X	X	X	Only for systems that require data population
Installation Qualification	X	X	X	X	Qualification documents for IQ, OQ, and PQ may be combined as long as test plans and test cases are collectively approved before testing; for standard COTS Category 2 software qualification may be based on calibration activities
Operational Qualification	X	X	X	X	
Performance Qualification	X	X	X	X	
Operation and Maintenance Prerequisites	X	X	X	X	Activities include User Procedures/Manuals, Maintenance, Backup and Restoration, Security, Training, Business Continuity Plans
Validation (Summary) Report	X	X	X	X	Response to Validation (Master) Plan including Statement whether validation was successful and the system is authorized for routine use

Note: There are no specific validation activities for Category 1 software (COTS compilers and operating systems) beyond documenting version details.

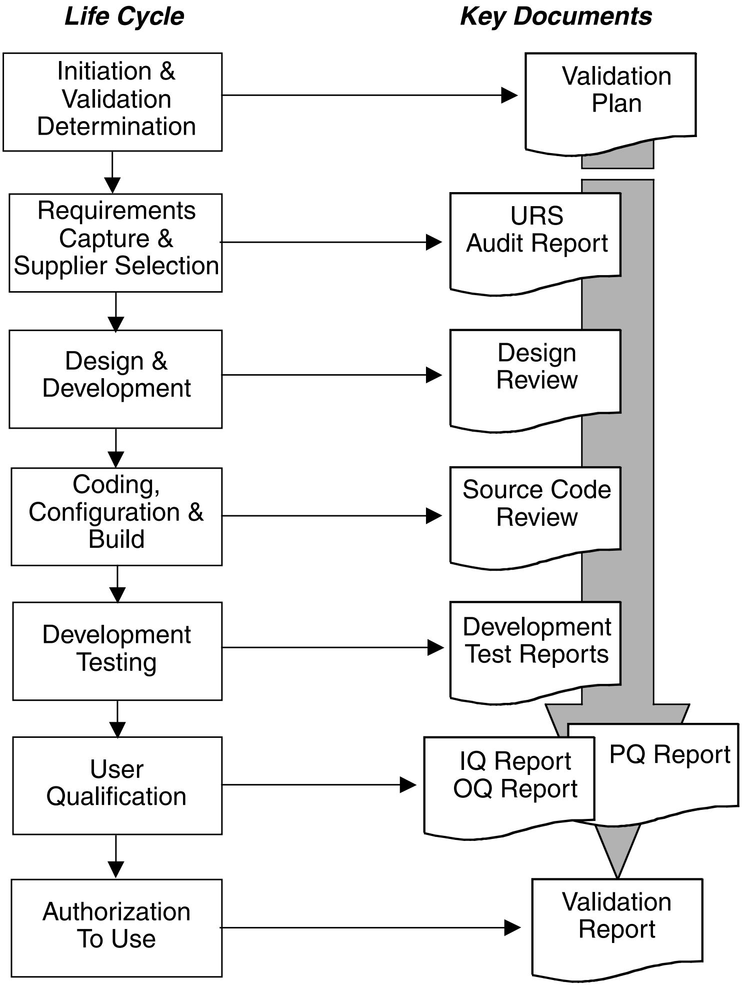


FIGURE 5.6 Key Validation Deliverables.

TABLE 5.6
Example Validation Package Reviews and Approvals

		Developer		Quality and Compliance	
Project Document	**System Owner/ User**	**User Project Manager**	**Supplier**	**Compliance Oversight**	**Operational Quality**
System Overview	O/A	—	—	Advice, support, and audit as required	R
Validation Determination Statement	A	—	—		O/A
Validation (Master) Plan	A	A	—		O/A
Project and Quality Plans	A	O/A	—		—
User Requirement Specification	O/A	—	—		A
GxP Assessment	A	—	—		O/A
Supplier Audit Report	R	R	—		O/A
Supplier Project and Quality Plans	—	R	O/A		R
Functional Specification	—	—	O/A		R
Architectural Design	—	—	O/A		—
Software Design and Program Specification	—	—	O/A		—
Hardware Design	—	—	O/A		—
Data Definition (incl. Configuration)	—	—	O/A		—
Operating Manual	R	—	O/A		R
Design Review (incl. Hazard Study)	A	O/A	R		A
Source Code Review	—	—	O/A		R
Unit/Module Testing	—	—	O/A		—
Integration/System Testing	—	—	O/A		—
Predelivery Inspection	A	O/A	—		R
Prequalification (Commissioning and Calibration)	A	O/A	—		R
Data Load (incl. Configuration)	A	O/A	—		A
Installation Qualification	A	O/A	—		A
Operational Qualification	A	O/A	—		A
Performance Qualification	A	O/A	—		A
Operation and Maintenance Prerequisites	A	O/A	—		R
Validation (Summary) Report	A	A	—		O/A

Note: O = Originator, R = Recommended Review, A = Approve, — = Review or Approval not mandated.

REFERENCES

1. U.S. Food and Drug Administration (2002*), General Principles of Software Validation; Final Guidance for Industry and FDA Staff.*
2. FDA (1987), *Software Development Activities*, Technical Report, Reference Materials and Training Aids for Investigators, Food and Drug Administration, Rockville, MD.
3. FDA (1997), *General Principles of Software Validation: Guidance for Industry*, Draft Guidance Version 1.1, June.
4. David Begg Associates (2001), *Computer Systems and Automation Quality and Compliance,* York (U.K.), June 5–7.
5. Wingate, G.A.S., Smith, M., and Lucas, P.R. (1995), *Assuring Confidence in Pharmaceutical Software*, Safety and Reliability of Software Based Systems, First Annual ENCRESS Conference, Bruges, Belgium, September.
6. Wingate, G.A.S. (1997), *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice*, Interpharm Press, Buffalo Grove, IL.
7. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (www.ispe.org).
8. Futrell, R.T., Shafer, D.F., and Shafer, L.I. (2002*), Quality Software Project Management*, Prentice Hall, Upper Saddle River, NJ.

6 Project Initiation and Validation Determination

CONTENTS

The first phase in the project life cycle is Project Initiation. This is the responsibility of the pharmaceutical or healthcare company. It consists of five main activities that may be conducted sequentially or concurrently. These are:

- Definition of the Project Scope
- Determination of the Validation Requirements
- Drafting of the Validation Master Plan
- Drafting of other subordinate Validation Plan(s)
- Definition of the Validation Strategy

The magnitude of validation effort[1] should be commensurate with the risk associated with the computer system, the computer's dependence on software for potentially hazardous or critical functions, and the capability of the software supplier. For example, while all software modules should be validated, those modules that are critical should be subjected to a more thorough and detailed design, development, and testing. Such assurance is derived far more securely from the methodology by which the code was developed and is supported than from functional testing by the user. Likewise, size and complexity is an important factor in establishing the appropriate level of effort and associated documentation for the software. The larger the system, the more extensive procedures and management controls are required. Small and large firms alike cannot justify a superficial validation on the basis of scarce resources.

PROJECT SCOPE

The validation approach adopted ought to be challenged at the outset, and any compliance rationales rigorously scrutinized for their long-term benefits in mitigating risk to the operation. This is far better than seeking penny-pinching economies under pressure of criticism after the project is under

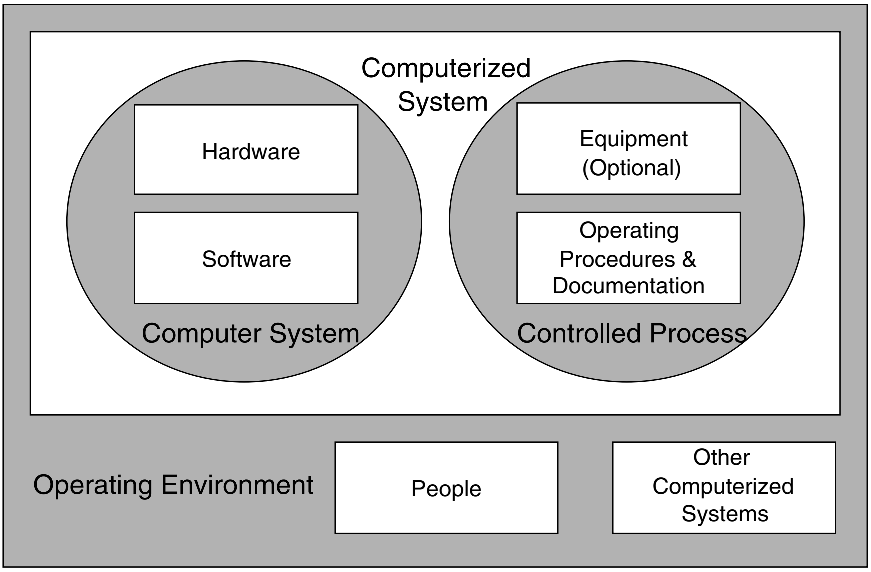


FIGURE 6.1 The PMA Conception of a Computerized System.

way. Understand the project risks, then manage them proactively. A well-specified and planned project has a good chance of reaching a successful completion.

Understanding the Scope of the Computer System

The scope of the computer system being validated must be clearly understood. Computer systems can be thought of as comprising a programmable electronic device that may be connected to actuators and sensors (and possibly other computer systems) via communication links. Closely associated with the computer system are data and equipment, people, and procedures. A schematic of a computer-related system based on some work by the U.S. Pharmaceutical Manufacturers Association (PMA)[2] is shown in Figure 6.1.

Outline Specification

An Outline Specification is usually required to sanction a project and authorize the allocation of a budget. It typically presents the business case for the computer system, defining key functionality and compliance requirements. This document may be used later on for developing the User Requirements Specification (URS) as discussed in Chapter 7. Where Commercial Off-The-Shelf (COTS) software and systems are being acquired, this document may be used as the URS. Outline Specifications must be reviewed and approved.

Project Risk Management

A league table of top project risks is presented in Table 6.1. If these are relevant in a particular case but in practice treated casually or even ignored, then project budgets, schedules, and system functionality will almost certainly be compromised. This in turn is bound to affect the standard of validation. It is therefore painfully clear that project risk management is very important not only in terms of project delivery but also in terms of the operational compliance that the computer system will be capable of achieving once put into use.

Potential risks to a project must be identified early so that they can be proactively addressed. It is naïve to assume there are no risks and therefore the project will automatically run smoothly — if only life were like that! The amount of effort devoted to identifying risks should correspond to the

TABLE 6.1
Project Risk Factors (Based on Van Vliet[2])

Risk	Description
Lack of competent personnel	Exemplified by inexperience with tools and development techniques to be used, personnel turnover, loss of critical team members, and a failure to match the size of the team to the complexity of the project.
Unrealistic schedule/budget	Too little money, too little time.
Wrong functionality	Causes include failure to understand customer needs accurately or comprehensively, and a lack of development process capable of implementing these faithfully into code.
Inappropriate user interface	The quality (e.g., ease of use) of the user interface may be critical to system success.
Gold plating	Developers may allow their enthusiasm for technical sophistication to outstrip the features demanded by the customer.
Requirements volatility	Requirements, and thus designs, altered during development often have a hugely damaging impact on system quality.
Poor external component/tasks	Suppliers may provide inadequate products/services.
Real-time shortfalls	The performance of the system may be inadequate.
Capability shortfalls	An unstable operating environment or new/untried technology poses a risk to the development schedule.

perceived criticality of the project. So, simple projects implementing COTS computer systems will generally be of a lower risk level than complex projects involving bespoke software.

Project risks can be mitigated using a five-stage approach:[3]

- Identify the risk factors such as those listed in Table 6.1.
- Determine the level of exposure to the identified risks (the seriousness these carry coupled with the probability that they will occur). Formal risk assessment methodologies such as Kepner-Tregoe® (KT) might play a useful part here.
- Develop strategies to minimize and mitigate risk. This would normally be done for those risks with the highest exposure level or for those whose risk probability/seriousness or KT factor exceeds some defined threshold.
- Implement risk management strategies.
- Review effectiveness of risk management measure.

Hopefully the risk management activities implemented will prove sufficient. Some risks may still threaten success despite risk management activities, so the strategy should be to continually keep risks under review and revise activities as necessary to deal with residual and emergent risks. It may prove necessary to invoke contingency plans or even move into crisis management mode if the risks appear to be leading the project to failure.

Project and Quality Plans

Project and Quality Plans provide a mechanism to control the management of the project. Planning needs to focus on scheduling activities, controlling cost, allocating resources, and managing technical issues. Good project management is a basic business expectation.

Project Plans should address:

- Establishment of project organization
- Allocation of resources

- Identification of all activities, critical path, and key milestones (Gantt charts)
- Recognition and management of project risk
- Ensuring project prerequisites are put in place
- Establishment of progress reporting method
- Budget management, ensuring business case maintained (beware of scope creep)
- Resolution process for open issues
- Links to Quality Plan for standards and quality control

Quality Plans should address:

- Quality responsibilities of project organization
- Standards for work conducted, including the software development life cycle to be used, and the documents to be delivered by it
- List of other deliverables to be produced (e.g., project life-cycle documents such as control charts) and quality criteria
- Quality control activities (e.g., software reviews, testing, and document reviews)
- Definition of change control procedures applicable to project
- Configuration management practices
- Document management practices
- Training requirements

Successful projects do not happen by accident. They result from all the activities being well managed and done right the first time. But as Tony Simmons acknowledges, this is an ideal that is seldom realized in practice, where change happens and people make mistakes.[4] Chapter 1 reviews some of the most common problems that afflict projects. Project Managers might want to consider using established project management methodologies such as PRINCE2 to help engineer predictability into the process and ensure a successful project outcome. A glossary of terms and acronyms should also be collated as an integral part of the planning exercise. There must be a common understanding of terminology used in a project. It is all too easy for individuals to attribute their own meaning to standard terminology and thereby proliferate endless misunderstandings!

Quality planning, and to a lesser extent project planning, overlap validation planning. Project Managers should consider how to avoid duplicating information. Project Plans and Quality Plans can be combined into a single document, although there are pitfalls here. The Project Plan forms the charter for the Project Managers who have a *driving* role. The Quality Plan, on the other hand, is the charter for Quality Assurance whose role is to *constrain* the project into the paths of engineered best practice. These roles are always potentially in conflict, but both are essential to project success. Keeping the two plans separate but complementary reinforces this distinction and helps build clarity of understanding of the roles among the project participants, especially the Project Sponsors. Both the Project Plan and the Quality Plan are typically approved by the System Owner/User and Project Manager. Project Plans and Quality Plans are not normally subject to regulatory inspection if Validation Plans are prepared to meet regulatory expectations.

VALIDATION DETERMINATION

Early Indication of Validation Requirement

An early decision as to whether a computer system requires validation or not can help ensure that the necessary support for validation is provided for at the outset of a project. Late identification of validation requirements could result in significant costs and delays if additional activities and documentation are required. In addition, many regulatory authorities devalue such *retrospective validation*.

Validation Criteria

Computer systems require validation when an affirmative answer must be given to one or more of the following questions:[5]

- Does the application or system directly control, record for use, or monitor product quality?
- Does the application or system directly control, record for use, or monitor laboratory testing or clinical data?
- Does the application or system affect regulatory submission/registration?
- Does the application or system perform calculations/algorithms that will support a regulatory submission/registration?
- Is the application or system an integral part of the equipment, instrumentation, or identification methods used in testing, release, and/or distribution of the product/samples?
- Does the application or system define materials (i.e., raw materials, packaging components, formulations, etc.) to be used?
- Can the application or system be used for product/samples recall, reconciliation, stock tracing, product history, or product-related customer complaints?
- Will data from the application or system be used to support Quality Control product release?
- Does the application or system deal with coding of materials, formulated products, or package components (i.e., labels or label identification)?
- Does the application or system hold or manipulate stock information, stock status, location, or shelf life information?
- Does the application or system handle data that could affect product quality, strength, efficacy, identity, status, or location?
- Does the application or system employ any electronic signature capabilities or provide the sole record of the signature on a document subject to review by a regulatory agency?
- Is the application or system used to automate a manual Quality Control check of data subject to review by a regulatory agency?
- Does the application or system create, update, or store data prior to transferring to an existing validated system?
- Is the application or system the official archive or record of any regulated activity and thus subject to regulatory audit?

An in-depth assessment is not required at this stage. It should be possible to furnish an immediate "yes" or "no" to each question.

Validation Determination Statement

A simple form such as the example given in Appendix 6A can be used to document the decision whether or not to validate a system. Validation Determination Statements should be completed no later than at the moment the URS is released. The basic information on which to base a decision should be available at this time. In any case, once issued, there is nothing to prevent the Validation Determination Statement being updated with additional information when available. The important thing is establishing early on whether or not validation is required, so that project planning can take this into account. To give a practical implication here, purchasing departments may need this information to implement contractual clauses relating to validation expectations with the suppliers.

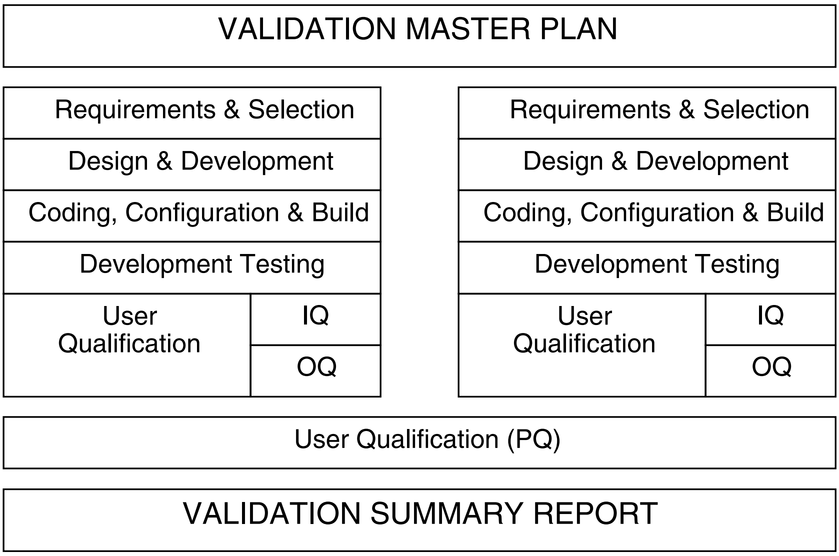


FIGURE 6.2 Simple Validation Master Plan.

VALIDATION MASTER PLAN

HIERARCHY OF VALIDATION PLANS

The term Validation Master Plan usually denotes that the associated project covers large or multiple computer systems. For example, where more than one Validation Plan exists to cover the implementation of a system, a supervisory plan should be produced to define the overall approach. Similarly, if the computer system is relatively small and contained in its entirety within a standalone piece of equipment, then the Validation Plan may be embodied within the overall Validation Master Plan for the equipment/area in question.

GAMP 4 suggests a hierarchy that might look like this:[6]

- Management Plan for multisite activities
- Site Validation Master Plan (for entire site)
- Validation Master Plans (for complex systems or supervision of multiple systems)
- Validation Plans (for individual systems)

Whatever pattern a supervisory validation planning takes, it must clearly identify the scope of any subordinate Validation Plans.

Figure 6.2 shows how the validation activities for a pair of linked computer systems might be structured within a Validation Master Plan. This can be compared to Figure 6.3 which shows how the validation activities might be further structured to include a third, and entirely independent, computer system within the same Validation Master Plan.

CONTENTS OF VALIDATION MASTER PLAN

Validation Master Plans typically have three main sections. The first section states how the pharmaceutical or healthcare company's own validation philosophy and policy address regulatory GxP requirements. The second section defines the scope of validation, identifying which computer system systems require validation. All computer systems whose malfunction could possibly affect

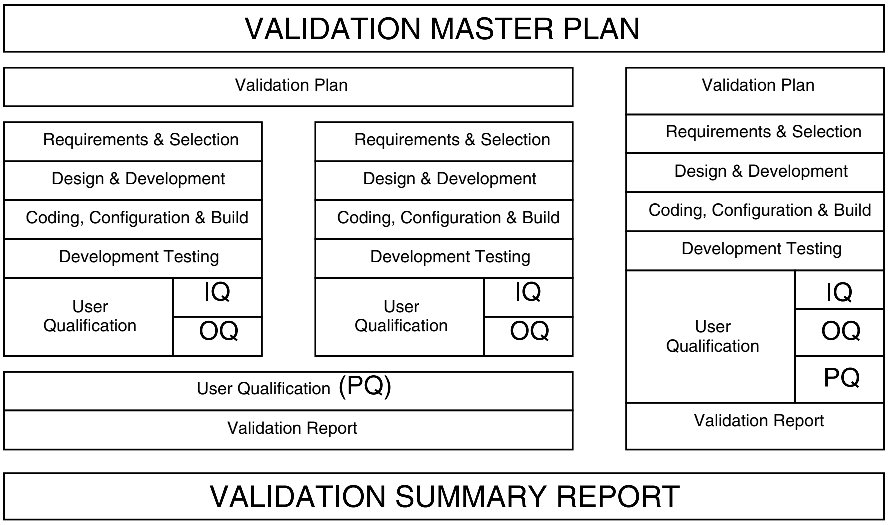


FIGURE 6.3 Complex Validation Master Plan.

the safety, quality, and efficacy (during development or manufacture) or batch tracking (during distribution) of drug products should be validated. A register or inventory of computer systems to be validated is sometimes attached to the Validation Master Plan as an appendix. Finally, the third section commits the pharmaceutical or healthcare company to some basic milestones. These milestones are usually associated with the launch of a new drug product, but may also be timetabled to satisfy anticipated inspections by GxP regulatory authorities. The milestones demonstrate the pharmaceutical and healthcare company's intent and may be revised during the course of a validation program to reflect changing conditions.

Senior managers should not be unduly worried about delays, as long as the pharmaceutical or healthcare company can demonstrate progress to the regulatory authorities and continued commitment to a revised timetable. However, senior managers must expect to be held to any completion dates specifically agreed to with regulatory authorities. It is usually a good idea to keep them regularly appraised of progress at suitable intervals. Failure to deliver on agreed dates is likely to be seen as demonstrating a distinct lack of commitment to address compliance requirements. As much notice as possible should be given to regulatory authorities if delays are expected to agreed completion dates. Care should be taken to explain to the regulatory authority concerned how the delays came about and how they are being addressed.

Structure of Validation Master Plan

The GAMP Guide suggests the following layout for a Validation Master Plan:[6]

- Introduction and Scope
- Organizational Structure
- GxP Criticality Assessment Process
- Validation Strategy
- Change Control
- Procedures and Training
- Document Management
- Timeline and Resources

In addition, a Glossary may be added as required to aid understanding of the Validation Master Plan. Bear in mind that not all terms routinely used within the organization will be familiar to those outside it.

Introduction and Scope

Here, reference should be made to relevant policies, and where the document fits into the level of planning should be described. The scope and boundaries of the site/area/systems being validated should be defined as appropriate. Reference should also be made to any subordinate Validation Plans, and the period within which this plan will be reviewed should be stated.

Organizational Structure

The roles and responsibilities such as ownership, technical support, and QA should be defined. Depending on the level of planning, these may be departmental roles and responsibilities, or corporate. Individuals should not be named; instead, their job titles should be identified.

GxP Criticality Assessment Process

How the computer system's validation requirement was identified should be explained, including how any prioritization was determined and how any changes in priority are managed. Identify procedures used in criticality assessment.

Validation Strategy

The overall computer validation strategy within the Validation Master Plan should be outlined, as well as the life-cycle model to be applied and the validation procedures to be used.

Change Control

The approach to change management should be mentioned, referring to relevant change control procedures.

Procedures and Training

The training requirements for new and existing SOPs should be described.

Document Management

The contents of validation packages to be produced should be defined, together with the identification of document management and control procedures to be used. Any special requirements need to be specified and clearly understood.

Timeline and Resources

Indicate planned end-dates and appropriate intervening milestones. Identify resources to be assigned to various activities. Note any critical dependencies that may impact progress.

Preparation of a Validation Master Plan

The time to prepare a Validation Master Plan depends on the number, type, and use of computer systems. For a facility with 30 computer systems, a Validation Master Plan with an inventory might take about 3 to 5 days to draft. The length of the document is likely to be between 5 and 15 pages, depending on its detail. The layout of an example Validation Master Plan is shown in Appendix 6B.

When writing a Validation Master Plan, it is important to minimize the duplication of information between it and other documents. Sufficient information must be provided so that personnel not directly involved in the planning can understand the validation approach in general. Clarity is important because Validation Master Plans may be examined by GxP regulators.

Validation Master Plans should be straightforward, easy-to-read documents written in the vernacular and as far as possible free of irritating jargon. Needless complexity usually retards the review process. Beware of demanding too many approval signatories in the interests of political correctness — it can make the review process cumbersome! Indeed, only those signatories whose presence lends weight to the technical or quality value of the document should be included, and these persons should then be able to justify the plan to a regulator on request.

The Validation Master Plan must be reviewed and approved before use. They are regulatory documents demonstrating an organization's intention to validation of one or more computer systems. At a minimum, a Validation Master Plan should be signed and dated by the Site Director/Area Manager, and Site/Area Quality Assurance Manager, and local Validation Manager.

Recent Inspection Findings

- There were no original planning … documents included in the validation materials for the programs. [FDA Warning Letter, 1998]
- Completion dates in Validation Master Plan inadequate to assure equipment/systems appropriate to intended use. [FDA 483, 2002]
- Validation Master Plan allowed only a single signature for both Validation and QC. [FDA 483, 2002]

VALIDATION PLAN

The purpose of the Validation Plan is to provide a clear strategy for the validation exercise based on risks arising from a number of factors:

- Functional criticality of application
- Size, complexity, standardization (maturity) of application
- Capability of organization (including supplier)
- Degree of customization and configuration

The Validation Plan and Validation Report are often some of the first documents to be examined by GxP regulatory authorities inspecting a pharmaceutical or healthcare facility.[7] Figure 6.4 illustrates this relationship. Validation Reports are discussed in more detail in Chapter 11. Without a plan to direct and organize validation activities, a chaotic or ineffective validation is likely to pass unnoticed. GxP regulatory authorities have frequently found projects where members were incorrectly assuming that somebody else was providing key validation evidence. (*Everybody thought that Somebody was validating it, but actually it was Nobody!*)

Contents of a Validation Plan

Validation Plans specify how a pharmaceutical or healthcare company intends to satisfy the GxP requirements affecting a single computer system or group of associated computer systems. Validation Plans should:

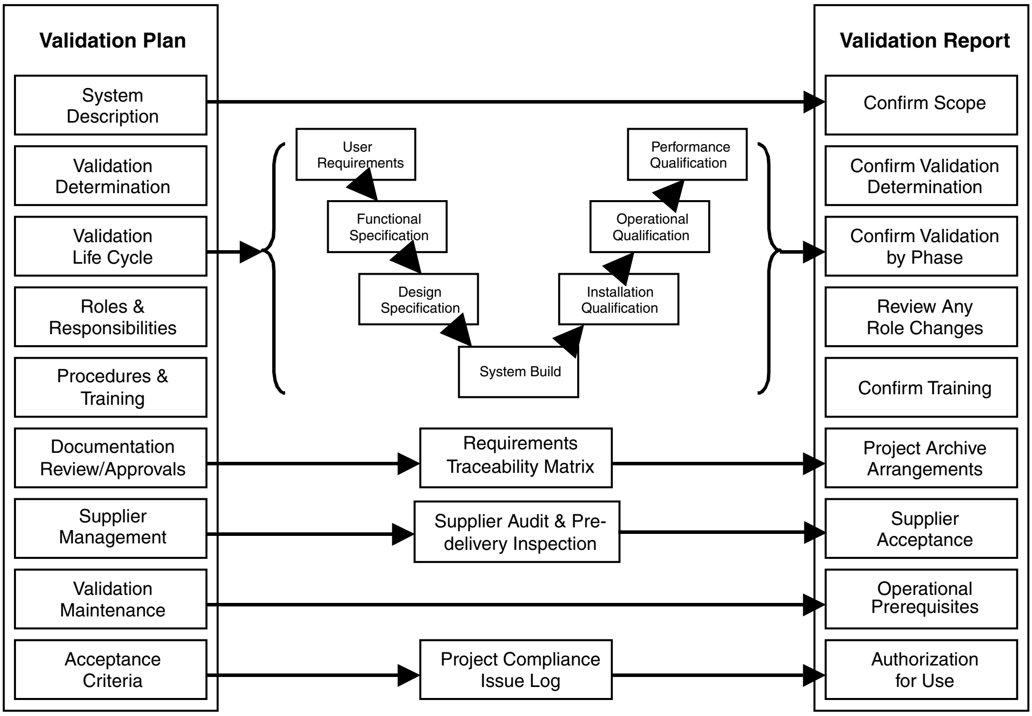


FIGURE 6.4 Relationship between Validation Plans and Reports.

- Identify the computer system being validated
- Give any relevant background information
- Reference validation procedures to be used
- Define validation package deliverables
- Specify review and approval responsibilities
- Identify personnel assigned to the project together with an indication of their competency to participate, and any training requirements
- Indicate project milestones

The depth and scope of validation depends on the degree of customization, complexity, and criticality of the computerized application.[6,8] The URS can provide useful information when determining the most appropriate approach to validation.

Structure of a Validation Plan

The following sections are suggested for a Validation Plan (based on the GAMP Guide[6]).

- Introduction
- System Description
- Validation Determination
- Validation Life Cycle
- Acceptance Criteria
- Roles and Responsibilities
- Procedures and Training
- Documentation Review and Approvals

- Supplier and Subcontractor Management
- Support Program for Maintaining Validation

In addition, it is often useful to include a glossary so that a terminology used by everyone during validation is unambiguous and clear. Although a common set validation terminology is now widely used, the precise understanding of these terms can vary dramatically. Just as with the Validation Master Plan, as we mentioned earlier, not all terms routinely used within the organization will be familiar to external personnel. Establishing a definitive understanding of key terms provides a point of reference when different perspectives arise.

Introduction

Validation Plans often begin by citing the authority under which they have been issued (perhaps a Validation Master Plan, or on the personal authority of a senior manager). They preview the validation requirements for the project.

System Description

A brief system description covering the system's history (whether it is an entirely new system, a replacement system, or an existing system) and its business purpose is necessary. GxP regulatory authorities expect replacement systems to be as least as reliable as their predecessor manual or automatic system.[9] The aim is to provide a management overview of the project boundaries, defining what is and what is not included within the scope of work. The system description may be supplemented later by a separate System Overview document, especially for larger systems (see Chapter 8).

Validation Determination

The Validation Plan should refer to, or include, a brief description why a particular computer system is being validated. The role of Validation Determination Statements was described earlier in this chapter. It is useful for the reader to have this placed early in the Validation Plan to put the validation into context. This information will be supplemented later with a GxP Assessment (see Chapter 7) in terms of detail once the User Requirement Specification is available.

Validation Life Cycle

The validation activities are laid out with a description of any issues for specific GxP consideration. The validation life cycle has already been described in Chapter 4 and consists of:

- Requirements Capture and Supplier (Vendor) Selection
- Design and Development
- Coding, Configuration and Build
- Development Testing
- User Qualification and Authorization to Use
- Operation and Maintenance
- Phase-Out and Withdrawal

The validation life cycle adopted should be customized for each individual project. Often there are opportunities for various phases to be carried out concurrently. However, the phases must be clearly defined in advance, with acceptance criteria and appropriate verification to ensure that each phase is completed in a state of control. This must be specified and documented in the Validation Plan. The following chapters examine in detail the activities associated with the life cycle outlined above. The resulting validation process — tailored for the computer system being validated — is in essence the Validation Strategy being adopted.

Acceptance Criteria

A clear definition of what criteria must be met for acceptance of the computer system must be defined. Reference to a Validation Report alone is not sufficient. The criteria to be used by the Validation Report in measuring the computer system for acceptability as validated must be stated, together with the manner in which compliance failures logged during the project are resolved.

Roles and Responsibilities

The roles and responsibilities divided and allocated between both the pharmaceutical or healthcare company and any suppliers need to be defined. A simple organizational chart can prove very useful when describing such interrelationships between project staff. Résumés of project staff should be collated and include references to their education and degrees obtained, professional certificates, previous job titles and responsibilities, but most importantly to their *competency* to fulfill their roles. Résumés for temporary staff, such as supplier personnel, are often included as an appendix to the Validation Plan or associated Validation Report. Some pharmaceutical and healthcare companies may prefer to use a central management system to collate the résumés of their own permanent staff.

Procedures and Training

The procedures to be adopted and the documentation to be produced must be identified. Planned deviations from validation procedures should be defined in advance within the Validation Plan to demonstrate management's acceptance of, and continued control over, the project. Key procedures in addition to those required to prepare the life-cycle documents include change control and document management. Training requirements against project procedures should be identified. Although already stated earlier, it is worth repeating that it is important to recognize that the competency of individuals is critical to a successful validation exercise.

Documentation Reviews and Approvals

Approval signatories for validation documentation should be specified. An individual's job title may not necessarily reflect the function performed and the responsibility shouldered by a person's signature. On numerous occasions, GxP regulatory authorities have challenged individuals who have signed validation documents, discovering to their alarm that the signatories had misunderstood their full responsibilities and performed less thorough reviews than were required.

Supplier and Subcontractor Management

Validation procedures and documentation to be provided by suppliers should be distinguished from the pharmaceutical or healthcare company's own documentation. The pharmaceutical or healthcare company's dependence on suppliers should be explained, identifying any necessary Supplier Audits (or reviewing the results of audits already conducted). The awareness that quality can only be built into products, not tested in retrospectively, has led the pharmaceutical and healthcare industries in recent years to place increasing emphasis on this activity. Regulators have endorsed this healthy trend, reflecting an increased technical understanding all around of the realities of complex product development. Another important factor to consider is who approves and accepts work as satisfactorily completed. Any requirements for monitoring a supplier's work should be defined.

Support Program for Maintaining Validation

A support program to maintain validation should be identified. It should include procedures for change control, maintenance and calibration, security practices, contingency planning, operating procedures, training, performance monitoring, and periodic review. Operation and maintenance is discussed further in Chapter 12.

Preparation of a Validation Plan

An experienced validation practitioner can produce a Validation Plan for a computer system (usually 10 to 15 pages long) in about 3 days. An example Validation Plan layout is illustrated in Appendix 6C. The structure of Validation Plans must be scaleable to fit the system or software being validated. For large projects, configuration management and functional test planning may be split out into separate documents. Validation Plans can also make use of appendices for project milestones and résumés of project team members.

The Validation Plan must be reviewed and approved before issue. They are regulatory documents specifying the quality and compliance controls used to manage the deployment of a computer system. At a minimum, a Validation Plan should be signed and dated by the System Owner as well as by the Quality and Compliance representative.

The time to review and issue the Validation Plan will vary between different organizations. It is primarily dependent on the number of people involved in the review process. Validation Plans should be clear, easy-to-read documents and, as we mentioned in the context of Validation Master Plans, written in the vernacular and free (to the greatest extent possible) of irritating jargon. If review is slow or protracted, it may be because it is too complex or imposes a cumbersome review-and-approval signatories process. Each signature should genuinely add value to the document. Signatories should not be added solely for political representation (to share the glory) or as a fail-safe (to share the blame). Every signatory must be able to explain and justify the plan to an inspector if required.

Maintaining Validation Plans

Validation Plans should be maintained to reflect changes in project activities. Specific revisions to plans should be considered when transitioning between major project phases:

- Requirements Capture and Supplier (Vendor) Selection
- Design and Development
- Coding, Configuration, and Build
- Development Testing
- User Qualification and Authorization to Use

Thereafter, Validation Plans should be updated or superseded by new plans when the architecture of the computer system changes from the original scope of validation; there has been a significant change in how the computer system is used compared to the original validation; there has been a change in operation and maintenance standards; or when a revalidation exercise has been initiated.

Supplier Relationships

The ultimate responsibility for the validation of a computer system provided by vendors, system integrators, and service providers lies with the user. That said, certain elements remain the province of the supplier, while others will be under the control of the user. Validation Plans must address the requirements relating to both.

Commercial Off-The-Shelf Computer (COTS) Systems

The validation documentation set provided by a supplier of COTS will be standard rather than being tailored in advance with the users. Nevertheless, pharmaceutical and healthcare companies should map supplier documentation to validation requirements in an attempt to satisfy the latter. This mapping may be quite complex, and a direct correlation impossible. However, at a minimum the mapping should cover:[10]

- Management of software development personnel
- Formal software development life cycle and associated documentation
- Programming standards
- Software fault management
- Documentation management
- Configuration control
- User manuals
- Release notes
- User support
- Upgrade provision mechanisms

The mapping exercise is typically conducted by pharmaceutical and healthcare companies as a desktop exercise, although a Supplier Audit should embrace most of these if conducted competently and thoroughly. The auditors chosen to conduct supplier audits should be formally qualified for their role and therefore able to demonstrate competency just as GxP regulations demand for all other participants in the validation exercise, including the developers. Pharmaceutical and healthcare companies are increasingly seeking supplier auditors qualified either to ISO 9001:1994 (TickIT) or preferably to the current standard ISO 9001:2000 through the International Register of Certificated Auditors (IRCA). Other internal accreditation measures may be sufficient for internal auditors. Validation strategies are discussed later in Chapter 14, specifically the X-Model for validating COTS software.

If the supplier's document set is insufficient to support the pharmaceutical or healthcare company's validation of the computer system, and the deficiencies cannot be remedied, then the system should not be used.

Contracted (Commissioned) Systems

Computer systems supplied by third parties (often known as *system integrators*) to specific user requirements should be validated entirely prospectively along the same lines that pharmaceutical and healthcare companies take for in-house developments. This, of course, requires that validation requirements are clearly identified and understood by the supplier at the outset of a project.

The definition of supplier validation activities and documentation should be embedded in contractual agreements. In addition, suppliers should agree to potential inspection by GxP regulatory agencies and Supplier Audit by pharmaceutical and healthcare companies. Supplier Audits can be conducted by the pharmaceutical or healthcare company's own personnel or, if this would compromise the supplier's commercial interests, by an independent software quality assurance auditor, consultant, or validation expert employed by the pharmaceutical or healthcare company. Auditors must be suitably qualified, for example by independent certification by examination to the quality system standards such as ISO 9001:2000. Supplier Audits are discussed in detail in Chapter 7.

Availability of Software and Reference Documentation

Another important aspect to take into consideration is access to proprietary source code and associated documentation.[11] All custom (bespoke) application-specific software and reference documentation should be available. COTS software and development documentation, however, is not always available.

Regulators now expect formal access agreements to be established (e.g., escrow accounts) where access to application software and associated reference documentation is restricted. It is generally accepted that formal access agreements for operating systems, compilers, and system software are unnecessary since the ubiquity of such systems provides adequate implied evidence of their fitness for purpose. Access agreements should be kept up to date; current and historical software versions should be covered by their respective documents and revisions.

Copies of the software and documentation must be retained within safe and secure areas, protected within fireproof safes. The duration of storage of legacy software and documentation needs to be defined. Guidance on this is given in Chapter 12. However, it must be borne in mind that should a supplier cease to trade, and the escrow agreement is invoked to secure access to the code, it is highly unlikely that this will prove to be an acceptable basis for the ongoing long-term use of the computer system. Support for code in these circumstances is fraught with difficulty. Pharmaceutical and healthcare companies have hitherto rightly concluded that there is far less risk shouldered by premature retirement and replacement of such systems rather than attempting to support them in the face of insolvency and lack of support on the part of their developers.

Recent Inspection Findings

- Validation Plan for XXXX not performed in accord with cGMPs: plan lacked number of qualified personnel, completion dates inadequate to assure valid performance of manufacturing processes, validation strategy did not contain procedures, standard protocols, specific requirements. [FDA 483, 2002]
- Validation Plan contained no instructions pertaining to what constitutes acceptable test results. [FDA 483, 2002]
- No information on qualifications of those reviewing and approving protocols/reports. [FDA 483, 2002]

VALIDATION STRATEGY

Validation approaches for both the hardware and software of computer systems need to be considered:

- Is hardware/software bespoke, COTS, or a combination?
- Is there any hardware configuration?

The fitness for purpose of a proposed solution should be assessed and any history of usage in similar applications considered when determining the validation strategy. A holistic perspective should be taken with computer systems comprising components found in multiple categories of software and hardware.

Approach to Hardware

The validation approach must reflect whether the associated computer system hardware is a unique combination of components put together by the pharmaceutical or healthcare company or preassembled as a standard product by the original equipment manufacturer (e.g., PCs packaged with laboratory analytical equipment, or production equipment containing an embedded control system). It must also reflect any hardware configuration (e.g., switch settings). Bespoke items of hardware must have a design specification and be subjected to acceptance testing. A Supplier Audit should be performed for bespoke hardware development. The design of standard hardware products must also be documented. In this regard, reference may be made to the hardware manufacturer's data sheet or other specification material as long as its source is recorded, typically the supplier's name and address. It should also include details of any hardware configuration. Complete systems consisting of a unique assemblage of hardware from various sources must be checked to ensure that interconnected hardware components are compatible with one another. Any hardware configuration must be defined in the design documentation and verified in the IQ. The model number, version number, and, where available, serial number of each component of the assemblage must be recorded. Preassembled hardware that is sealed does not have to be dismantled as such action often invalidates the manufacturer's warranty. Rather, the hardware details should be obtained from the system data sheet or other specification document.

Approach to Software

GAMP Category 1 Software: Operating Systems

Within any project in which validation of the application forms part of the process, it is not normally necessary to attempt to validate established, commercially available operating systems. Because of their ubiquity and because they are exercised every time the applications installed upon them are used, these should be considered to be already validated. Only the name and version number need to be recorded in the hardware acceptance tests of equipment IQ. New versions of operating systems should be reviewed prior to use, and consideration given to the impact of new, amended, or removed features on the application. This might lead to formal retesting of the application, particularly where a major upgrade of the operating system has been necessary.

Summary of validation requirements:

- Specify version for installation.
- IQ — check version installed.

GAMP Category 2 Software: Firmware

The name, version number, and any configuration should be recorded in the equipment IQ. Functionality is verified during OQ. Calibration should be conducted as required. The unintended and undocumented introduction of new versions of firmware during maintenance must be avoided through the application of rigorous change control. The impact of new versions on the validity of the IQ documentation should be reviewed and appropriate action taken.

Summary of validation requirements:

- Specify version for installation.
- Specify scope of use.
- Review and accept standard instrument documentation.
- Specify any configuration parameters.
- Define calibration procedure in accordance with supplier's recommendations.
- IQ — check version installed with configuration.
- IQ/OQ — calibrate instrument for operation.
- PQ — define calibration schedule.

GAMP Category 3 Software: Standard Software Packages

These do not normally need extensive validation if the version to be acquired has already been exposed to the marketplace for an extended period. However, new versions are a different matter and should be treated with caution. Validation effort should concentrate on functionality, critical algorithms and parameters, data integrity (security, accuracy, and reliability), and operational procedures. Change control should be applied stringently since upgrading these applications, while initially easy, can turn out to be painful. User training should emphasize the importance of change control and the validated integrity of these systems. A Supplier Audit may be an appropriate defensive measure for critical applications.

Summary of validation requirements:

- Specify version for installation.
- Specify scope of use.
- Develop and approve user procedures.
- Develop user training materials.
- Review and accept software package documentation.
- IQ — check version installed.
- OQ — verify any data load.
- OQ — verify general operation as used.
- PQ — establish ongoing reliable operation.

GAMP Category 4 Software: Configurable Software Packages

These systems permit users to develop their own applications by configuring or amending predefined software modules. In many cases the development of additional new application software modules may be needed. Clearly therefore, each application (of the standard product) is unique.

Particular attention should be paid to any additional or amended code and to the configuration of the standard modules. A Source Code Review of the modified or added code (including any algorithms in the configuration) should be undertaken. In addition, for large, complex, or critical software applications a Supplier Audit is essential to determine the level of quality and innate structural integrity of the standard product. The audit must recognize the possibility that the development of the standard product might have involved a prototyping methodology without any eventual users being involved at all. GxP standards require that the development process is controlled and documented. A Validation Plan should be prepared to document precisely what activities are necessary to validate an application, based on the findings of the audit and on the complexity of the application.

Summary of validation requirements:

- URS — specify scope of use.
- Supplier Audit for complex and/or critical software packages.
- Functional Specification for configuration in context of software package.
- Develop and approve user procedures.
- Develop user training materials.
- Review and accept software package documentation.
- Hardware and Software Design for bespoke code/macros.
- Source Code Review for bespoke code/macros.
- Specify version of software package for installation.
- IQ — check version of software package installed.
- OQ — verify any data load.
- OQ — verify general operation as used.
- OQ — comprehensive user acceptance of configured functions.
- PQ — establish ongoing dependable operation.

Testing should cover positive functional testing based on defined user operation (it does what it should do) and risk-focused negative functional testing (it does not do what it should not do where risk assessment suggests a vulnerability).[12]

GAMP Category 5 Software: Custom (Bespoke) Software

For these systems, the full life cycle defined in Chapter 4 should be followed for all parts of the system. An audit of the developer is essential to measure their development capability maturity and to examine their quality system. A Validation Plan should then be prepared to document precisely what activities are necessary, based on the insights gleaned in the audit and on the complexity of the proposed bespoke system.

Summary of validation requirements:

- URS — full specification.
- Supplier Audit.
- Functional Specification.
- Develop and approve user procedures.
- Develop user training materials.
- Hardware and Software Design — program specifications as necessary.
- Extensive Programming and Source Code Reviews.
- Unit/Module Testing.
- Integration/System Testing.
- IQ — check installation against specification.
- OQ — verify any data load.
- OQ — comprehensive user acceptance.
- PQ — establish ongoing dependable operation.

Testing should cover comprehensive positive functional testing (it does what it should do), and risk-focused negative functional testing of all custom software (it does not do what it should not do where the risk assessment suggests a vulnerability).[12]

Optional Extras for GAMP Category 3, 4, and 5 Software

Additional requirements may be necessary as appropriate to the project, including

- Data migration protocols, records, and reports
- System installations for development environments
- Strategy for phased deployment

Criticality–Impact Analysis

Senior managers may wish to consider using a Criticality and Impact Analysis on functions and components of a computer system to support the planning of individual validation projects. The rigor of validation should reflect the direct or indirect impact of the computer system on drug or healthcare product development, manufacturing, or distribution. This concept is illustrated by Figure 6.5 (based on the *Baseline Pharmaceutical Engineering Guide*[13]). System components are only permitted to exist in three of the four boxes in the matrix — critical components cannot by definition have an indirect impact. Chapter 7 discusses the use of GxP Assessments to determine criticality. Direct impact systems have one or more critical components. Indirect impact systems cannot have critical components. Direct impact systems may have noncritical components.

Components can themselves be treated as systems and further subdivided into components, and so on. The level of granulation need not be exhaustive; a common-sense approach should prevail. Usually only one or two stepwise refinements are appropriate.

Some practitioners have suggested that managing two levels of assurance increases complexity and thus aggravates the likelihood or error. Others maintain that this is the only way to control compliance costs, especially on large or highly integrated computer systems. If criticality–impact

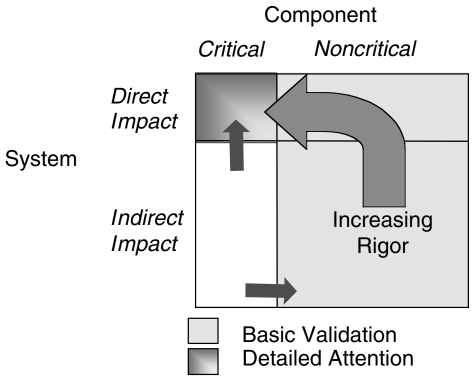


FIGURE 6.5 Criticality and Impact Analyses.

analyses are used, these can furnish important evidence to a regulatory authority demonstrating why certain aspects of a computer system cannot affect drug product quality. They can also be used to show the level of assurance needed for aspects that do affect drug product quality.

Managing Compliance Issues

Potential or actual noncompliances that arise during the course of a project need to be logged and managed. The process used to manage these issues should be defined and referenced in Validation (Master) Plans.

Many pharmaceutical and healthcare companies create a Project Compliance Issues Log during projects to track compliance issues. The structure of such a log should include (based on *Managing Successful Projects with PRINCE2*[14]):

- Project issue number
- Author and date identified
- Description
- Resolution (change control reference or justification for no action)
- Status (outstanding, in progress, or closed)

Larger validation exercises might consider implementing a RAID (Risk, Actions, Issues, Decisions) Log rather than confine themselves to the simpler Project Compliance Issue Log.

A Change Control Process that initiates appropriate corrective action or delivers a documented rationale justifying the acceptance of the defect or characteristic causing the noncompliance without further action should be used. Change Controls may prompt revision of SOPs, revision of documents, further training, or other activity.

The aim should be to complete open Change Controls wherever possible within the project. If a Change Control cannot be completed within the project, then a rationale should be prepared justifying completion of action retrospectively after the project closure.

Validation Reports are expected to review the compliance matters raised during the project and itemize any outstanding corrective actions with a corresponding justification of the state of affairs. The Issues Log should be retained as part of the Validation Package and the responsibility for any outstanding corrective actions handed over to the organization managing the ongoing support of the computer system. Periodic reviews should verify that outstanding corrective actions are implemented in a timely manner.

REFERENCES

1. FDA (2002), *General Principles for Software Validation (Medical Devices): Guidance for Industry*, Final Guidance, January.
2. PMA (1986), Validation Concepts for Computer Systems used in the Manufacture of Drug Products, in *PMA Proceedings: Concepts and Principles for the Validation of Computer Systems Used in the Manufacture and Control of Drug Products*, Pharmaceutical Manufacturers Association.
3. Van Vliet, H. (2000), *Software Engineering*, 2nd Edition, John Wiley, New York.
4. Simmons, T. (2001), *Producing a Quality Plan*, Pharmaceutical Automation Updates, Sue Horwood Publishing, West Sussex, U.K.
5. SQA (2001), *Risk Assessment/Validation Priority Setting*, Computer Validation Initiative Committee, Society of Quality Assurance, Charlottesville, VA.
6. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (www.ispe.org).
7. Trill, A.J. (1996), *Regulatory Requirements for Computer Validation,Computer Systems Validation: A Practical Approach*, Management Forum Seminar, London, March 26 and 27.
8. ICH (2000), *Good Manufacturing Practice Guide for Active Pharmaceutical Ingredients*, ICH Harmonised Tripartite Guideline, November 10.
9. European Union (1993), *Annex 11 — Computerised Systems*, European Union Guide to Directive 91/356/EEC.
10. ACDM/PSI (1998), *Computer Systems Validation in Clinical Research: A Practical Guide*, Version 1.1, December.
11. OECD (1995), *The Application of the Principles of GLP to Computerised Systems*, No. 10 OECD Series on Principles of Good Laboratory Practice and Compliance Monitoring, GLP Consensus Document, Environmental Monograph No. 116, Organisation for Economic Co-operation and Development Environmental Directorate, Paris.
12. GAMP Forum (2003), Risk Assessment for Use of Automated Systems Supporting Manufacturing Process Part 1 — Functional Risk, *Pharmaceutical Engineering*, May/June.
13. ISPE (2001), *Baseline Pharmaceutical Engineering Guide: Qualification & Commissioning*, International Society of Pharmaceutical Engineering, Tampa, FL.
14. CCTA (1999), *Managing Successful Projects with PRINCE2*, Central Computer and Telecommunications Agency, The Stationery Office Books, London.

APPENDIX 6A
EXAMPLE VALIDATION DETERMINATION STATEMENT

Computerized System	Determination Reference No.: VDS/001
System Identification:	**PLC/002/SMA**
System/Equipment Name	Sterile Manufacturing PLC
System Name/Equipment No.	Number 2
Location/Department	Manufacturing Unit A
System/Equipment Used By:	Production
System/Equipment Used For:	Autoclave control

Justification for whether or not validation is required
(single line strike out inappropriate positive or negative inclination in all sentences below)

- The system is used to monitor, control, or supervise a drug manufacturing or packaging process
- The system manipulates data, or produce reports, to be used by quality related decision authorization/approval processes
- The system is used for batch sentencing or batch records
- The system manages, stores GxP records

Acknowledgment of Validation Requirements
It is the responsibility of the group(s) using, developing, and supporting the application to notify the Computer Validation Department of any changes in the use of an application that might impact compliance with GxP regulations.

	Name and Title	**Signature**	**Date**
AUTHOR (including Department)			
APPROVED BY User/Project Manager			
APPROVED BY Quality Assurance			
APPROVED BY Computer Validation			

APPENDIX 6B
EXAMPLE CONTENTS FOR VALIDATION MASTER PLAN

Introduction and Scope

- Author/organization
- Authority
- Purpose and scope
- Contractual status of document

Organizational Structure

- Resource allocation: organizational responsibilities

GxP Criticality Assessment Process

- Define basis of determining criticality
- Justify any prioritization

Validation Strategy

- Description of validation life cycle to be adopted (reference to relevant standards)
- Approach to managing suppliers and subcontractors
- A statement to the effect that the computer system will only be authorized for use once satisfactorily validated

Change Control

- Description of change management process to be adopted

Procedures and Training

- Identify validation SOPs to be adopted
- Commitment to training

Document Management

- Definition of how documents will be managed and controlled

Timeline and Resources

- Target completion date
- Interim milestones as appropriate

References

Appendices

- Glossary
- Others

APPENDIX 6C EXAMPLE CONTENTS FOR VALIDATION PLAN

Introduction

- Author/organization
- Authority
- Purpose
- Relationship with other documents (e.g., Validation Master Plans)
- Contractual status of document

System Description

- Define boundaries of system (e.g., hardware, software, operating system, network)
- Constraints and assumptions, exclusions and justifications

Validation Determination

- Rationale behind validation requirement (may be reference to Validation Determination Statement)

Validation Life Cycle

- Outline of life cycle being undertaken (reference to validation standards)
- Approach to validation for different hardware and software categories (see Chapter 5)

Acceptance Criteria

- A statement to the effect that the computer system will only be authorized for use once satisfactorily validated
- Description of how project compliance issues will be managed

Role and Responsibilities

- Resource allocation: organogram and role descriptions
- CVs (qualifications and experience)

Procedures and Training

- Identify validation SOPs to be adopted
- Training requirements and Training Records

Document Review and Approvals

- List of documents to be prepared
- Review and approval set in accordance with roles and responsibilities

Supplier and Subcontractor Management

- Supplier responsibilities
- Anticipated supplier audits
- Supplier documentation controls

Support Program for Maintaining Validation

- Description on how the validation status will be maintained

References

Appendices

- Glossary
- Others

7 Requirements Capture and Supplier (Vendor) Selection

CONTENTS

The second life-cycle phase addresses the capture of requirements and the selection of suppliers. It entails the following main activities, usually conducted sequentially: User Requirements Specification (URS), GxP Assessment, Supplier Selection, and Supplier Audits. Getting this phase right is crucial to ultimate success. The issues that have to be managed can be summarized as follows:

- Suppliers are often expected to work with incomplete or ill-defined user requirements.
  - Users frequently do not spend enough time and effort evaluating and documenting their requirements.
  - Commercial pressures tempt many suppliers to agree to a contract but delay working out the details until later.
- Suppliers are typically audited late in the user's selection process.
  - There is no strictly defined standard to which suppliers are audited; it is common for different auditing companies to apply various GxP compliance expectations.
  - Suppliers often adopt the most lax standards they believe acceptable (i.e., minimal quality assurance).
- A supplier's ability to invest in quality systems supporting GxP compliance may be limited by
  - GxP regulations applying only to a small proportion of its market
  - Prices being driven down by competitive pressures in its marketplace

The fundamental challenge is to understand how a cost-effective balance between a supplier's quality system and the user's GxP compliance requirements can best be reached.

USER REQUIREMENTS SPECIFICATION (URS)

The business objective to be met by the computer system is expressed in a URS. This must provide a firm foundation for the project. A URS may exist as a single document or as a suite of documents. Supplementary documentation might include Business Requirements and Technical Requirements. However, the requirements are collated if they are not precise, concise, and complete, major problems appearing later in the project are almost a certainty.

The URS should not delve into *design* detail that should be postponed to the Functional Specification. This may be difficult when specifying bespoke or customized systems, as the design is often anticipated when developing the URS. It may also be tempting to include far too little detail in a URS when a COTS product solution is anticipated.

A multidisciplinary team including production, engineering, and quality assurance staff should draft the URS. End users should be involved as soon as possible and approve the requirements before design and development begins.

CONTENTS

The level of detail contained in the URS varies depending on its relationship with the Functional Specification (FS). Information that ought to be considered for inclusion in a URS is provided in Appendix 7A. Diagrams should be used wherever possible to promote greater understanding and clarity. Spreadsheets are also useful for defining data and clearly showing omitted information.

Some pharmaceutical and healthcare companies give their suppliers a relatively free hand, while others may impose particular equipment and design requirements. A closer examination of any imposed constraints will often expose the fact that preferred equipment and design are not, in fact, critical. Indeed, a measure of choice in the details could be delegated to the supplier and recorded in the design documentation.

The operational requirements section is typically the largest portion of the URS. This information will be presented in the form of textual descriptions, flowcharts, state transition diagrams, or some other similar illustrative form; diagrammatic information is often demoted to appendices. It should include:

- Identification of halt states, error routines, and error recovery
- Specification of failure modes to protect both the plant and the personnel
- Description of any special calculation requirements

Other important sections in the URS include access security, the human interface, and the process interface. Specifically:

- Access security should be considered a means of reducing the risk of unauthorized access inadvertent modification, or data loss or corruption.
- Security may be achieved through the use of passwords, key switches, or other more sophisticated mechanisms such as those based on biometric features.
- The human interface definition should include screen layouts (or prototypes where these have not been finalized) and other requirements such as configuration pages, alarm pages, mimics, and system responses to data entry errors.
- The minimum information for the process interface includes equipment tag numbers, unit references, descriptions, input/output (I/O) types, and any special treatment such as segregation requirements.
- Consideration should be given to the extent of any interface work needed to commission the new system and to identify any changes of functionality in the existing system.

Those defining the URS should consider how the requirements might be tested (see Figure 7.1). Alternative phrasing of requirements could considerably clarify the objectives of Development Testing and User Qualification. In response to a poorly written URS, some suppliers have suggested using word searches on “must,” “shall,” and “will” to determine the minimum set of requirements. All other expressions would be taken to define optional (“nice to have”) features. It is easy to see how vague expressions of specifications lead to defective acceptance criteria! Test protocols and their relationships to the URS, FS, and other development documentation are discussed in detail later in this chapter.

Within the URS, mandatory and desirable requirements must be distinguished (see Table 7.1). Features that are necessary to validate a computer system must always be considered as critical and hence mandated requirements. Desired features should be prioritized in order of their

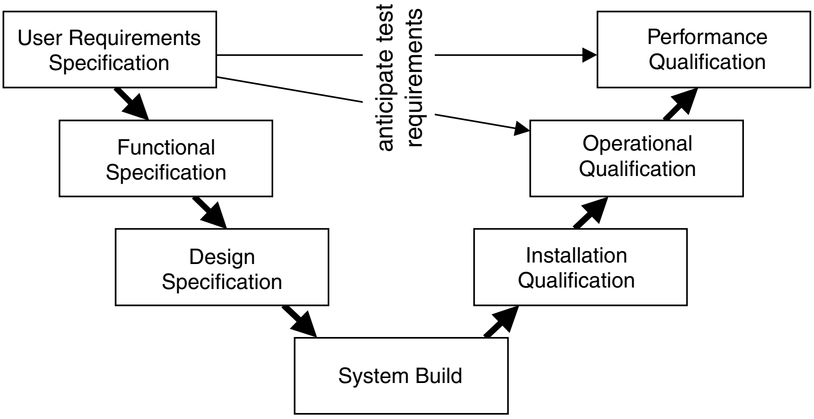

User Requirements Specification
Functional Specification
Design Specification
System Build
Installation Qualification
Operational Qualification
Performance Qualification
anticipate test requirements

FIGURE 7.1 User Requirements Anticipate Testing Requirements.

TABLE 7.1
Example of Requirements Numbering in a URS

Specification Reference	Requirement	Priority (M/D)	Criticality (C/N)
4.6.1	The XXXX computer system is able to allocate either any or all of the system configurable access rights to each of three user levels as defined in Appendix 1.	M	C
4.6.2	Allocation of all configurable access rights to individual users requires on-screen confirmation by System Administrator.	D	N

Note: M = Mandate, D = Desirable, C = Critical, N = Noncritical.

relative importance. The FS should include a table showing how it complies with the URS. A list of known deviations from the current release of the URS should be highlighted. To make this cross-referencing easier to administer, each URS requirement should be allocated a unique reference number. The GAMP Guide recommends that each requirement statement is no longer than 250 words to aid traceability.[1] Any assumptions made by the FS relating to such URS deviations should be readily identifiable so that ensuing misunderstandings can be clarified before the project progresses too far. Ian Johnson, now at AstraZeneca, recalls an instance of an extractor fan fitted on a powder-filling machine for operator protection. The machine was controlled by a Programmable Logic Controller (PLC) that monitored the extract from the fan and displayed an alarm to the operator if the extractor had failed.[2] However, the supplier providing the application software had not appreciated the potential danger of powder in the operator's working environment. They had accordingly programmed the powder-filling machine to continue functioning after the alarm of extract failure had been displayed. The URS should have specified the need for an interlock so that if the operator did not notice the alarm and intervene, the powder-filling machine would stop.

Requirements, if appropriate, can be developed, approved, and released incrementally, but care should be taken that interactions and interfaces between software (and hardware) requirements are properly reviewed, analyzed, and controlled.[3] While incomplete or ambiguous requirements present an opportunity to proactively manage future clarification of requirements, more often than not they hinder effective design. Sam Clark, a former FDA investigator, cited an example of a poor requirement:

> *A novice must be able to use the system in a simple manner and a sophisticated user in a sophisticated manner.*[4]

A URS is easy to criticize, but to write it is much more difficult than is often believed. The first draft of the document is inevitably unstructured as authors tend to include information in the order it comes to mind. The creation of a standard format helps, but most authors still customize the layout to reflect their own particular viewpoints. Project managers must accept and plan for rearranging the URS so that a more readable and useful document evolves. Revisions to the URS must be subject to version control so that the project team is kept abreast of updates.

Diligent effort at the early stages will be abundantly rewarded by the avoidance of failures later in the project. The cost of retrospective modifications later in the life cycle can be 10 or 20 times that of instituting the amendment at this stage. Getting it right first time is the cheapest way of doing anything, in the long run. Once the requirements have been agreed upon, they must be documented in a manner understandable by both users and development staff. The user may ask a supplier for assistance in producing this document.

Electronic Record/Signature Requirements

Functionality that handles electronic records subject to specific regulatory requirements should be specifically identified in the URS. Individual records requiring ERES controls should be defined. The following aspects should be outlined for either the entire set of electronic records or a subset or for individual electronic records (a detailed discussion is provided in Chapter 15), as appropriate:

- The ability to discern invalid or altered records.
- The ability to generate accurate and complete copies of records in both human readable and electronic form suitable for inspection, review, and copying.
- Secure, computer-generated, time-stamped audit trails to independently record the date and time of operator entries and actions that create, modify, or delete electronic records.
- Protection and retention of records and audit trail documentation to enable their accurate and ready retrieval for review and copying throughout the records retention period.

System access in these cases must be limited to authorized individuals; authority checks must be in place to ensure that only authorized individuals can use the system, electronically sign a record, access the operation or computer system input or output device, alter a record, or perform the operation at hand. Where appropriate, operational system checks that enforce only the permitted sequencing of steps and events must be included. Use of device checks to determine, as appropriate, the validity of the source of data input or operational instruction should be present. During a single, continuous period of controlled system access, the first signing needs to verify the signatory's identity with password, while subsequent signings need verify only the password. Signings not performed during a single, continuous period of controlled system access must verify both the signatory's identity and password.

Approval and authorization of electronic records should be accomplished using electronic signatures that are unique to individual users and inextricably linked to the electronic records to which they are applied. Where electronic signatures are to be used, each instance of this should be unambiguously specified.

Signed electronic records should contain information associated with the signing that clearly indicates:

- The name of the signer
- The date and time when the signature was executed
- The meaning (such as review, approval, responsibility, or authorship) associated with the signature

These items must be included as part of any human readable form (e.g., printed) of the electronic record.

Recent Inspection Findings

- There were no written requirements/specifications at the time of validation.
- Requirements documentation was incomplete. [FDA 483, 1999]
- Necessary actions have not been predetermined and documented when responding to alarms from the XXXX … [alarm events] are not recorded … There is no secondary review of alarm events, and any corrective actions are not recorded. [FDA Warning Letter, 2000]

GXP ASSESSMENTS

GxP Assessments provide a useful tool in helping to identify where to focus attention during validation. The assumption here is that GxP processes/functions require a higher level of assurance

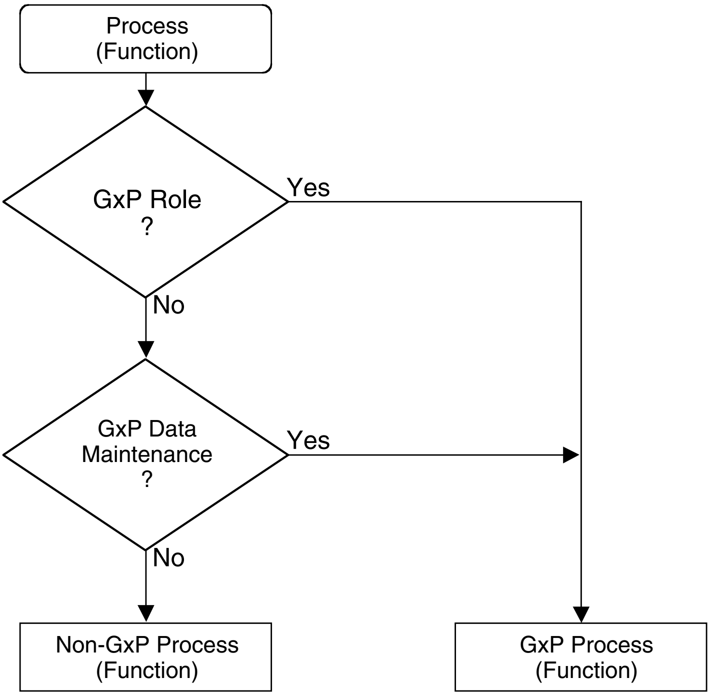


FIGURE 7.2 GxP Assessment Decision Tree.

than non-GxP processes/functions. Similarly, GxP critical components/devices require a higher level of assurance than non-GxP critical components/devices. The risk assessment process does not reduce the need for complete specifications; rather, it can be used later in the life cycle to improve the usefulness of the testing efforts.

GxP assessments are usually conducted by multidisciplinary teams with expert knowledge of regulatory requirements, the relevant process (development, manufacturing, or distribution), and the computer application.

All GxP functions, processes, components, and devices identified within the GxP assessment should be challenged as part of the Design Review. Consideration may also be given to occupational health matters such as the potential effects of the computer system and associated equipment on the personnel who may use or contact the system. GxP functionality includes the use of electronic records and signatures. Hybrid systems must be defined and subject to a verification process to determine whether or not they are robust. It is often useful for processes to be mapped, showing critical points in the process and how various computer systems support these critical process points.

Assessments are only as useful as the available information (opinion and documentation) pertaining to the computer system under scrutiny. Such information may be excessive, incomplete, inconsistent, or incorrect. Where there is insufficient information to complete the GxP Assessment, a preliminary assessment may be conducted and reviewed when further information becomes available during the later phases of the project life cycle.

Identifying GxP Processes and Functions

The decision over whether a process or function affects GxP should be dictated by its operational role and if it is used to maintain GxP master data, as illustrated in Figure 7.2. GxP roles include:

- GxP procedural controls
- GxP decisions (e.g., accept, reject, refer)

- GxP approvals and certifications
- GxP authorizations
- GxP data submissions to regulatory authority

GxP data maintenance activities include:

- Create GxP master data
- Modify GxP master data
- Delete GxP master data

Examples of GxP processes (functions) include supplier management, procurement, goods receipt, materials management, production control, quality control, batch release, distribution, recall, customer complaints, batch tracking, and compliance management (e.g., SOP management, electronic data archiving).

Examples of non-GxP processes (functions) include capacity scheduling, finance, human resources (excluding training), marketing (excluding medical information), purchasing, legal affairs, insurance management, and business reporting.

Examples of product quality GxP data include study data (e.g., stability trial data, clinical trial data, patient and animal records/results), regulatory submissions (e.g., stability data, development summary reports), analytical production data (e.g., analytical methods, quality reference data), and compliance management (e.g., indexes to archived documents/records).

Examples of manufacturing GxP data include purchase order information (order number, supplier batch number, supplier quality approval status), bill of materials information (items, quantity, units of measure, conversion factors, work centers, yield factors, critical process parameter), batch information (batch number, batch status, expiry and receipt dates, quantity, potency, conversion factors, and any special instructions or batch record annotations), user security information (name and password), warehousing information (item number, item note, location, type, quality status, shelf life, and retest days), customer order information (order number, customer address, batch number, batch status, expiry date, quantity, potency), distribution information (distributor code number, distributor address, date collected), shipping information (customer order number, customer address, shipping address, shipping notes, dispatch date, goods return note number), secondary batch traceability, inventory control, manufacturing and expiry dates, label control, critical manufacturing process parameters, environmental monitoring, and calibration and maintenance records.

Some systems such as financial management systems will have no impact on GxP unless they contain special functionality that can affect GxP data. Examples would be the use of a zero price to indicate that a product should not be supplied to a particular market/customer. Another example would be financial material reconciliation affecting batch materials usage data. Even if a computer system is not deemed wholly non-GxP, it does not imply that general quality assurance principles are no longer applicable. Good business sense dictates that a quality management approach should always be applied such as TickIT, IEEE, SWEBOK.

Identifying Critical Components and Devices

Critical components/devices are usually identified within the Architectural Design as part of Design & Development (see Chapter 8). Components or devices should be considered critical (unless there is some form of backup mechanism — backup system or procedural control — for GxP functionality) if they are:

- Used to control, monitor, or assess a quality or GxP aspect of production process, including pack integrity

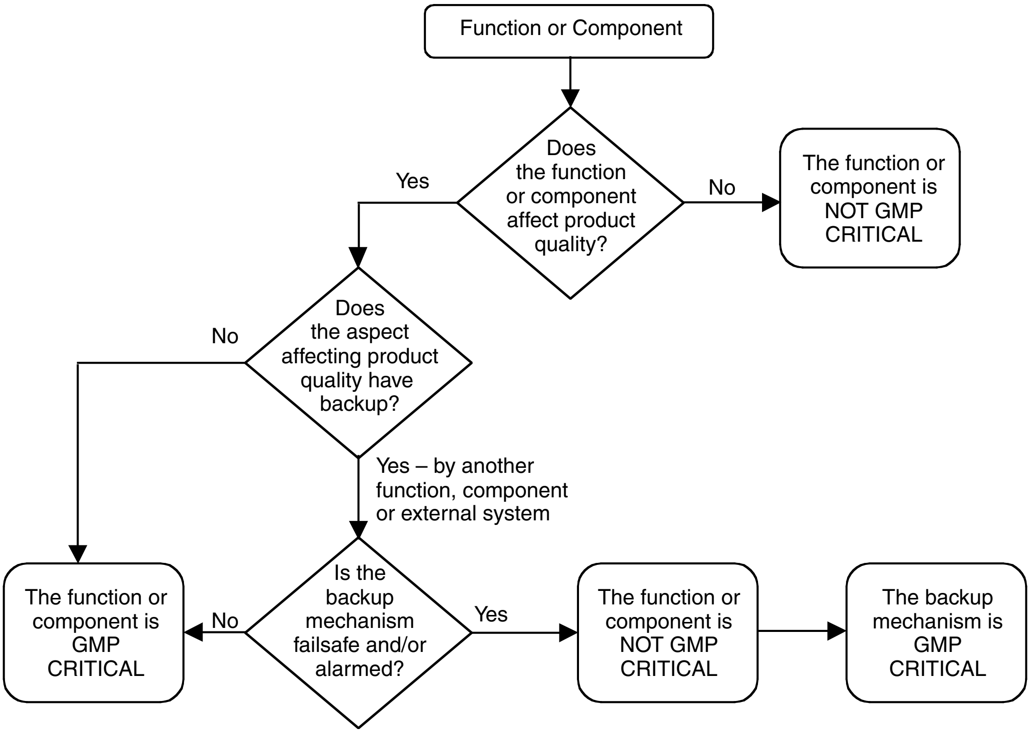

Function or Component
Does the function or component affect product quality?
Yes
No
The function or component is NOT GMP CRITICAL
Does the aspect affecting product quality have backup?
No
Yes – by another function, component or external system
The function or component is GMP CRITICAL
No
Is the backup mechanism failsafe and/or alarmed?
Yes
The function or component is NOT GMP CRITICAL
The backup mechanism is GMP CRITICAL

FIGURE 7.3 Backup Mechanism Decision Tree.

- Used to manage records related to quality or GxP aspect of production process, including pack integrity
- Used where a malfunction could result in substandard product quality, pack integrity, or a GxP control being deemed acceptable when in reality it is not
- Used to test or calibrate a critical device/component

Backup mechanisms for GxP functionality include an independent “parallel” or “downstream” component/device to detect any malfunction (e.g., independent monitoring systems). The focus for validation can then shift from the component/device to the backup system, as illustrated in Figure 7.3.[5] Where this implies implementing a backup system that is simpler to validate, the shift can relieve the burden of validating more complex components and devices. For a backup mechanism to be accepted as a validated alternative, it must be able to independently manage key quality assurance functions. Such functions include but are not necessarily limited to:

- Provision of independent system access controls
- Provision of independent functions for alarms (including management and calibration as appropriate of alarm set points and management of alarm log records)
- Provision of independent data sources for GxP records (e.g., QA monitoring and records for critical parameters, and QA investigations of out-of-specification incidents)
- Provision of data and functionality for understanding trends over short and long periods

All these functions must be managed by the backup mechanism; otherwise both the component/device and backup will require validation. Care must also be taken when implementing system changes not to inadvertently undermine the case for directing validation effort to backup mechanisms.

When conducting a review, care must be taken to address system interfaces. Computer systems receiving GxP data, even if such data is just passing through the machine for processing elsewhere, are likely to require validation as they could potentially compromise the integrity of the GxP data being transmitted.

Recent Inspection Findings

- Failure to identify and analyze the system/software critical functions. [FDA 483, 1996]
- Quality Assurance critical modules within YYY program have not been identified. [FDA 483, 1999]

SUPPLIER SELECTION

Regulatory Expectations

Pharmaceutical and healthcare companies are obliged to determine the suitability of proposed suppliers providing products and services, particularly those providing software. Regulatory authorities neither prohibit or endorse specific systems despite what some supplier marketing might suggest.

> *... software is a critical component of a computerized system. The user of such software should take all reasonable steps to ensure that it has been produced in accordance with a system of Quality Assurance.*[6]

This requirement is usually met by means of the audit conducted by the pharmaceutical or healthcare company. The audit examines the quality assurance attributes of the supplier's process, the firm's general capability maturity, and the suitability of its equipment or service suggested for use on the project. Suppliers in this context may be understood to include equipment vendors, service suppliers, or the pharmaceutical or healthcare company's in-house software development department. Regulators hold pharmaceutical and healthcare companies accountable for the use of suppliers whose capability assessment indicates their inability to deliver validatable, compliant software.[7]

In order to fulfill regulatory requirements for computer validation, pharmaceutical and healthcare companies are expected to mitigate deficiencies they identify in the quality of their suppliers' software. The U.K. MHRA (formerly MCA) accepts that "Where there is little or no supplier cooperation over validation evidence, it can be difficult to assess the QMS in place at the supplier, let alone have access to source-code and testing records."[8] In such situations, Supplier Audits are impractical and the industry has often had to rely on functional testing and change control alone. This is an enormous handicap as functional testing of this or any kind can never furnish an equivalent level of assurance over innate structural integrity that may be derived from an in-depth examination of the supplier's development process. However, the regulatory authorities expect that quality issues form an integral part of the supplier selection process and that wherever possible quality-minded suppliers should be selected. The implication here is that neglect of quality issues during supplier selection is irresponsible and unacceptable. Technical and quality aspects must be accorded equal weight when determining the fitness for purpose of a product chosen for use in the pharmaceutical and healthcare industry.

While the U.K. MHRA concedes that audits of firms like Microsoft, which produce ubiquitous "standard software," have not been routine, it does not condone complacency on this difficult question. Indeed, it continues to press the industry to address this issue. Microsoft should never be regarded as being above the scrutiny and accountability to which all other software firms are properly subjected. Companies like Microsoft are just as susceptible to poor software development as other firms; a recent U.S. court case challenged Microsoft for allegedly losing certain core design documents for one of its products.[9] The PDA Supplier Auditing and Vendor Qualification Initiative, supported by the FDA, could be a useful mechanism to pursue in this respect, rather than individual pharmaceutical and healthcare manufacturing companies attempting to tackle the issue. Even if Microsoft acceded to a pharmaceutical and healthcare industry Supplier Audit, there is no assurance that it would be willing to address any issues identified to improve its software processes. From its dominant monopoly position it is virtually able to dictate to the industry what is, and what is not, an acceptable user requirement.

There have been many debates over whether or not regulators have the right to scrutinize Supplier Audit reports. It is the policy of the FDA and other regulatory authorities not to request to inspect such audit reports without due cause (clause 20(c)[10] and clause 22,[11] respectively). All that the regulatory inspectors require is that the pharmaceutical or healthcare company can demonstrate that audits are being performed in accordance with a documented standard procedure. Such demonstration must include the preparation of written reports indicating that required corrective actions affecting suppliers and vendors as a result of the audits have been followed up to a satisfactory conclusion. This is usually achieved through presenting audit schedules, audit procedures, Validation Plans outlining specific audit requests, and Validation Reports noting progress on audit findings. The regulators do not wish to discourage honest and relevant audit reports being written because these might be scrutinized by them. The regulators realize that such reports are sensitive and confidential, and that it is in their interest that pharmaceutical and healthcare companies readily identify supplier and vendor strengths and weaknesses. They want to encourage pharmaceutical and healthcare companies to take effective corrective actions where necessary.

Invitation to Tender

Invitations to Tender are typically distributed with an accompanying information pack. This should provide contact details for purchasing departments, technical enquiries, etc., and define expected response times. A key document to include within the information pack is the URS. Without a clear understanding of requirements, assumptions, and constraints, the supplier will find it difficult to confidently respond with a clear proposal. In recent years the regulatory authorities have exerted increasing pressure to improve the content of URS because of the adverse impact that a poorly drafted URS can have on the ensuing validation. Such an emphasis on investing in a well-prepared URS document has been welcomed by supplier organizations. Other items to be included in the information pack might be copies of relevant company standards (e.g., compliance checklists and a summary of their control and operability philosophy). Suppliers are normally expected to have their own copies of industry standards and guidelines.

Supplier Proposal

Depending on the nature of the Invitation to Tender, the supplier's response may amount to little more than a covering letter with accompanying standard literature. This is often the case for COTS products that the supplier believes meets the user's requirements, or when a dedicated supplier proposal for a more bespoke or customized solution is to follow. Some suppliers may even draft an initial Functional Specification to demonstrate the suitability of the solution they have in mind. Functional Specifications are discussed in detail in Chapter 8. Other proposal documentation might include draft Project and Quality Plans.

Proposal Evaluation

The proposals that are received in response to the initial requests need to be evaluated against those requests to highlight any deviations. Preferred suppliers are then typically shortlisted.

Supplier evaluations should assess a number of factors including:

- The capability of the supplier organization within a quality-orientated culture
- Whether or not a quality management system exists, and is applied and maintained
- The technical competency of staff, as well as the awareness/understanding of pharmaceutical and healthcare industry regulations/practices
- Whether or not the supplier routinely supplies the pharmaceutical and healthcare industries, and is therefore familiar with regulatory expectations
- Whether the company is sufficiently financially stable to be able to support the system throughout its operational life

These factors should be assessed for each supplier being considered, so that the best balance of business fit and compliance can be achieved. An example supplier evaluation matrix is provided in Figure 7.4. Of course, identifying a clear winner in the selection process is seldom clear-cut, and the winner rarely meets all the selection criteria. Some pharmaceutical and healthcare companies weigh the importance of various factors and for the outcome compare the sum totals for each supplier. A summary of the supplier selection process applied to larger systems should be retained for possible subsequent inspection.[12]

Suppliers have an inherent legal responsibility to all their users that their products and services are fit for purpose (e.g., in the U.K. this responsibility is enshrined in the Sale of Goods Act). Pharmaceutical and healthcare companies will sometimes conduct an audit of their suppliers to assess their quality systems and management. The audit may be part of a supplier selection exercise. Auditors may be available among the pharmaceutical or healthcare company's own staff or engaged and commissioned from specialist audit firms providing these services. Any shortfalls in the supplier's capability uncovered in the audit will have to be mitigated, preferably by the supplier correcting these through process improvement. If not, the pharmaceutical or healthcare companies must initiate action themselves either directly or through third-party support.

Suppliers can benefit enormously if they adopt a positive, enlightened attitude to audits, especially if the auditors are able to direct their attention to process improvements. This is because such improvements will aid the supplier to get the products right first time, thus saving money in the long term: such has been the experience of the overwhelming majority of firms who have brought their quality system to a level compliant with the GAMP Guide or ISO 9001:2000. It is

	Supplier A	Supplier B	Supplier C
Quality Organization	✓	✓	✓
Quality Systems Exist	✓	☒	✓
Quality System Applied	✓	☒	☒
Quality System Maintenance	✓	☒	✓
Technical Competence	✓	✓	☒
Pharmaceutical Experience	✓	☒	☒
Support Infrastructure	✓	☒	✓
Commercially Robust	✓	✓	☒

FIGURE 7.4 Example of a Supplier Evaluation Matrix.[13]

most important that pharmaceutical and healthcare companies motivate their suppliers in this way, as far as possible, in a spirit of a long-term business partnership.

The use of qualified auditors is preferable. An example of such qualification would be the formal written examination for ISO 9001:2000 (TickIt) and subsequent registration under the auspices of the International Register of Certificated Auditors.

Contract of Supply

Contracts of supply should exist for all computer systems acquired from external suppliers. The contract should include terms and conditions to define individual responsibilities, the assignment of responsibilities to third parties, confidentiality, intellectual property rights (IPR), and terms of payment. It is very important that infringements, liabilities, and insurance are also covered, and it is determined how these are affected by circumstances outside the control of the customer and the supplier. Contract details should be reviewed during Supplier Audits as appropriate. Contract documentation should be retained as specifically required by some regulatory requirements in support of validation.

Contracts made in relation to customized or bespoke computer systems should make reference to the Validation Plan, any Supplier Quality Plans, and, where appropriate, the System Specification to define the scope of work, goods, and services.

Copies of purchase orders and corresponding supplier dispatch notes should be retained for COTS products. These should specify relevant model/version numbers so that, if need be, supporting information can be traced later with the supplier. Version numbers of technical documents and user manuals provided with the COTS product should also be noted on the purchase order.

A purchasing process that addresses compliance issues is presented in Figure 7.5. This covers supplier selection, award of contract, delivery, and validation. The process is initiated with a determination as to whether or not an audit is required. Once the need for an audit has been established, the supplier concerned should be contacted and an audit request made, explaining the context of the audit.

Recent Inspection Findings

- There is no written agreement or contract that establishes quality requirements that must be met by suppliers and contractors; that defines the type and extent of control to be exercised over the product or service; and there is no record maintained of listing acceptable suppliers and contractors. [FDA Warning Letter, 1999]
- No documentation is available to show that either internal quality audits or an evaluation of the contract software developer had ever been performed. [FDA Warning Letter, 2001]

SUPPLIER AUDITS

Applicability

Supplier Audits are not appropriate for all computer systems. Chapter 5 has already reviewed Supplier Audit expectations for different categories of hardware and software. Supplier Audits should be undertaken for custom (bespoke) applications and systems configuration. In addition, Supplier Audits should be considered for COTS configurable packages when they are used for GxP critical applications, especially so if the COTS product is highly complex. Figure 7.6 summarizes those situations when Supplier Audits are appropriate. It assumes that the specific version of the COTS product being considered is in successful use to a sufficiently wide extent. *In successful use* here implies that the COTS product is stable and that it is highly unlikely that a significant number of important defects remain to emerge. Of course, by its very nature this

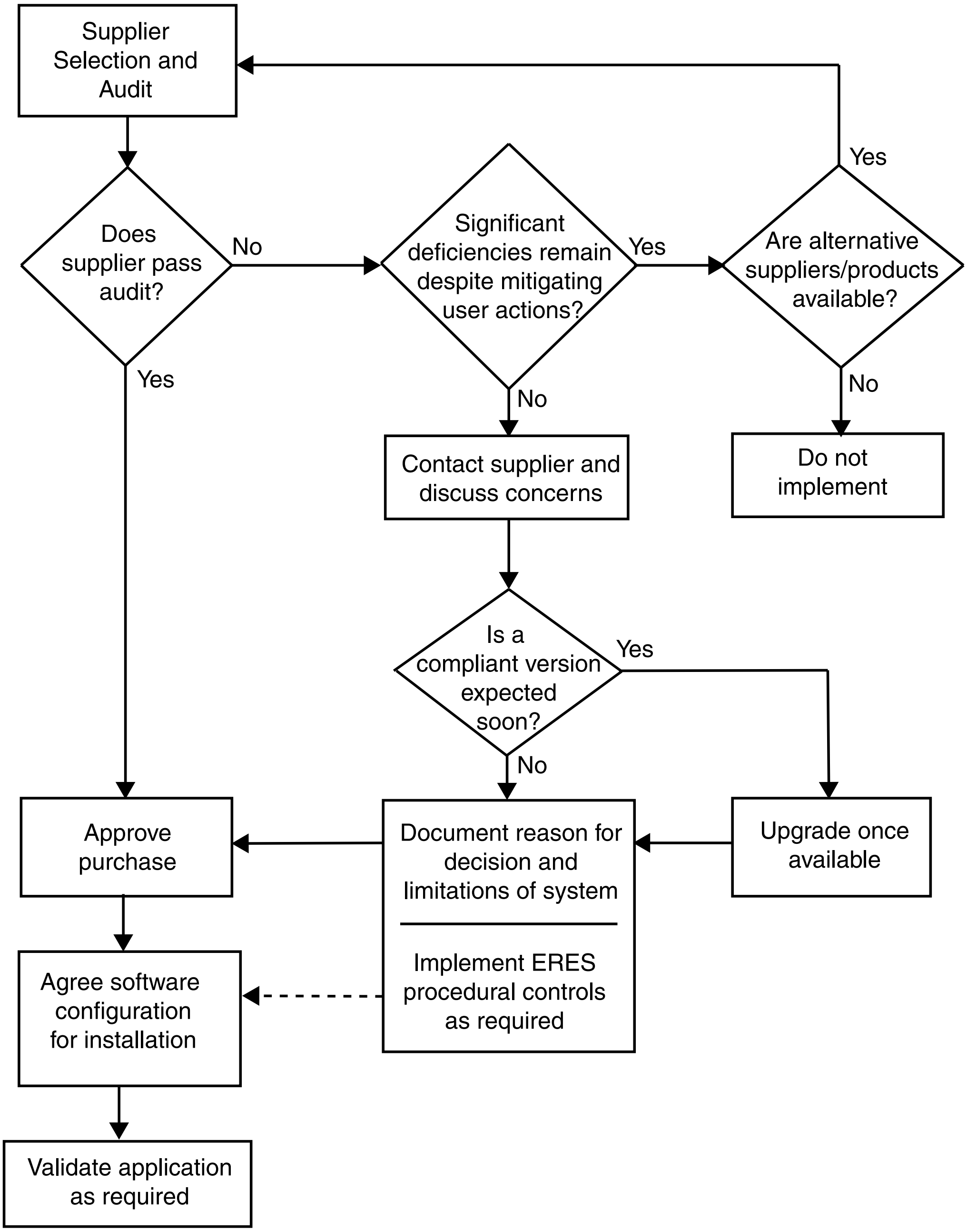

Supplier Selection and Audit
Does supplier pass audit?
No
Yes
Significant deficiencies remain despite mitigating user actions?
Yes
No
Are alternative suppliers/products available?
Yes
No
Contact supplier and discuss concerns
Do not implement
Is a compliant version expected soon?
Yes
No
Approve purchase
Document reason for decision and limitations of system
Implement ERES procedural controls as required
Upgrade once available
Agree software configuration for installation
Validate application as required

FIGURE 7.5 Purchasing Process Addressing Compliance Issues.

can never be known with anything approaching certainty, and there is no fixed minimum number of successful users. However, we would expect no less than 100 active applications of exactly the same version (no special adaptations!) would be required to be in successful use in this context. The so-called *early adopters* of COTS products (i.e., those acquiring products at a time when less than 100 active implementations of the product release exist, and/or when the product is still under anywhere from 6 to 12 months old) should treat them as custom (bespoke) computer systems and audit them accordingly.

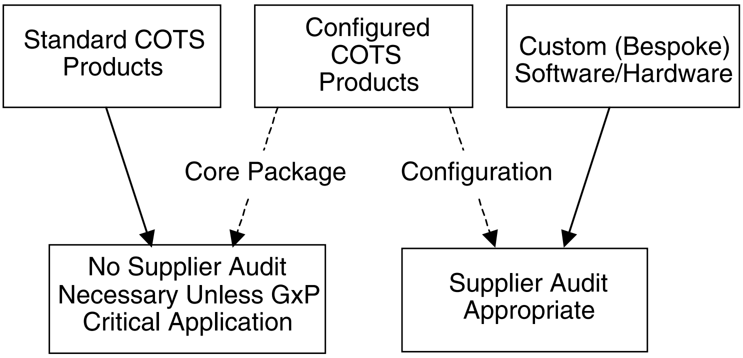

Standard COTS Products
Configured COTS Products
Custom (Bespoke) Software/Hardware
Core Package
Configuration
No Supplier Audit Necessary Unless GxP Critical Application
Supplier Audit Appropriate

FIGURE 7.6 Supplier Audit Determination.

Audit Preparation

Supplier Preparations

A supplier may be contacted informally to discuss the most appropriate audit route, prior to any formal approach being made to conduct an audit. Any proposal or bid made by a supplier should contain the contact details for the quality managers and personnel so that the auditor can easily liaise with the appropriate representative in the supplier's organization to discuss the next steps. When the auditor contacts the supplier to organize an audit, the supplier should also get the auditor's details (full name, postal address, telephone, fax, e-mail, and even video conference details where appropriate) for future communications.

It is important that, even at this early stage, formal confidentiality agreements should be signed for nondisclosure of proprietary information to unauthorized third parties. Suppliers should consider preparing a standard proforma that can be completed and faxed/mailed to the auditor. The agreement must state whom the signatories represent. Where a pharmaceutical or healthcare company engages the services of a third party auditor or company, the supplier must ensure that the agreement covers both of them and the auditor or his/her company. It should be made clear that written permission is required from the supplier to authorize the disclosure of the audit findings to the commissioning pharmaceutical or healthcare company, whether or not they are favorable to the supplier. Nonetheless, from a legal standpoint it would almost certainly be viewed in law that the supplier had conceded this right by allowing the audit to occur in the first instance.

Further guidance for suppliers is summarized in the checklist given in Appendix 6D; it can be used to monitor progress during audits. It is suggested that suppliers consider taking a photocopy of the checklist and annotate it with user details (proposal number and/or contract number), along with comments as appropriate to the audit. It should then be retained with their user/customer files. By planning audit schedules, conducting preparatory work, and ensuring that the right people are available to present and explain the supplier's ways of working during the audit, a lot of the stress associated with audits can be relieved. A mood of resentment felt by suppliers after an audit benefits no one.

ISO 9000 Accredited Suppliers

Many suppliers have quality systems registered as compliant with one of the ISO 9000 quality management standards.[14] The earlier versions of these standards, based on compliance with procedures, consisted of the following:

- *ISO 9000-1:* Guidance on selection of ISO 9000 series standards
- *ISO 9000-2:* Guidance on generic application of ISO 9000 series standards

- *ISO 9001:*1994: Addresses quality practice for product design, development, production (including inspection and testing), installation, and services
- *ISO 9000-3:* Guidance on applying ISO 9001:1994 to software (TickIT)
- *ISO 9002:* Addresses quality practice for production (including inspection and testing), installation, and services
- *ISO 9003:* Addresses quality practice for final inspection and testing

The ISO 9001:2000 standard has now replaced ISO 9001:1994. Emphasis is now being placed on achieving a documented state of control, with customer satisfaction, management endorsement, and continuous improvement. These were no more than implications in the earlier standards. The TickIT Guide version 5 accompanies the new standard and may be regarded as a much more detailed guideline for software companies in best practices. In this context, it is a worthy companion to the GAMP Guide. Auditors would do well by commending both to suppliers.

It is important to acknowledge that software can be developed under a quality system registered as compliant with ISO 9001:2000 without necessarily being fully compliant with the guidance given either in ISO 9000-3 or in the TickIT Guide. It must be remembered that a supplier's certification to ISO 9000, certainly the earlier standard, does not necessarily imply a capability to develop GxP compliant software. The auditor should carefully read the ISO 9000 certificate for its qualifying statement and validity duration, for it may not even apply to software development. A system or equipment supplier may subcontract software production, and thus not be accredited for this activity. If possible, the auditor should identify such circumstances before the visit and determine whether the supplier has the necessary safeguards to ensure the quality of subcontracted work. An audit of the subcontractor might then follow as appropriate.

Regardless of whether suppliers are accredited to ISO 9000 or not, they should be aware of how their quality system measures against requirements of this standard. Suppliers should consider preparing a short report mapping their current working practices to the clauses of ISO 9000. This report could then be supplied quickly and easily, if requested, to existing and prospective users. In addition, suppliers should collect reference site material (possibly as sales literature) and user contacts that can be supplied to potential new users. Suppliers may wish to consider collecting testimonials at the end of projects rather than trying to locate users subsequently who might have moved on by that time.

Postal Audits

Postal Audits are usually used as part of the supplier selection shortlisting process. They may recommend an audit on a supplier's premises where extra detail is required, but typically such Supplier Audits are reserved for the preferred supplier in order to confirm their acceptability from a compliance perspective.

Howard Garston-Smith, formerly of Pfizer, has published a book on software quality assurance.[15] It provides a postal audit checklist reproduced in Appendix 7C. If the supplier has already prepared an internal ISO 9000 mapping or an internal audit report on how it aligns to industry standards such as the GAMP Guide, this can be offered as an alternative to the auditor's postal checklist. A reduced postal checklist may be agreed upon, at the very least. Wherever possible, photocopies of actual example documents and test records should be inspected for documentary evidence of validation. Remember that the pharmaceutical and healthcare companies are themselves being inspected for documentary evidence of validation.

Auditors should agree to response times with suppliers when using the postal checklist. Busy contract periods, illness, holidays, and staff turnover may all delay the responses. Before agreeing to a response time, suppliers should ask to see a copy of the audit checklist so that the amount of work needed to reply can be gauged.

Visiting Supplier Premises

Audits on supplier premises are used to assess a supplier's quality management system at first hand with detailed examination of procedures and documentation relating to a product or service. As discussed earlier, such audits are primarily conducted for suppliers of bespoke (custom) software and systems.

Visits to supplier premises can be expensive not only to the supplier concerned but also to the pharmaceutical or healthcare company. It is important to both parties that they are conducted quickly and efficiently. Reimbursement to suppliers to cover the costs associated with audits are not normally offered or requested. Some suppliers have tried to levy fees, but this has created ill feeling, and supplier firms adopting this approach are generally felt to have blotted their reputation in the marketplace — after all, a quality culture should have already been established in the supplier firm. The audit should be seen by a supplier as an opportunity not just to defend the supplier's software development capability maturity but also to sell quality as a distinguishing feature to the product or service on offer. Some suppliers have offered pharmaceutical and healthcare companies the chance to audit collectively in managed groups, perhaps through a user group structure. Suppliers offering this facility are limiting the number of audits to two or three per year, and have reduced their annual audit costs by up to 80%.[5]

Assembling Audit Teams

Audits are usually conducted by an audit team and led by a qualified, accredited auditor. Accredited auditors should have completed a certified development program, be accredited under an appropriate standard (ISO 9000:2000 TickIT in this context), and should have conducted a number of qualifying audits. Pharmaceutical and healthcare companies that do not have their own accredited auditors can engage an independent auditor, as we have seen. Names and addresses of accredited auditors can be found in national registers of certified auditors. The International Register of Certificated Auditors is one such register, associated with the Institute of Quality Assurance (IQA).

The audit team's size should be kept to a minimum but should adequately represent the following areas of expertise:[4]

- Auditing practices
- Computer system engineering
- Computer system quality methods
- Regulatory compliance
- Pharmaceutical and healthcare industry validation practices

Team members should have a preliminary understanding of auditing before they begin an audit. The ISO 10011 auditing standard[2] provides useful material on audit practice, the qualifications of auditors, and the management of audit programs.

Conducting Audits

Audits normally take 2–3 days to conduct. One-day audits are possible when the auditor and the firm being audited are well prepared or there is a reduced audit scope. Care must be taken with shorter audits to make sure that the assessment is not unacceptably superficial and lacking the required scope or depth.

A typical audit schedule might consist of:

- A general introduction by the supplier giving details of the audit plan and an overview of the supplier's business operations and quality department organization
- A presentation by the supplier of the firm's quality management system, perhaps reviewing the supplier's internal ISO 9000 mapping report

- A presentation by the supplier of the quality process adopted for a particular product or service under scrutiny, perhaps including a review of other user quality expectations with emphasis on pharmaceutical and healthcare industry users
- An opportunity for the auditor to view actual quality documents and records (example checklist for audits is included as Appendix 6E)
- A summary by the auditor of preliminary findings with an opportunity for the supplier to discuss and clarify

Alternatively, very experienced auditors often have a well-practiced structure or framework by which they conduct audits and draft their reports. This can be of great value where a pharmaceutical or healthcare company decides to audit several potential suppliers for a critical system. In such cases, it usually needs the reports with an identical layout, scope, and depth. This makes comparisons on a level basis easier. Advising the supplier of this in advance is a great help all around.

It is useful if the supplier has nominated a representative to host the audit, ideally someone who has been trained in auditing and therefore can understand and anticipate the auditor's perspective. Training courses are available, for instance, along the lines of industry auditing standards like ISO 10011. Good interpersonal and language skills are often beneficial, too, especially where the audit is not conducted in the participants' mother tongue.

The supplier may require a briefing in the proposed audit process objectives and scope. The audit is then conducted and a report is prepared by the pharmaceutical or healthcare company concerned. The supplier should be given a chance to review and comment on the report so that any factual errors may be corrected, vendor comments added, or other necessary clarifications can be made at an early stage. Any corrective actions undertaken by the supplier should be followed up, and therefore the audit process may loop back by the arrangement of a follow-up visit.

The timing of the audit should be considered here, as some potential members of the team may not be available due to holidays, and their deputies might need to be in place to deal with any unforeseen absences. The logistical challenge of undertaking a supplier audit should not be underestimated. Items to consider are as follows:

- Who is the auditor (by name and function)?
- Will the auditor be assisted by anyone (name and function) or observed by anyone (e.g., trainee or other interested party)?
- Who is the host firm representative (name and function)?
- Who is assisting the host representative (name and function)?
- Where is the audit being conducted; have maps and times been sent to participants?
- Has hotel accommodation been organized where needed?
- Have presentation slides and materials been prepared and are copies available for the auditor?
- Will relevant information be brought to the audit by host firm representatives?
- Has a tour of the supplier premises been organized?
- How long is the audit estimated to take? (Do not let the audited organization dictate the duration of the audit; take the time needed to do a proper job.)

It is useful to assign a base camp room for the duration of the audit — a place reserved for the audit team whether or not they are permanently located there. It provides a focus for the audit team and a place for it to leave members' personal baggage and collect files of information pending review, as well as leave information behind that has served its purpose.

Subcontractors

The use of subcontractors needs to be clearly understood well before the audit visit, together with the oversight planned by the prime supplier. If the prime supplier does not have adequate control over subcontractors, then Supplier Audits by the pharmaceutical or healthcare company may be required. Key activities and documentation under the management and control of subcontractors should be subjected to review by the prime supplier. The topics listed in the general Supplier Audit expectations in Appendix 7D are relevant to subcontractors. Any deficiencies found in a subcontractor's software development must be mitigated by supplementary work to bring it up to an acceptable standard.

Audit Report and Follow-Up

The structure of a Supplier Audit Report is suggested in Appendix 7E and covers the following aspects:

- Introduction and scope
- Summary of audit process
- Review of supplier quality assurance system
- Audit Results with list of corrective actions
- Audit outcome (conclusion)

Suppliers should be advised at the outset of an audit that they will be accorded the opportunity to review and correct a draft of the Audit Report before it is issued, as we have mentioned above. There should also be some agreement as to the timetable expected for the issue of a draft report. It is not proper for a supplier to seek to approve the audit report since that could amount to an attempt to gag the auditor. By permitting the audit in the first place, the supplier has effectively given the auditor permission to exercise his/her professional impartiality and this freedom may not subsequently be withdrawn. The contents of the report are the auditor's opinions and should not be influenced except where they are based on factual misconceptions that must be corrected by the supplier. The auditor has a duty of care to the supplier to permit this. The supplier should retain a copy of the draft Audit Report with a note of their review comments. A final copy of the approved and issued report must also be furnished to the supplier by the auditor as the basis for future business relationships.

Actions and recommendations should be differentiated. Actions must be completed or must progress to an acceptable state before the computer system can be authorized for use. Open actions should be listed in the Project Compliance Issue Log (see Chapter 6). Recommendations are just that; they do not need to be completed before the computer system is authorized for use. Follow-up audits should track actions and recommendations as appropriate in the context of that audit. The management of actions and recommendations are summarized in Figure 7.7.

The Audit Report should indicate the significance, importance, and priority to be adopted for the various corrective actions. After being informed of the outcome of audit, it is the responsibility of the supplier to agree to follow-up actions and timescales. If the auditor has been competent enough to engender a positive spirit into the auditor/supplier relationship, the supplier should be defensive at this point but cognizant of weaknesses and motivated to effect improvements. As intimated earlier, here is a marvelous opportunity for audit findings to act a springboard for an ongoing program of continuous improvement. The ability to demonstrate such an attitude to improvement will always reassure and impress both the user and the user's organization. Consider using the expertise of the auditor to debate options for quality improvement; it is often a free resource within the pharmaceutical or healthcare company, though consultancy fees might be incurred if the auditor is external. Finally, agree on how to communicate the completion of agreed follow-up actions with the auditor/user. Not all recommendations for action have to be implemented by the supplier, but try to accommodate the auditor's findings.

It is very important that after the audit the auditor continues to be willing to act as a mediator between pharmaceutical or healthcare company and supplier to promote trust, to assist the supplier in any way, and to promote the business partnership. The pharmaceutical or healthcare company

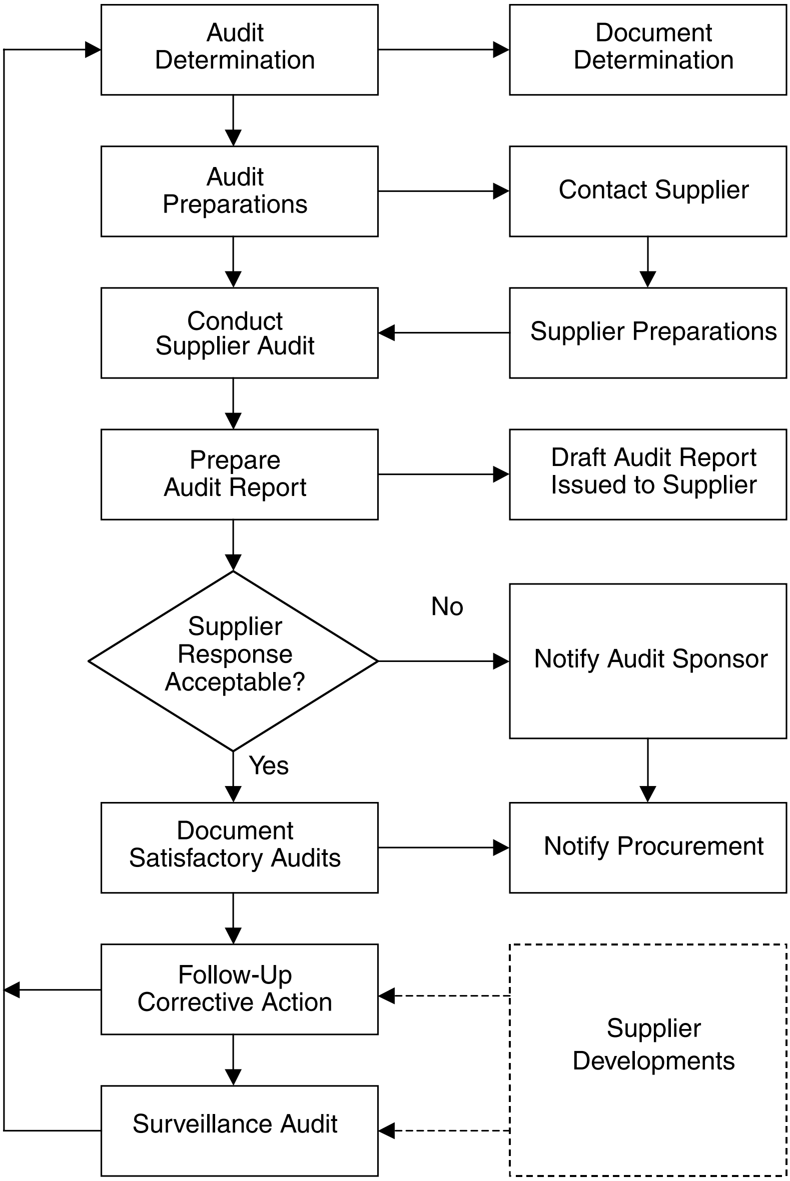


FIGURE 7.7 Audit Process.

may organize a follow-up audit to check whether correction actions have been taken. The number and frequency of recommended follow-up audits should be recorded in the audit report.

Preferred Suppliers

Supplier Audits may be conducted in advance of validating an individual computer system or as part of an overall strategy to select preferred suppliers. Some pharmaceutical and healthcare companies rate the capability maturity level of suppliers within a scoring system. Suppliers might be ranked with a numbering system or with a combination of keywords such as "excellent," "satisfactory," "noncompliant," and "ISO certified." Bernard Anderson of GlaxoSmithKline has suggested the following rankings:[16]

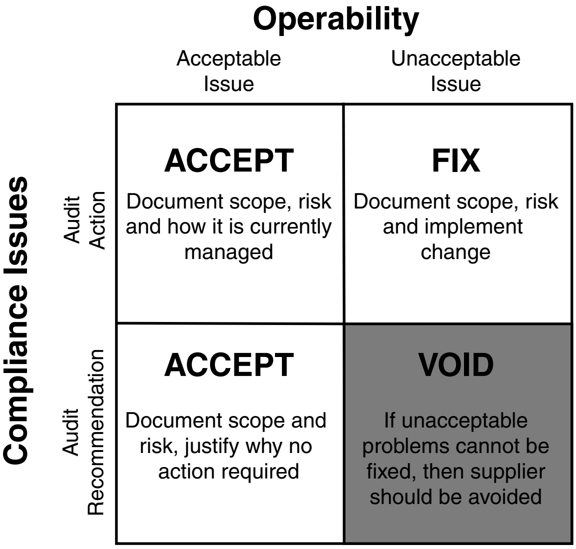
- The supplier can be used for any type of software development work.
- The supplier can be used for specific types of software development work.
- The supplier can be used for specific types of software development work, subject to *minor* corrective actions being performed.
- The supplier can be used for specific types of software development work, subject to *major* corrective actions being performed.
- A documented quality system must be imposed on the supplier in the event a contract is awarded.
- The supplier cannot be used for any systems requiring validation.

Whatever system is used, the ratings must be defined so that users and regulators alike have an unambiguous understanding of their meaning and the suitability of the supplier. Project teams can use the rating system to select suppliers without unnecessarily duplicating audits. Companies using this approach must carefully consider how long a rating remains valid until refreshment of the information is required since a supplier's capability can change — and not always for the better.

User Groups

It may be advantageous to join or establish a User Group for a particular computer system to exchange user experiences, disseminate support information, organize collective training, and influence the direction of further development. Later when the systems require replacement, the group might develop a migration strategy. Many suppliers are keen to facilitate User Groups as a means of:

- Building a shared vision for product developments
- Gathering direct feedback on system operability and any problems experienced
- Helping to prioritize the correction of known defects
- Standardizing audit requirements
- Encouraging users to conduct shared Supplier Audits



Operability

Compliance Issues	Acceptable Issue	Unacceptable Issue
Audit Action	**ACCEPT** Document scope, risk and how it is currently managed	**FIX** Document scope, risk and implement change
Audit Recommendation	**ACCEPT** Document scope and risk, justify why no action required	**VOID** If unacceptable problems cannot be fixed, then supplier should be avoided

FIGURE 7.8 Audit Actions and Recommendations.

User Groups may be confined to staff within a pharmaceutical or healthcare company's own organization, perhaps for reasons of confidentiality or competitiveness. Generally speaking, however, there are greater benefits to be had from the participation of individuals from multiple user companies.

Recent Inspection Findings

- No written procedure for vendor audits. [FDA 483, 2002]
- No approved written procedures for vendor qualification. [FDA 483, 2002]
- No record of vendor audits performed for suppliers. [FDA 483, 2002]
- Your firm failed to ensure that the supplier of the XXXX documented all of the required test results to indicate the supplier's quality acceptance of the XXXX delivered to your firm. [FDA Warning Letter, 2002]

REFERENCES

1. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (www.ispe.org).
2. ISO (2000), *ISO 10011-1 (1990): Guidelines for Auditing Quality Systems — Part 1: Auditing*, International Organization for Standardization, Geneva.
3. FDA (2002), *General Principles of Software Validation; Final Guidance for Industry and FDA Staff*, U.S. Food and Drug Administration, Rockville, MD.
4. Grigonis, G.J. and Wyrick, M.L. (1994), *Computer System Validation: Auditing Computer Systems for Quality*, Report on behalf of PhRMA Computer Systems Validation Committee, Pharmaceutical Technology Europe, 18 (9): 32–39.
5. Wingate, G.A.S. (1997), *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice*, Interpharm Press, Buffalo Grove, IL.
6. European Union Guide to Directive 91/356/EEC (1991), *Annex 11 — Computerised Systems.*
7. FDA (1985), *Vendor Responsibility*, Compliance Policy Guides, Computerized Drug Processing 7132a, Guide 12, U.S. Food and Drug Administration, Rockville, MD.
8. Trill, A.J., Computerised Systems and GMP — A UK Perspective, Part 3: Best Practice and Topical Issues, *Pharmaceutical Technology International*, March 1993, pp. 17–30.
9. Fragments of Windows Go Missing, *New Scientist*, September 12, 1998, p. 16.
10. U.S. Code of Federal Regulations Title 21: Part 820, Good Manufacturing Practice for Medical Devices.
11. FDA (1986), *FDA Access to Results of Quality Assurance Program Audits and Inspections,* Compliance Policy Guide 7151.02, U.S. Food and Drug Administration**,** Rockville, MD.
12. Pharmaceutical Inspection Co-operation Scheme (2003), *Good Practices for Computerised Systems in Regulated GxP Environments*, Pharmaceutical Inspection Convention, PI 011-1, Geneva, August.
13. Reid, C. (2001), *Effective Validation Strategies*, Third Annual Conference on Computer Systems Validation for cGMP Pharmaceuticals, London, March 28 and 29.
14. ISO (2000), *ISO 9000: Quality Systems — Model for Assurance in Design, Development, Production, Installation and Servicing*, and *ISO 9001-3: Quality Management and Quality Assurance Standards — Part 3: Guidelines for the Application of ISO 9001 to the Development, Supply and Maintenance of Software*, International Organization for Standardization, Geneva.
15. Garston-Smith, H. (1997), *Software Quality Assurance — A Guide for Developers and Auditors*, ISBN 1-57491-049-3, Interpharm Press, Buffalo Grove, IL.
16. Anderson, B. (1996), *Vendor Audits*, Computer System Validation: A Practical Approach, Management Forum Seminar, London, March 26 and 27.
17. Supplier Forum (1999), *Guidance Notes on Supplier Audits Conducted by Customers*, www.ispe.org/gamp.

APPENDIX 7A
EXAMPLE CONTENTS FOR USER REQUIREMENTS SPECIFICATION

Introduction

- Author/organization
- Authority
- Purpose
- Relationship with other documents
- Contractual status of document

System Overview

- User perspective of overall function
- Data flow
- Control dependencies
- Operator interface
- Operating environment
- (Reference to separate System Overview document if appropriate)
- Assumptions

Operational Requirements

- Major functions
- Secondary functions (error handling, monitoring, data integrity checks)
- Start-up and shutdown
- Normal and abnormal operations
- Error and failure reporting
- Recovery and fallback
- Alternative modes of operation
- Electronic records and signature requirements
- Hard copies — print requirements for reports, events and fault logs, preference for printer types and stationery
- Operator control consoles — layout and functionality
- Operator input devices — keyboards and touch-sensitive displays
- Displayed information — layout of mimics, menus, reports, tables, display hierarchies, and colors
- Ergonomic factors — operator comfort and usability of controls and keys
- Critical system timings — throughput, response, and update of data
- Volume of transactions
- Number of simultaneous users
- System security — use of passwords and keys
- Data security — backup and recovery
- Safety of personnel

Design Constraints

- Hardware
  - Requirements and environment

  - Standards
  - Interface
  - Tolerance, margins, contingency
- Software
  - Standards and programming languages
  - Interfaces
  - Software packages
  - Databases
  - Operating systems
  - Tolerance, margins, and contingency
- User interface
  - Interface characteristics
  - Environmental constraints
- Operational constraints

System Interfaces

- Content of user displays
- Levels of user access for different user groups
- Digital/analogue inputs/outputs to external equipment
- Serial inputs/outputs to external equipment
- Parallel communications to external equipment
- Network communications to external equipment

System Environment

- Services — electrical power and heat removal
- Environmental conditions — temperature, humidity, noise, and contamination; intrinsic safety in hostile environments

System Nonfunctional Requirements

- System availability
- Recovery from failure
- Preferred diagnostic methods
- Required level of maintenance support and support period
- Training
- Required documentation
- System hardware maintenance
- Possible enhancements that may be required in the future

Documentation

- Operator instruction
- Maintenance procedures
- User manuals
- Training materials
- Validation documents/package
- Supplier project quality plan

Development Issues

- Host development system requirements
- Design methodologies, CASE tools
- Programming languages
- Subsystem testing
- Integration testing
- Configuration control
- Installation considerations
- Support services
- Maintenance requirements
- Expansion capability
- Expected change

Installation Considerations

- Conversion/migration instruction from existing system
- Hardware issues including upgrade and maintenance
- Software issues including upgrade and maintenance
- Training

Testing Requirements

- Levels of testing to be carried out (unit, module, system, integration, user acceptance, qualification)
- Amount of user vs. supplier involvement
- Use of simulation and prototyping

Appendices

- Glossary
- Others

APPENDIX 7B SUPPLIER CHECKLIST FOR RECEIVING CUSTOMER AUDITS[17]

Customer Details

- Name
- Proposal and/or Contract Reference

Preparation Work

- Prepare Internal ISO 9000 Mapping Report.
- Prepare Internal GAMP Audit Report.
- Prepare/Agree Confidentiality Agreement.
- Train supplier personnel responsible for receiving user audits in ISO 10011 audit process.

Invitation to Audit

- Send out and sign Confidentiality Agreement.
- Agree on scope and method of audit.
- Get auditor contact details for communications.
- Agree on response time for postal questionnaire.
- Agree on date, duration, and participants for full supplier audit.
- Agree on supplier review and approval of Audit Report.

Postal Audit

- Complete Postal Questionnaire, retaining own copy.
- Consider Issue of Internal ISO 9000 Mapping Report.
- Consider Issue of Internal GAMP Audit Report.

Full Supplier Audit

- Agree on audit schedule in advance of audit.
- Facilitate audit logistics.
- Consider Issue of Internal ISO 9000 Mapping Report.
- Consider Issue of Internal GAMP Audit Report.
- Prepare and Issue Audit Presentation Slides.
- Take own notes during audit.
- Have summary at closeout of audit.

Audit Follow-Up

- Review draft Audit Report, retain your review comments.
- Retain copy of draft and final Audit Report.
- Agree on follow-up actions and timescales.
- Agree on how to communicate completion of agreed followup actions.

APPENDIX 7C
EXAMPLE POSTAL SUPPLIER AUDIT QUESTIONNAIRE

1. Does your company have certification to any recognized internal standards?
2. Do you have a written Project Management System including the definition of responsibilities within projects?
3. Do you have a separate and independent quality assurance individual or group?
4. Do you have a written Quality Management System describing the controls on the software engineering process?
5. Is a written software development life-cycle methodology in use?
6. Is a written user requirements document a mandatory prerequisite for all projects?
7. Is a written functional specification or design document a mandatory prerequisite for software development?
8. Does a written programming standards guide or coding standards manual exist to define standards for programming?
9. Are structured test plans produced in advance of testing?
10. Are the testing personnel independent of the development personnel?
11. Are test results recorded and retained?
12. Is a written formal change control system for software and documents in operation?
13. Are formal written backup and contingency procedures in place?

APPENDIX 7D
EXAMPLE SUPPLIER AUDIT CHECKLIST[1]

- ***Audit Details***
  What is the name and address of the firm being audited?
  Who is the contact at the firm?
  What are the names and qualifications of the auditors?
  What is the date of the audit?
  Is this an initial or follow-up (surveillance) audit?

- ***Business***
  How long has the company existed?
  How is the company organized?
  How many pharmaceutical-related customers does the company have?
  Does the company have any customer citations for good work?
  Is the company's pharmaceutical-related business profitable?
  What is the long-term pharmaceutical-related business plan?
  Is the company in litigation?
  Does the company hold a recognized quality certification?

- ***Organization***
  Has a Quality Control System been established? Is it documented?
  Who is responsible for Quality Management?
  Is there a Quality Assurance management structure?
  Is there a project management structure?
  Are project work practices documented?
  How does project work conform to quality standards?
  Has accreditation/registration been achieved? (ISO 9000, specify other)
  Is the Quality System audited on a regular basis?

- ***Employees***
  How many permanent, contract, or temporary people does the company employ?
  How long, on average, do employees stay with the company?
  What is the company's training policy? Are there any records?

- ***Planning***
  Are project and quality plans produced for projects? Who approves them?
  Does planning include a contract review?
  Is there a defined project life cycle? What is it?
  How are project documents controlled?
  How is conformance to customer GxP requirements ensured?

- ***System Design and Development***
  Do design considerations cover reliability? Maintainability? Safety?
  Do design considerations cover standardization? Interchangeability?
  Are design reviews carried out? Are they minuted?
  Are customers invited to attend design review meetings?
  Are design changes proposed, approved, implemented, and controlled?

- ***System Build***
  Are there guidelines or standards for software programming and hardware assembly?
  How does the company ensure it conforms to current industry requirements?
  Are there records showing projects conforming to company practices? What third-party hardware or software is used? Are they supplied by reputable firms?
  How would changes to third-party products affect the customer's end product?

- ***Predelivery Testing***
  Are test specifications produced? Are expected results defined?
  Who performs the tests? How is testing organized?
  Are versions of hardware and software inspected?
  Is software "black box" (functional) testing conducted?
  Is software "white box" (structural) testing conducted?
  How rigorous is testing? Are abnormal situations tested?
  How are failed tests documented and corrected?
  Are test results recorded, signed, and dated? Are these records maintained?
  Who signs for overall acceptance of testing?

- ***Project Completion***
  What is the mechanism for deciding a project is complete?
  Is there a certificate of conformity? Is there a warranty?
  Are project documents handed over to the customer?
  Are project documents archived?
  Is there an access agreement for regulatory inspections (e.g., escrow)?

- ***Control Procedures and Activities***
  Is there configuration and version control within projects?
  Does the Quality System provide for the prompt detection of failures?
  Are all failures analyzed? Are correction actions promptly taken?
  Are regular internal audits carried out? Are auditing procedures documented?
  Are audits planned? Are corrective actions taken?
  Are audit records stored and available?
  Are responsibilities for document review assigned?
  Are responsibilities for change control assigned?
  Are obsolete documents withdrawn?
  Are changes notified to the customer?
  Are subcontractors audited? How are they managed?
  Are subcontract documentation standards audited?

- ***General and Housekeeping***
  Are customers solicited for feedback?
  How are customer responses folded into development plans?
  Are customers kept informed of development plans?
  List other customers provided with a similar service/product that is the subject of this audit.
  Are customers advised of problems found by other users?
  Who is responsible for ongoing customer support? Is there a support fee?
  What are the response mechanism and timings for customer problems?

APPENDIX 7E
EXAMPLE CONTENTS FOR SUPPLIER AUDIT REPORT

Introduction

- Author/organization and authority
- Purpose
- Relationship with other documents
- Contractual status of document

Scope

- Systems and software covered
- Dedicated audit or shared audit
- Details of single or multiple suppliers covered

Audit Process

- Reference procedure to be followed
- Reference Checklist used to guide audit (e.g., GAMP)
- Qualifications of Audit Team
- Identification of Lead Auditor
- Identification of individuals, with job titles, receiving audit

Quality Assurance System

- Describe supplier's quality assurance system
- Describe supplier's quality assurance organization
- Reference any independent certification of above, noting the supplier's scope of supply defined by the certification (e.g., ISO 9001, ISO 9001-3)

Audit Results

- Record of audit observations (good, bad, and ugly)

Corrective Actions

- Identify critical issues to project implementation
- Define acceptance criteria for closure
- Indicate follow-up requirements for project implementation

Audit Outcome

- Unconditional use of supplier
- Use supplier subject to corrective actions
- Supplier must improve quality — repeat audit
- Prohibit use of supplier

References

Appendices

- Glossary
- Audit team notes (possibly on checklist)
- Examples of documents reviewed (as appropriate)
- Others as appropriate

8 Design and Development

CONTENTS

Design and Development is the responsibility of the System Developer, although system users often take a leading role in the Design Review. The activities associated with this phase vary between projects, but generally follow the established pattern of Supplier Project/Quality Plans, Functional Specification, Software and Hardware Design, and Design Review. Facilitating requirement traceability is one of the most important activities. Throughout Design and Development, project managers should beware of extensions to the scope of the URS (sometime referred to as the "scope creep"). This is because each modification is likely to lead to a revision to the Functional Specification and subsequent documents, with associated incurred costs and project delays. If such extensions are still occurring by the time coding has commenced, and the supplier's development capability is not up to managing this safely, final software quality may be seriously degraded.

SUPPLIER PROJECT AND QUALITY PLANS

GMP regulatory authorities require work conducted by suppliers for pharmaceutical and healthcare companies to be formally agreed in a contract[1] covering:

- GMP requirements
- Responsibilities of parties involved
- Inspections of suppliers
- Customer agreement of subcontractors

The contract usually consists of a supplier's proposal, together with a customer's purchase order. Reference may be made to a URS and other documentation. It is advisable to incorporate, or make reference to, the supplier's Project and Quality Plans, which may be separate or combined. These plans share the same purpose as the Validation Plan and define the supplier's approach to its designated validation tasks. Without agreement on validation practices and management, it is unlikely that a project will be completed fully, on time, and within budget. It is not sufficient merely to state that the work must comply with the requirements of various GMP regulatory authorities. Resolving misunderstandings can be a complex and time-consuming task: "Bad contracts can seriously complicate your life!"[2]

The supplier's Project and Quality Plans specify project responsibilities, procedures, and deliverable documentation. The supplier's scope of work usually revolves around Design and Development, Coding, Configuration and Build, and Development Testing. The URS and User Qualification may also be specified in the plans as requested by a pharmaceutical or healthcare company. The supplier firm should be encouraged to include a statement of its development capability for the service or equipment they are providing. This statement could be based on a firm's ISO 9000 accreditation. Résumés of key staff should be available to support the capability statement.

Pharmaceutical and healthcare companies are expected to review the contractual arrangements before beginning a project. The combined information provided in the Supplier Audit and supplier's Project and Quality Plans gives a good indication of the competence of suppliers and their working relationship with the pharmaceutical or healthcare company. A simple checklist might include the following questions:[2]

- Does the customer order reference the final supplier proposal?
- Are the terms and conditions quoted in the customer order or supplier proposal acceptable?
- Are the milestones in the supplier's Project Plan accepted by all parties?
- Is the program of work in the supplier's Quality Plan accepted by all parties?
- Are the resource requirements available and used on the contract?
- Are you satisfied that the contract can be delivered in full, on time, and within budget?

If the answer to any of these questions is "no," then the actions and responsibilities to be taken to resolve the deficiencies should be recorded and verified as complete. The contract should not start until any deficiencies are understood and corrected. If a checklist is used, it should be annotated with answers to the questions and signed as complete by the contract reviewer (who may well be the quality manager) and contract manager (who may well be the project manager). In this simple manner, documentary evidence is produced to demonstrate that the contract for a specific project was reviewed and was satisfactory.

Suppliers should also be encouraged to conduct their own contract reviews. They must be satisfied that they can fulfill the contract, and that the pharmaceutical or healthcare company is aware of any necessary support and can provide it.

Contract reviews for computer systems will become increasingly important as pharmaceutical and healthcare facilities become more highly automated and dependent on software. Supplier Project and Quality Plans should, therefore, be adopted as a standard working practice for all projects. A combined Project and Quality Plan will normally be 10 to 15 pages long, but it does depend on the complexity of the project. Remember that the size of a document is not important; the key requirement is quality of content. An example of a content checklist for Project/Quality Plans is given in Appendix 8A. An organization's commitment to Project and Quality Plans is often a good indicator of its commitment to quality management and validation.

FUNCTIONAL SPECIFICATION

The Functional Specification is the supplier's response to the URS, specifying a proposed solution to the URS; therefore, the pharmaceutical or healthcare company must approve it. In some exceptional circumstances a URS may not be produced; in this case the Functional Specification stands in its own right.

CONTENT

Wherever possible, the Functional Specification should follow the same structure of the URS (see Appendix 8B), and can refer to the URS rather than duplicate information. A Requirements Traceability Matrix can be developed as described later in this chapter. Compliance, omissions, and nonconformances with respect to the URS should all be readily identifiable.

The Functional Specification should, as far as possible, avoid detailed design and concentrate on defining the operation and user interaction with the computer system. This is generally more difficult than it sounds. In some instances it is not even applicable because the URS specifically requests particular equipment or a particular design to be used. Similarly for small projects, it is often more convenient to combine the Functional Specification and Design documents into what is often referred to as a Functional Design Specification, System Definition, or System Description.

The GAMP Guide recommends the following content headings for a Functional Specification:

- System Architecture (scope/overview)
- Functionality (including information flows)
- Data (storage structures and data load)
- Interfaces (users and equipment)
- Nonfunctional Attributes (performance)
- Capacities (including expansion capability)

When preparing content for each section heading, it is useful to consider what regulatory authorities such as the FDA have indicated they look for from a system specification:[3]

- All inputs that the system will receive
- All functions that the system will perform
- All outputs that the system will produce
- The definition of internal, external, and user interfaces
- What constitutes an error and how errors should be managed
- All safety requirements, features, and functions that will be implemented

- All ranges, limits, defaults, and specific values that the system will accept
- All performance requirements that the system will meet (e.g., data throughput, reliability, timing, etc.)
- The intended operating environment for the system (e.g., hardware platform, operating system, etc., if this is a design constraint)

GxP functionality and configuration affecting drug product quality should be identified, and acceptable operating limits specified.[4] Messages for information should be distinguished from alarms generated by unacceptable situations.[5]

When considering electronic record and electronic signatures, attention must be given to particular regulatory preferences for functionality to be implemented entirely within the computerized system. Specific electronic record and electronic signature aspects to be covered by the specification include:

- Built-in checks where appropriate for correct data entry and data processing.
- Data should only be entered or amended by persons authorized to do so.
- The identity of person entering GxP data must be recorded by the computerized system or change control process.
- Manual entry of critical process and test data (excluding function and menu selection) must be verified by a second person or validated by electronic means.
- Audit trails should be established either by the computerized system or change control process for the creation and amendment of GxP data.
- Electronic batch sentencing (release) must only be done by authorized person (a *Qualified Person* within the European Union).

Many regulatory authorities also expect a high-level diagram to be included with supporting detail as appropriate. An example system overview diagram is given in Figure 8.1.

Dealing with COTS Products

In many circumstances, the supplied system will be based on a standard COTS product and include additional features that are superfluous in the intended context. These features cannot normally be disabled because they are integral to the COTS product. Such redundant features should be included in the Functional Specification, noted as superfluous and, if possible, rendered inaccessible to users within the implemented computer system. Standard features that support compliance, such as audit trails for electronic records, should be used even if not defined within the URS. In such circumstances it may be necessary to make additional design allowances for the inclusion of these features (e.g., for audit trail functionality, extra storage capacity may be required). Standard documentation for COTS products can be referenced by the Functional Specification, if available for inspection, rather than reproduced. Care must be taken to refer to the correct version of COTS documentation and to keep cross-references up to date following any system upgrades.

Anticipating Testing

The Functional Specification should be written in a manner that anticipates functional testing. The testing relationship for Functional Specifications is illustrated in Figure 8.2. Specifications should be specific, measurable, and achievable, so that testing can clearly demonstrate their acceptance. For instance, a sterilizing cycle time may be specified to be 90 sec when a variation of ± 5 sec is quite acceptable. If an acceptable variation of ± 5 sec is not recorded, then a test outcome of 89 sec will fail.

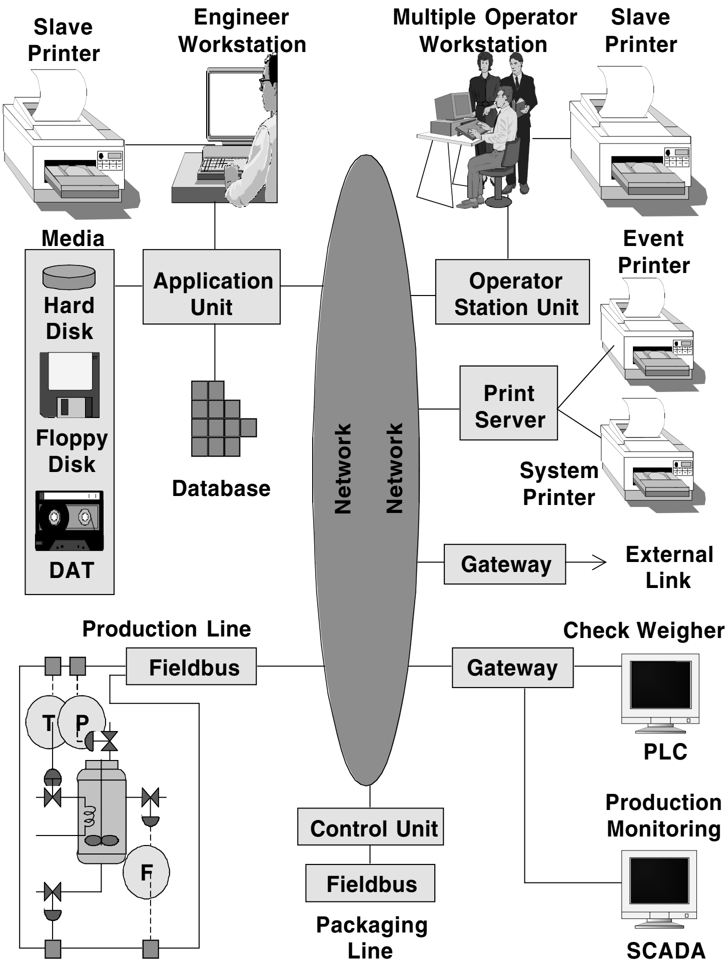


FIGURE 8.1 System Diagram.

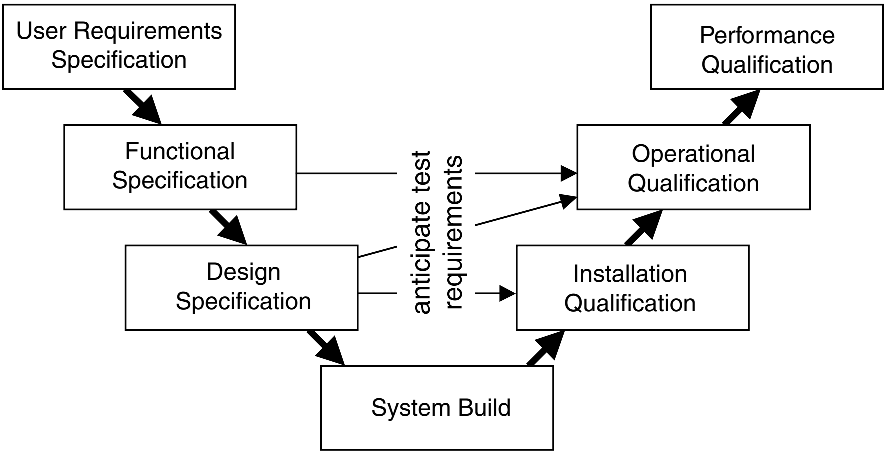

User Requirements Specification
Functional Specification
Design Specification
System Build
Installation Qualification
Operational Qualification
Performance Qualification
anticipate test requirements

FIGURE 8.2 Anticipating Testing Requirements in Specifications.

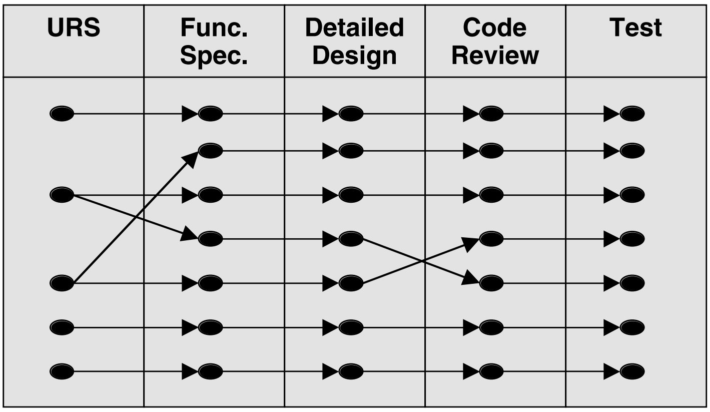


FIGURE 8.3 Requirements Traceability Matrix.

Recent Inspection Findings

- There were no functional designs included in the validation materials for the programs. [FDA Warning Letter, 1998]
- System design documentation including functional and structural design and specifications was not maintained or updated. [FDA Warning Letter, 2001]
- No original or current Functional/Structural Diagrams have been generated through the life of this program. [FDA 483, 1999]
- No system design documents included in the validation materials for the programs. There were no structural designs included in the validation materials for the programs. [FDA Warning Letter, 1998]

REQUIREMENTS TRACEABILITY

Requirements should be traceable through the design specification, source code, development testing, and user qualification. In ISO terminology, traceability demonstrates that design input is linked to design output and has been verified. For example, does the Functional Specification fulfill the User Requirements Specification? This information is often recorded in a tabular form commonly known as a Requirements Traceability Matrix (RTM). The principle of an RTM is shown in Figure 8.3. Completed RTMs should be retained as part of the Validation Package.

The use of RTMs is not limited to the lifetime of a project. They are equally useful for change management, determining the scope of testing to address design revisions. RTMs can also be used to support inspections later by quickly identifying where certain aspects of the system are tested. Traceability is one of the most important ways in which quality can be built into software, and should *always* be checked by the auditor during Supplier Audits.

Contents

The structure and format of specification, design, and test documents may have a major impact on the traceability process. These factors should be considered before the structure of the RTM is fixed. The RTM may exist, for instance, as one or more tables. Each table may have a slightly different structure depending on any special needs. An example RTM is provided in Table 8.1.

RTMs are usually created immediately after, or in parallel with, the Functional Specification. The Functional Specification represents the contractual system definition. The URS is then

TABLE 8.1
Example RTM Extract

URS Reference	Functional Specification Reference	Design Specification Reference	DQ Reference	Source Code Review Reference	IQ Protocol Reference	OQ Protocol Reference	PQ Protocol Reference	Change Control Reference
1	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable
1.1	1.1	1.1, 1.7	1.1, 1.2	Not Applicable	IQ-ABC-001	Not Applicable	Not Applicable	Not Applicable
1.2	1.2, 1.4	2.3	1.3	1.5	Not Applicable	OQ-ABC-001	Not Applicable	Not Applicable
2	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable
2.1	2.1	2.1, 2.3, 2.7	2.1	2.2	Not Applicable	Not Applicable	Not Applicable	CC-001
2.2	2.2	2.1, 2.7	2.2, 2.3, 2.4	4.3	Not Applicable	OQ-ABC-002 OQ-ABC-003 OQ-ABC-004	PQ-ABC-001	CC-002
Not Applicable	2.2.1	2.2.1, 2.2.2	2.5	Not Applicable	Not Applicable	OQ-ABC-005 OQ-ABC-006	PQ-ABC-001	CC-003
2.3	2.3	2.1, 2.7	2.6	Not Applicable	IQ-ABC-002	Not Applicable	Not Applicable	Not Applicable
7	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable
7.1	7.1	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable	CC-005
7.2	7.3, 7.6	Not Applicable	Not Applicable	Not Applicable	IQ-ABC-021	Not Applicable	Not Applicable	Not Applicable
Not Applicable	7.2.1	7.2.1, 7.2.6	3.1	Not Applicable	Not Applicable	OQ-ABC-037	PQ-ABC-022	Not Applicable
7.2	7.3, 7.6	Not Applicable	Not Applicable	Not Applicable	Not Applicable	OQ-ABC-037	PQ-ABC-023	CC-006
8	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable	Not Applicable
8.1	8.7	Not Applicable	Not Applicable	Not Applicable	Not Applicable	OQ-ABC-038	PQ-ABC-024	Not Applicable
8.2	8.7	Not Applicable	Not Applicable	5.6	Not Applicable	Not Applicable	Not Applicable	Not Applicable

retrospectively mapped against the Functional Specification. Subsequent validation activities are prospectively mapped.

Maintenance

Manual maintenance of RTM documents can be labor intensive and error prone, so it is recommended that the availability of software solutions or CASE tools for managing traceability information be considered as an alternative. Spreadsheet applications are an obvious choice in simple cases. For more complex document sets, a more sophisticated automated solution is much more likely to be up to the job.

Recent Inspection Findings

- Your response fails to trace back to source code, and the related software development cycle that establishes evidence that all software requirements have been implemented correctly and completely and are traceable to system requirements. [FDA Warning Letter, 2001]
- Your response did not evaluate requirements to trace changes to determine side effects. [FDA Warning Letter, 2001]
- The software currently in use included no description of how thorough test coverage is to be achieved. [FDA Warning Letter, 1997]

ARCHITECTURAL DESIGN

Large computer systems often benefit from an architectural design to define the structure and organization of subcomponents comprising the computer system. In essence it provides the link between the Functional Specification and the detailed design documentation. The use of diagrams to explain structures should be encouraged. Indeed, a high-level overview diagram of the software is expected by some GMP regulatory authorities.[6] An example architectural diagram is shown in Figure 8.4. Architectural Designs should only be used for configuration management where the frequency of change is sufficiently limited to make this feasible.

Architectural Designs should clearly identify the use of COTS software and hardware. Many modern computer systems make extensive use of COTS products. Architectural Designs should be included in the Requirements Traceability Matrix (RTM) as they will provide the vital linkage

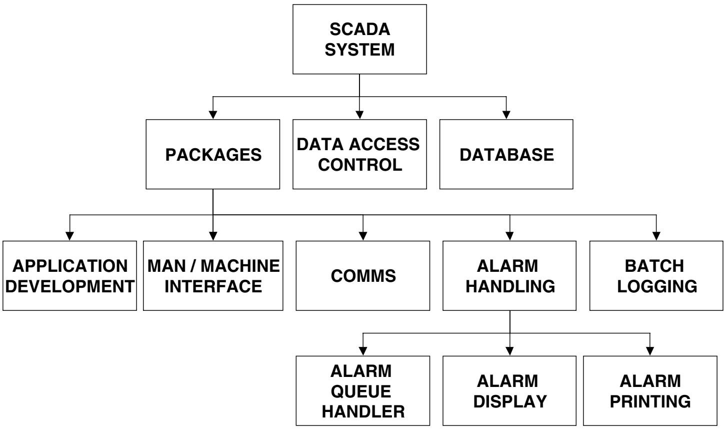


FIGURE 8.4 Architectural Diagram.

between requirements and responding design details. They should also be included within the scope of Design Reviews.

Recent Inspection Findings

- System managers must maintain a System Definition SOP including a high-level description of the software architecture, general considerations, responsibilities, and a list of current user manuals. [FDA 483, 1999]
- The firm has failed to generate approved high-level system definition documents explaining the systems architecture and functions. [FDA 483, 1999]
- An overview with high-level specifications was presented by the firm in support of validation. However, this document was not a controlled record and it lacked review and approval. [FDA 483, 2001]
- The System Architectural Diagrams do not document that the Quality Unit has approved the diagrams. [FDA 483, 2002]

SOFTWARE AND HARDWARE DESIGN

Software and Hardware Design may be segregated as two discrete activities, as described here, or combined. In either case, the design describes the implementation of the Functional Specification in increasing levels of detail until the components of the design (hardware or software) can be mapped directly to a standard product or implemented as bespoke elements of the computer system. The increasing levels of detail may be partitioned into different documents for larger systems. It is important to realize that there will be some feedback of information in the design process as the design is refined. The combined contents of the design documentation should address all the items listed in Appendix 8C and Appendix 8D at the end of this chapter. They should be cross-referenced to the Functional Specification to demonstrate how it is being fulfilled. Again, any omissions and assumptions should be unambiguously stated.

The design must ensure that records generated by computer systems conform to their specified content.[7] Records covering computer inputs, outputs, and data must be accurate.[7–9] The computer system should also include built-in checks of the correctness of data that are entered and processed. Manual entry of critical data should be subjected to a confirmation check within the design that may be either manual or automatic.

Software Design

The Software Design partitions the Functional Specification into operational units, referred to as modules. Some modules may be suitable for implementation with COTS software packages, in which case the software packages and any configuration requirements should be defined. Other modules will require custom (bespoke) programming.

Computer system software can be divided into five categories, as defined in Chapter 5:

- Operating System
- Firmware
- Standard Software Packages
- Configurable Software Packages
- Custom (Bespoke) Programming

A list of the software used should be prepared defining the program names, their purpose, version, and the language adopted in their use (not the language in which they were written). The list may exist as a document in its own right or as part of another document. For custom (bespoke) software, version numbers may not be known until design and development has been completed, at which point the list of software should be updated.

Care must be taken when developing application software that uses a COTS software package. Here is an example: A hospital used a COTS software package to calculate the dosage of a drug based on a patient's height and weight. A student then developed some front-end application software that allowed dispensary staff to calculate dosages, based on imperial units rather than the package's metric units. The application software was regularly modified over a number of years, during which time there was no change control, no supporting documentation was produced, no comments were embedded in the software code, and no acceptance testing was performed. Subsequently, over 2 years after the application software was written, it was discovered that the conversion between imperial and metric values had been incorrect all along. It is understandable in such circumstances why the suppliers of COTS software packages include a limited warranty clause in their license agreements. A typical one is reproduced below:

> *Although the software producer has tested the software and reviewed its associated documentation, the software producer makes no warranty or representation, either expressly or implied, with respect to the software or documentation, its quality, performance, merchantability, or fitness for a particular purpose. The licensee assumes the entire risk with regard to the use of the software and documentation.*

The design of custom (bespoke) modules should start by defining inputs, functionality, and outputs. Where appropriate, a hierarchy of modules may be described with submodules. Inputs, outputs, and internal module values provide data structures; consideration should be given to the grouping of data items within databases and their access. There may be more than one design document, depending on the use of standard software packages and the amount of bespoke programming. Both used and unused aspects of COTS software should be defined in the document, along with self-test and diagnostic facilities.

In many organizations, the software design process remains *ad hoc*, in other words, the traditional craft-based approach founded solely on a programmer's creativity, ingenuity, and apprehension of the functionality required. Given the Functional Specification, usually written in the vernacular or everyday language, an informal design document is prepared based on this. Coding commences, but the design intent gets modified as the system is implemented. Once the software is complete, the design may bear little resemblance to the original software design document. In such circumstances, the design document must be rewritten to fully reflect the final design. The design document must also not be too brief. This is often the case when the design evolves through a prototyping exercise. Programmers who develop prototypes are often very reluctant to fully document their resultant designs. Sam Clark, a former FDA investigator, recalled an inspection where the entire design description for one computer program consisted of the statement *"This program modifies standard errors."*[10] The design description for a category of computer programs in another system was simply defined as *"Those collating data each time a unit is tested or whatever."* The project manager must curb this tendency among programmers so that a meaningful and complete design is produced.

More methodical design approaches are available from "structured methods." These are sets of notations (a more structured language form aimed at eliminating the ambiguities of everyday speech) and guidelines that orchestrate a finished software design before coding begins. Some examples of widely used structured methods include the following:

- Yourdon, Jackson System Development, and MASCOT, which can be used for real-time system development
- Shlaer–Mellor Object-Oriented Analysis for batch control

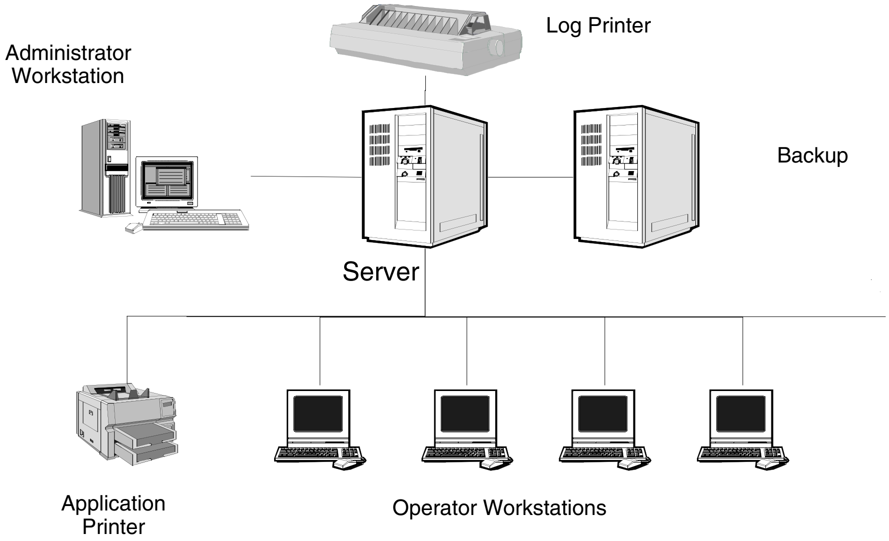


FIGURE 8.5 Example High-Level Hardware Diagram.

No particular design methodology is being advocated here. When selecting a suitable design methodology, the skills and experience of the project team applying the methodology must be taken into consideration. Furthermore, the availability of Computer-Aided Software Engineering (CASE) tools and Integrated Projects Support Environments (IPSE) supporting the methodology must be considered. It is recommended that development tools, such as CASE and Real Time System-Analysis System-Design (RTSASD), used to support validation, should be examined to bolster confidence in their capability. The reasons for selecting a particular design methodology should be clearly documented.

Hardware Design

The Hardware Design describes the equipment constituting the computer system and its configuration, in readiness for installation and commissioning. Some GMP regulatory authorities expect to see a high-level schematic diagram of the system's equipment.[6] An example of a high-level hardware diagram is shown in Figure 8.5. The equipment will be comprised of instrumentation, computer-related hardware including communication links, and any interface devices, other computer system equipment, and operations staff. The Hardware Design is likely to consist of a number of documents and make reference to preassembled information from suppliers providing standard models of equipment.

Computer hardware includes all equipment making up the computer system: processing units, user screens, keyboards, printers, and network interfaces. Switch settings and firmware configuration need to be specified. Some computer systems may have elements of computer hardware distributed at a central site or in a central computer room. User screens and keyboard inputs are often placed at distributed locations to be closer to the operator's place of work. Intrinsically safe user screens and sealed keyboards may be required in some classified areas. If the computer system makes use of a fingerprint-sensitive mouse for biometric user identification, this should be specified also. Details of bar-coders must also be defined where used. The operational limitations of hardware and other key parameters for individual items of equipment must be specified, along with any calibration or special maintenance requirements. Designated spare and redundant parts should be included in the Hardware Design documentation.

Instrumentation will include field instruments used in the manufacturing process and other instruments associated with special tasks, such as that of monitoring laboratory or computer room environmental conditions. The accessibility of instruments must be such as to permit their cleaning and maintenance. Siting is also important, and instruments should be installed as close to the point of measurement as possible. The placement of flowmeters in piping dead-legs should be avoided. Careful consideration should also be given to the appropriate position of other instruments such as thermometers and thermocouples so that they, too, can fulfill their measurement and control functions.[5,7] Construction materials that come into direct contact with the pharmaceutical or healthcare production process stream must not contaminate or affect the manufactured product in any way. Instrument lubricants and coolants must not come into contact with in-process product or equipment. The reliability of instruments should also be considered; for instance, a pressure transmitter that uses atmospheric pressure as its reference may suffer from poor reliability.[11] A draft calibration schedule may also be prepared.

Computer systems may need an Uninterruptible Power Supply (UPS), or protection from electrical interference such as Electro-Magnetic Interference (EMI), Electro-Static Discharge (ESD), and Radio Frequency Interference (RFI). Sometimes, chart recorders will be needed to monitor the temperature and humidity of the installation site. The accumulation of dust is another danger that can lead to the impairment or breakdown of equipment. Even fire and flood should be considered. The performance or limitations of computer hardware must not constrain the effectiveness of the Software Design. The software may require minimum CPU performance specifications to execute effectively. Memory size and other features of the system (e.g., hard disk, RAM, cache, DAT, CD-ROM) may also be critical to performance. The clock accuracy also needs defining closely, taking care to specify the frequency of the power supply (50 Hz in European countries but 60 Hz in North America), as this could have a fundamental impact on clock performance. Databases should not operate at capacity, and access contentions between multiple processes or users should be examined. Additional memory and memory management software may be needed to improve the operating speed of any computer system. The transmission limitations of communication links — Local Area Networks (LANs) and Wide Area Networks (WANs) — should not render any real-time processing requirements impossible to achieve.

Dealing with COTS Software and Hardware

It may be tempting to assume that COTS software and hardware needs no validation since they come from a reputable source and are market-tested. This is often an invalid assumption, as many products have unknown provenance and supplier auditing is frequently impractical. In the U.K. such software is sometimes referred to as Software of Unknown Pedigree (SOUP). Soups, while often tasty, are sometimes thick and murky — one cannot see to the bottom of the bowl! The basic validation requirements for COTS software and hardware should answer the following questions (based on *Off-the-Shelf Software Use in Medical Devices*[12]):

- What is it? (Provide title, version, etc., and state why this product is appropriate to fit the purpose of the design.)
- What are the computer system specifications? (Specify hardware, operating system, drivers, etc., including version information.)
- What function does the COTS software/hardware provide?
- How will you ensure that appropriate actions are taken by the end user? (Specify training, configuration requirements, and steps to disable or exclude the operation of any functionality not required.)
- How do you know it works? (Describe any list of known faults.)
- How will the COTS software/hardware be controlled? (This should cover installation, configuration control, storage, and maintenance.)

- Demonstrate how the maintenance and support of the COTS software/hardware will be continued should the original developers/suppliers go out of business or no longer be able to support their products/services, or any other reason.

Chapter 14 discusses the use of standard software in more detail and how a design strategy based on exploiting COTS products (as opposed to basing the design strategy on bespoke software) can reduce the validation effort required by a pharmaceutical or healthcare company.

RECENT INSPECTION FINDINGS

- Software validation packages lack detailed specifications to test against. Without detailed specifications to test against, thorough testing cannot be performed. Without thorough testing proper validation cannot be accomplished. [FDA 483]
- There are no detailed specification documents for any of the computerized process control systems that contain sufficient information on how these systems/software were represented and developed. The only specification documents made available and referred to as the design document were the “system specifications”; however, these documents only provide a high-level explanation of what the systems do. They lack sufficient detailed description of specific and complete data structure, data control flow, design bases, procedural design, development standards, and so on to serve as the model for writing code and to support future changes to the code. [FDA 483, 2000]
- The computer system lacked documentation defining database, operating system, location of files, and security access to database. [FDA Warning Letter, 2001]
- No explanation for nonsequential file or run numbers. [FDA Warning Letter, 2000]
- The computer system uses a purchased custom configurable software package. The software validation documentation failed to adequately define, update, and control significant elements customized to configure the system for the specific needs of the operations. [FDA Warning Letter, 2001]
- Networked system can only support four interfaced systems, but had up to five systems attached. There was no validation showing this configuration to be acceptable. [FDA 483, 2000]
- System layout and wiring were not part of the validation documentation. [FDA Warning Letter, 2001]
- Documentation regarding layout diagrams were found obsolete. [FDA Warning Letter, 2001]
- Validation did not address signal lines between detection devices and computer. [FDA Warning Letter, 2001]
- Diagrams related to system layout, installation, and wiring were not part of the validation documentation. [FDA Warning Letter, 2001]
- Validation records did not address wiring diagrams. [FDA Warning Letter, 2001]
- Failure to create and maintain specifications for the software programs. [FDA 483, 2001]

DESIGN REVIEW (INCLUDING HAZARD STUDY)

Design Reviews are used to confirm that the proposed design fulfills its specification (including compliance requirements) and is suitable for its intended purpose.[13] Postponing the search to discover problems and defects until Development Testing or User Qualification is virtually certain to delay the project and to increase the overall expense. This happens for the simple reason that getting anything right the first time, rather than putting it right retrospectively, is the cheapest and quickest way of accomplishing anything. Perhaps, surprisingly, whether or not Design Reviews are used and the timing of any such use remains a business risk–driven choice and not a regulatory expectation.

In general the RTM introduced earlier in this chapter will provide verification that the design fully addresses the computer system specification. Additional checks are not necessary if the individual review of each design and development document includes such a check and there have been no omissions or ambiguities in its relationship to other documents.

To verify that the design is not only complete but also fit for purpose requires a review to identify threats and controls affecting computer system operability. Three techniques are described in this chapter: HACCP, CHAZOP, and FMEA. Computer system failure (complete, partial, or intermittent) may have an adverse effect upon drug product quality, pack integrity, or regulatory compliance. The process being controlled by the computer system must be brought into a safe condition following a failure in order to protect the integrity of the process.[14] If risks cannot be managed to an acceptable level, the computer system cannot be considered fit for purpose and must not be used. The nature of this review often prompts recursive refinements to the design.

The results of the Design Review must be documented, ideally using report forms, and any corrective actions that arise from the review received by consensus and owned by a member of the review team. The actions must be described in detail in the meeting record, with a proposed completion date. The actions must only be signed off and dated as accepted when evidence of their completion is forthcoming. Some actions may require modification in the system design and associated documentation. Other actions may identify specific Development Testing or User Qualification test scenarios. Alternatively, operating procedures may need to be developed or refined. A final report detailing the personnel involved with the review, the topics considered, and the findings of the review are usually written when all of the review actions are completed. The report should be kept under change control.

Ian Johnson, now of AstraZeneca, reports several examples where design weaknesses could all have been revealed in a Design Review; these concerned a fluid bed dryer, a tablet press, and a water deionizer system.[15] The alarms in the fluid bed dryer were not latched by the embedded PLC. Alarms should have remained flagged until acknowledged by an operator pressing the "alarm reset" button. Four alarm conditions occurring in the same fluid bed dryer did not sound a horn or illuminate flash warning lamps because they were dependent on another status condition being present. This was incorrect. Again, a tablet press had a PLC compression force monitoring system with a compressed air–driven reject mechanism. When the compressed air supply failed, no audible alarm or visible warning was triggered (flashing lamp, audible horn, or a message displayed on an operator terminal). With this particular tablet press, a tablet that should have been rejected but was in fact accepted passed into the acceptance chute. This situation presented a direct threat to product quality. In another case, the PLC-controlled deionizer did not isolate its water supply when the PLC failed. There were no hardwired alarms or interlocks to alert operations staff and protect the body of deionized water already in the plant, thereby directly threatening drug product quality. A Design Review would have provided early warnings of these deficiencies before the computer systems were implemented.

Hazard Analysis and Critical Process Point (HACCP)

The application of HACCP is mandated for Food and Drink GMPs,[16] and the FDA is exploring its further use for pharmaceutical and healthcare GMPs.[17] It provides a process-orientated approach to identifying and reducing known and potential hazards to an acceptable level. The technique is not specifically intended for computer systems but its principles can be applied with a little modification (see Figure 8.6).

HACCP is best applied by a multidisciplinary team reviewing the design of a computer system. Team members are selected for their particular knowledge of the production process, the computer system, and the software.

Each hazard is assessed in terms of whether it is suitably controlled, either directly by the computer systems affected or by a backup mechanism or by another computer system. The use

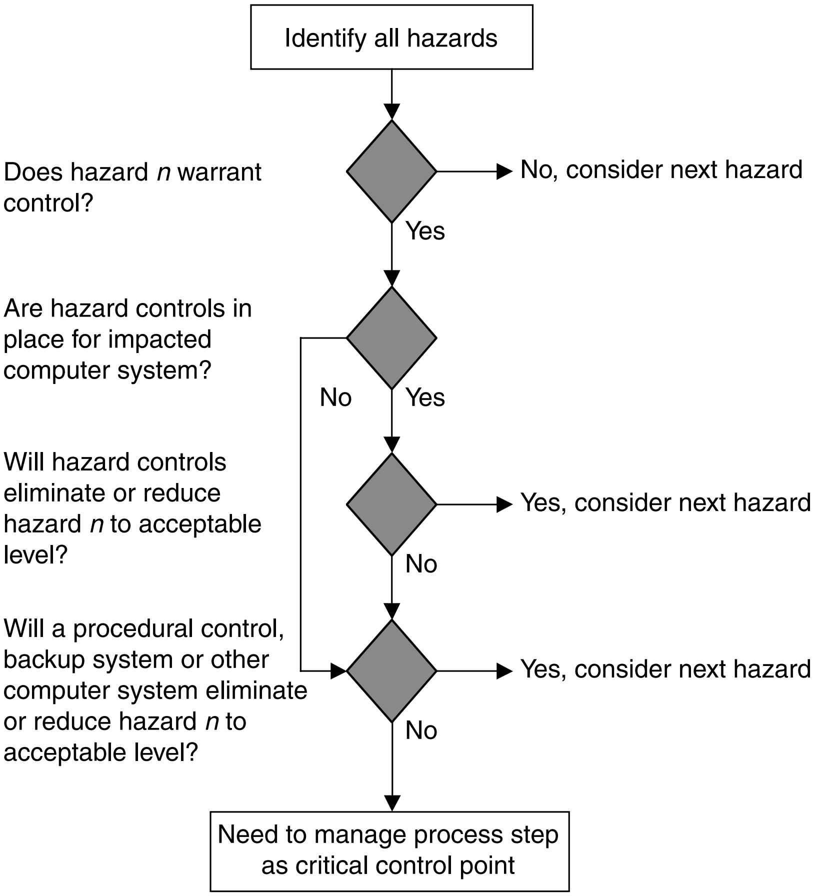

Identify all hazards
Does hazard *n* warrant control?
No, consider next hazard
Yes
Are hazard controls in place for impacted computer system?
No
Yes
Will hazard controls eliminate or reduce hazard *n* to acceptable level?
Yes, consider next hazard
No
Will a procedural control, backup system or other computer system eliminate or reduce hazard *n* to acceptable level?
Yes, consider next hazard
No
Need to manage process step as critical control point

FIGURE 8.6 Critical Control Point Decision Tree. (Based on FDA (1997), *Hazard Analysis and Critical Point Principles and Application Guidelines*, National Advisory Committee on Microbiological Criteria for Foods, August.)

of backup mechanisms (backup systems and procedural controls) were discussed earlier in Chapter 7. Hazard controls provided by supplementary computer systems are based on the functionality of existing computer systems used elsewhere. More than one hazard control may be used to manage individual hazards. Equally, more than one hazard may be mitigated by an individual hazard control. Hazards not satisfactorily controlled are considered critical control points and will need to be addressed.

An HACCP report should be prepared outlining the process involved, identifying hazards, detailing how individual hazards are controlled, and presenting any resultant recommended actions for hazards that require further controls. The report should be reviewed for its accuracy, and signed and approved accordingly. Actions should be reviewed for satisfactory closure as part of the Validation Report. Actions may affect specification and design documentation, and/or testing.

HACCP provides a very basic approach to hazard analysis and control. Computer systems supporting licensed products with medicinal properties should consider more rigorous techniques, such as CHAZOP and FMEA, described below.

Computer Hazard and Operability (CHAZOP) Study

During the 1960s, ICI developed the HAZOP (Hazard and Operability) process to identify hazards in chemical plant design. The HAZOP process is now well established, and was successfully

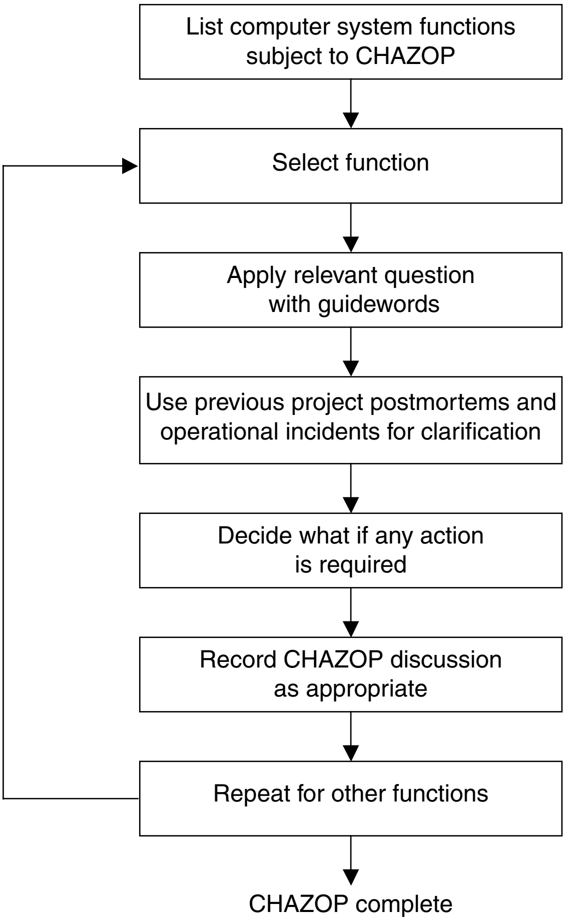

List computer system functions subject to CHAZOP
Select function
Apply relevant question with guidewords
Use previous project postmortems and operational incidents for clarification
Decide what if any action is required
Record CHAZOP discussion as appropriate
Repeat for other functions
CHAZOP complete

FIGURE 8.7 CHAZOP Process.

extended by ICI during the 1980s to include computer control systems. The Computer HAZOP (CHAZOP) process is shown in Figure 8.7.

During the CHAZOP Study meeting, the team will go through diagrammatic representations of the system. The experience of the CHAZOP leader will steer the review team through the configuration of the computer system, the software control scheme, the effect of GMP-related functions on the process, and the general operability and security of the computer system. Possible deviations from the design intent are investigated by systematically applying guidewords. Some example guidewords are given below, based on CHAZOP's use in the chemical industry:[19]

- on, off, interrupt
- as well as, only, other than, part of
- no, not, wrong
- less, none, more
- reverse, inverse
- more often, less often
- early, late, sooner, later, before, after

All guidewords to be used by the CHAZOP review team should be defined and documented. Care must be taken to ensure that all CHAZOP participants understand their precise meaning and whether or not they are suitable in the context in which they are being used. The CHAZOP chairman should remove inappropriate guidewords from the study. At the discretion of the study chairman, new guidewords may be added as appropriate to the computer system being reviewed.

The guideword process can be supplemented by additional topics/questions based on an analysis of previously experienced design deficiencies and operational incidents. For instance, ICI has collated a database of over 350 operational incidents that it uses to refine its CHAZOP Study process.[20] Some example questions for the CHAZOP Study are given in Appendix 8E at the end of this chapter. Of particular interest to the study is the effect of partial or catastrophic failures, recovery mechanisms (e.g., rollback and roll-forward), and the general usability of the system (e.g., the need for multiple screens to access data, screen refresh times, meaningful information displays). The list of questions can be expanded with operational and regulatory experience.

CHAZOP is based on a multidisciplinary team reviewing the design of a computer system. Team members are selected for their particular knowledge of the production process, the computer system, and software programs. The CHAZOP meeting is led by a chairman to manage discussions, using guidewords and learning from earlier studies. A holistic approach is required covering hardware failures, software functionality, and human factors (manual dependencies).

A CHAZOP report should be prepared outlining the process undertaken, the definition of guidewords used, and any resultant recommended actions. The CHAZOP report will normally include the completed CHAZOP table. An example of part of a CHAZOP table is shown in Table 8.2. The report should be reviewed for its accuracy, and signed and approved accordingly. Actions should be reviewed for satisfactory closure as part of the Validation Report. Actions may affect specification and design documentation, and/or testing.

The main strength of the CHAZOP process is that it facilitates systematic exploratory thinking. The use of guidewords and deviations prompts the review team to think of hazards that might have otherwise been missed. Recently, both the U.K. Department of Defense[21] and the U.K. Health and Safety Executive[22] have issued a CHAZOP standard. It is important to recognize that the effectiveness of CHAZOP studies is dependent on guidewords and the capture of learning from project postmortems and operational incidents. The relevance of guidewords only becomes evident through practical application. The significance of any learning will be dependent on routine collection and analysis of project postmortems and on the reliable reporting of operational incidents.

Failure Mode Effect Analysis (FMEA)

FMEA provides a hardware-orientated approach to identifying and reducing known and potential hazards to an acceptable level. The FMEA process in Figure 8.8 is based on FDA medical device guidance.

FMEA is best applied by a multidisciplinary team reviewing the design of a computer system. As with CHAZOPs, team members are selected for their particular knowledge of the production process, the computer system, and the software. The team steps through various computer system failures, considering their effect, risk, and how they might be controlled. The outcome of a FMEA is then documented in a template such as the example given in Table 8.3.

More sophisticated FMEAs examine the level of concern over various hazards in terms of GxP criticality.[23] Figure 8.9 describes how to determine three levels of concern: low, medium, and high. The decision tree presented considers only the impact on drug product quality. Some pharmaceutical and healthcare companies may want to include operator safety, business impact, and even the GAMP categories of software affected in their determination of these levels of concern.[24]

Once the level of concern is understood, the FMEA team needs to appraise its likelihood. Remote and rare events may be acceptable without further action. The term ALARP is often used for such events; it means reduce risk as low as is reasonably practical. Higher likelihood hazards will demand more attention, as indicated in Figure 8.10. The acceptance of a hazard without control needs to be justified. It is important to note that the FDA does not include "likelihood" in its guidance

TABLE 8.2
Example CHAZOP Extract

Operating Sequence	Deviations — Using Guidewords	Possible Causes	Consequences	Action Required
User log-on	Wrong	User forgot password	Retry; lock-out after three unsuccessful attempts; user then has to formally apply to Security Manager for new password	None; users trained to keep passwords secure and not to share them
		Unauthorized access attempt	Retry; lock-out after three unsuccessful attempts	SOP for periodic review by Security Manager and escalation to senior management as appropriate
	As well as	User already logged on at another terminal	If more than one terminal being used by a user then likelihood is that at least one has been left unattended and hence is insecure	Recommend design altered to only allow one active terminal per user at a time
Enter product code and analytical method	Other than	Invalid product code or analytical method entered	Option given to user to reenter or cancel process initiated and return to main menu	None
Enter analytical data	More, less	Invalid numerical data entered such as alphabetic characters, zero, or negative values	No check made on range of data entry; may cause data calculation error such as divide by zero elsewhere in system	Recommend automatic range checks included in design
Completing user transaction processing	Interrupt	Transaction not completed due to partial system failure or catastrophic system failure	No commitment of data to database and roll-back, so all transaction data lost	None, except manual reentry of transaction data

for medical devices. Instead, the FDA focuses on reducing the "level of concern," essentially because any occurrence of a failure in a safety-critical medical device could be catastrophic.

There are many methods of control to eliminate or mitigate identified hazards. The selected controls for a hazard should be recorded on the FMEA template. Examples of hazard controls include change in design specification, implementing alarms/warnings/error messages, and instituting a manual process. The most cost-effective hazard control should be selected and described in the FMEA template.

An important factor to consider when examining suitable controls for identified hazards is the time required to restore the full integrity and operability of the computer system if the hazard occurs, and whether and when to switch over to manual operation. Recovery times are very important for delay-sensitive processes. Control options include:

- Hazard mitigation (protective measures)
- Hazard avoidance (inherent safe design)
- Hazard tolerance (user procedures and training)

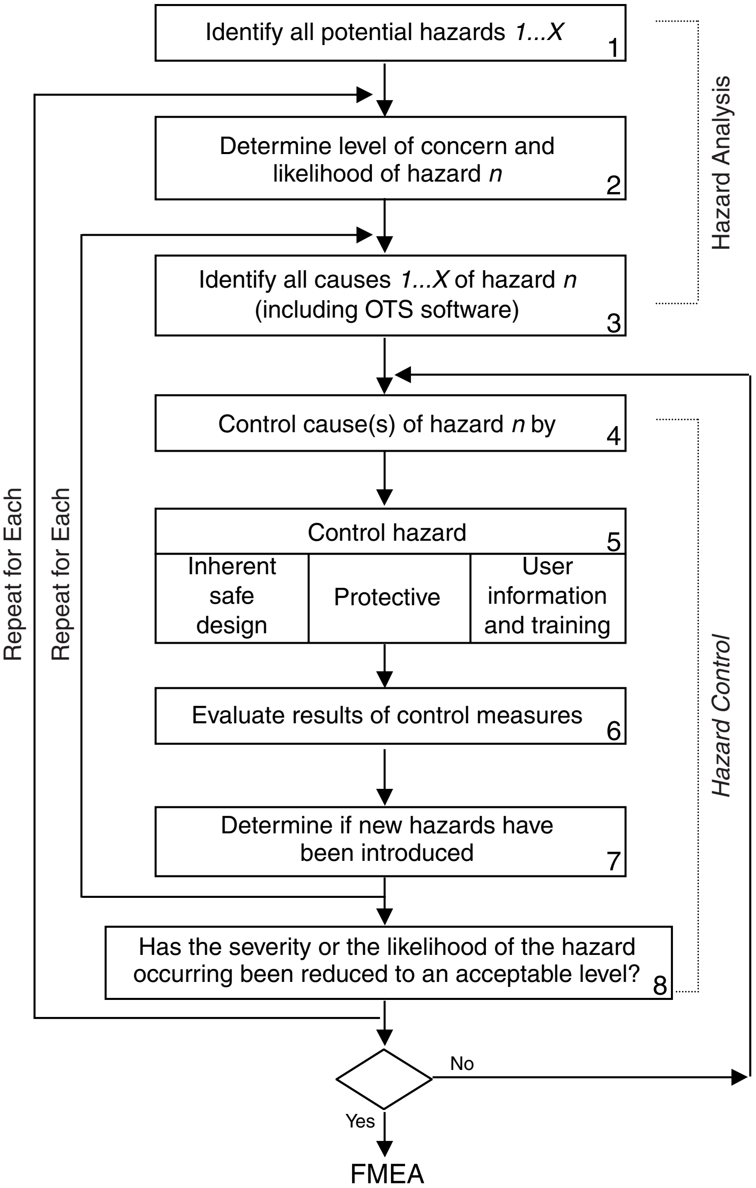

Identify all potential hazards 1...X
1
Determine level of concern and likelihood of hazard n
2
Identify all causes 1...X of hazard n (including OTS software)
3
Hazard Analysis
Control cause(s) of hazard n by
4
Control hazard
5
Inherent safe design
Protective
User information and training
Evaluate results of control measures
6
Determine if new hazards have been introduced
7
Has the severity or the likelihood of the hazard occurring been reduced to an acceptable level?
8
Hazard Control
Repeat for Each
Repeat for Each
No
Yes
FMEA

FIGURE 8.8 FMEA Process. (Based on European Union, *Annex 15 — Qualification and Validation*, European Union Guide to Directive 91/356/EEC.)

Careful consideration should be given to how best to verify the hazard control (i.e., validation activities/documentation). Hazard controls that cannot be verified as being in working order will not satisfy regulatory authorities during inspections.

A modified FMEA template is provided in Table 8.4. This calculates relative risk before and after hazard controls are applied.[24] Risk is calculated as a function of the likelihood of occurrence, the severity of the hazard, and the probability of detection. Oliver Muth of Pfizer suggests the calculation of a Risk Priority Number (RPN), calculated from multiplying the scores given to likelihood, severity, and detection.[25] The *likelihood* of an occurrence is rated on the following scale:

TABLE 8.3
Example FMEA Extract

Failure Mode	Effect	Hazard	Hazard Control	Control Verification
Control system power failure	Operator console goes blank	Facility to collect process data is lost — batch rejected	Recommend addition of UPS	IQ check UPS supports continued control system operation during power outage
Control system is defective (short-circuit)	Unknown — vendor expects system to freeze and/or operator console go blank	Facility to collect process data is lost or corrupted — batch rejected	None; remote chance event will occur — replace unit on failure, not financially viable to include second redundant control system into architecture	Not applicable
Incorrect set-point downloaded from control system to product critical instrument	Equipment appears to operate normally	Unacceptable temperature in manufacturing process not detected	Recommend addition of independent monitoring system to check process temperate	Validation of independent monitoring system
Wire break on control system interface to equipment	Instruments run on last known set-point, alarm conditions are not received by control system	Equipment does not stop; operator might not notice monitored process values are unduly static	Recommend control system design modified to alarm on no-instrument signal	OQ check alarm when control system interface disconnected
Instrument power failure	Error will not automatically alarm on control system if last received process parameter within acceptable range	Operator might not notice monitored process values are unduly static	Recommend control system design modified to alarm on instrument power failure	IQ check alarm when instrument power failure

1. Remote, unlikely to occur within 2 years
2. Code tested, reviewed, and proven to be reliable
3. “Normal code” tested
4. Complex code, interfaces (automated/manual), or many variables from other parts of the application used
5. Complex code, unknown, or test results unknown

The *severity* of the hazard is marked on the following scale:

1. No negative impact
2. Results in minor deviation
3. Deviation from quality profile
4. Causes production or shipment of noncompliant product
5. Could cause injury to a patient using the product of the process

The *probability of detection* is marked on the following scale:

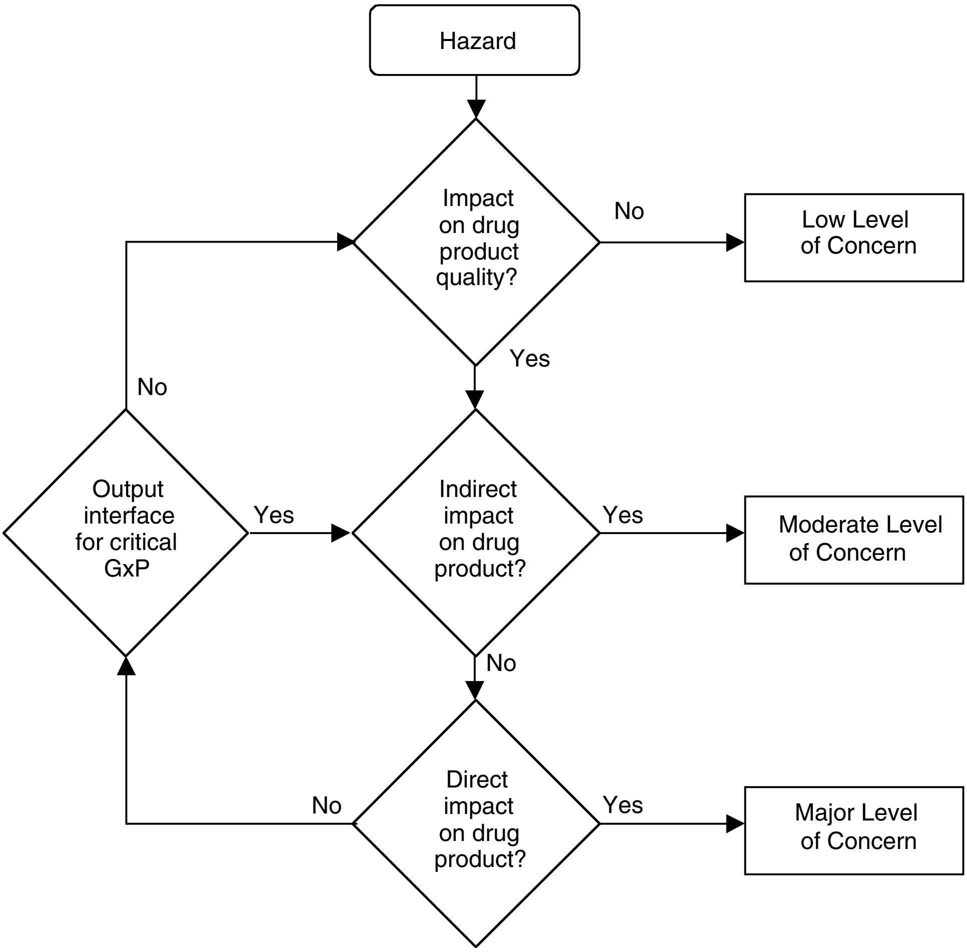

Hazard
Impact on drug product quality?
No
Low Level of Concern
Yes
No
Output interface for critical GxP
Yes
Indirect impact on drug product?
Yes
Moderate Level of Concern
No
No
Direct impact on drug product?
Yes
Major Level of Concern

FIGURE 8.9 Example Decision Tree for Determining GxP Level of Concern.

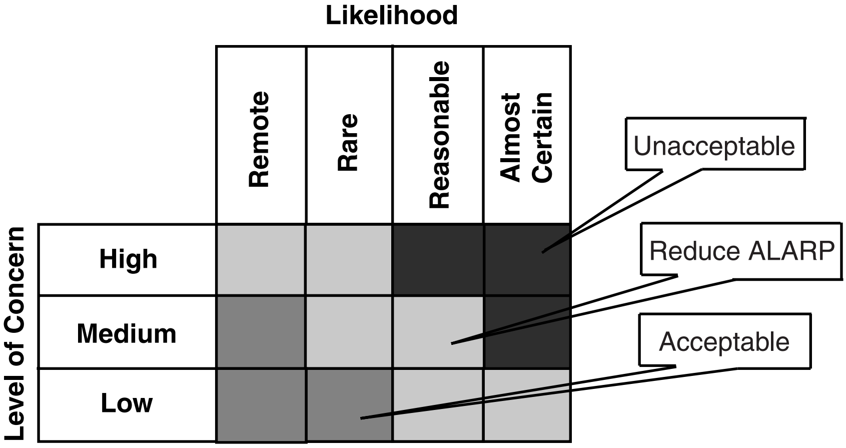


FIGURE 8.10 Managing Levels of Concern.

TABLE 8.4
Example FMEA Extract

Hazard Identification							Hazard Control					
			RPN						Residual RPN			
Description	Cause and Consequence	Level of Concern (GxP)	L	S	D	RPN	Overview Description	Control Verification (Doc. Reference)	L	S	D	RPN
Batch Failure	Steam temperature not controlled leading to excessive steam temperature	High	2	5	4	40	Add temperature alarm with control limit	DQ, IQ, OQ	2	5	1	10
Batch Time Increased	Steam temperature not controlled leading to increased batch time	Low	2	3	4	24	Add temperature alarm with control limit	DQ, IQ, OQ	2	3	1	5
Batch Traceability Lost	Corruption of data on save leading to erroneous batch data	High	3	4	4	48	Add checksum check on data save	DQ, Source Code Review, IQ, OQ	3	4	1	12
Batch Traceability Lost	Unauthorized data modification leading to erroneous batch data	Medium	4	4	4	64	Put application on network with password security and authorized user access profile	DQ, Source Code Review, IQ, OQ	1	4	2	8
Batch Traceability Lost	No backup of data leading to absent batch data	High	2	4	4	32	Put application on network and configure for daily automatic backups	DQ, IQ, OQ	1	4	3	12

1. 100% automated detection and alarm
2. Sub-100% detection, independent automated detection and alarm
3. Combined manual and automated detection
4. Detected through routine manual quality-control checks
5. Not detectable, or normally not tested for

It is important to appreciate that risk is a relative concept. An unacceptable high risk might be an RPN score exceeding 24. All RPN scores below 13 would indicate a low risk (and therefore perhaps acceptable). RPN scores between 12 and 25 might indicate an acceptable risk, but one that should be reduced if it is possible to do so within the practical constraints imposed on the project.

Dealing with COTS Software and Hardware

Configurable and customized COTS software and bespoke (custom) hardware should be subject to an HACCP, CHAZOP, or FMEA as appropriate. It is vital that GxP processes are not compromised. The following questions should be rhetorically posed:

- What does the computer system do?
- How might it fail to do what it is supposed to do?
- What causes these failures?
- What is the impact/consequences of these failures?
- How are these hazards to be controlled?

Hazards requiring control can be managed through design modifications (including using alternative COTS products), employing protective measures (e.g., monitoring systems to identify hazard manifestations and to take corrective action), or the application of procedural controls.

Nonconfigurable COTS software and standard hardware do not require an HACCP, CHAZOP or FMEA if operational experience exists to support the view that the COTS products are stable and robust. Operating experience should be considered suitable when the following criteria are met:[26]

- The intended version COTS product has achieved a sufficient cumulative operating time.
- The intended COTS product has operated in several similar installations (and hence there has been more chance that hidden errors would be exposed).
- Defects/errors are routinely reported as they occur and are corrected in a timely fashion.
- Meaningful information on reported defects/errors, remedial actions, and status are available.
- No significant modifications have been made, and no errors have been detected over a significant operating time, recommended as at least one full year of operation.

The rigor of the analysis of operating experience should be commensurate with the GxP use of the computer system. Operating experience should be under conditions similar to the conditions during intended use of the computer system. When the operational time of other versions is included, an analysis of the differences and history of these versions should be made.

Recent Inspection Findings

- Failure to establish and maintain procedures for verifying the XXXXXX design. [FDA Warning Letter, 1999]
- No documented risk assessment and hazard analysis was done. [FDA 483, 1996]
- The decision not to perform a hazard assessment for XXXXXX was not justified. [FDA Warning Letter, 1999]

- The firm’s protocol indicated three levels of risk, the lowest level required no validation and the highest level required full validation. The procedure did not provide for recourse in the event of a mid-range risk. [FDA Warning Letter, 1999]

ACCELERATED DEVELOPMENT

It is often tempting to implement tools to support design techniques as part of the system/software development processes. Care must be taken, however, not to get carried away with technology. Many tools that promise improved productivity and quality are in the market. While these tools may be designed to support particular development techniques/processes, they are often used out of context. In consequence, the tools are not exploited fully and may even be misused. The promised benefits may not materialize.

PROTOTYPING

It is often difficult to secure and maintain a sufficiently accurate or comprehensive definition of a proposed system’s requirements from a prospective user. Prototyping has proved to be a useful approach whereby one or more working models are developed and used as an explanatory model before the real system is implemented. Such working models can be examined by the prospective users to exercise their own understanding of the requirements that must be met. Seeing a working model helps users clarify, explain, refine, and, most important, *express* their requirements. Care has to be taken to prevent indiscriminate scope creep, with the prototype taking on a whole life of its own and running away with time and money. Extending the scope beyond requirements to include marginally desirable rather than only essential features (“nice to have”), can become an irresistible temptation to excited prospective users. This must be curbed. Some advantages and disadvantages of prototyping are outlined in Table 8.5.

Typically, prototypes are discarded once the functional requirements and user interface requirements have been fully defined. The final system is then developed through a prospective life cycle of design, coding, configuration and build, development testing, and user qualification. QA need not be involved with the development of prototype software since it is destined to be discarded and does not have to have any significant level of innate quality.

Prototypes, however, do not necessarily have to be discarded. They can be taken forward as the final application. The implications of this approach should be clearly understood before proceeding as there are great dangers here. Retrospective documentation of prototype systems can cost 25 to 50% more than if documentation were drafted prospectively. Some reengineering is likely to be required; this is expensive, dangerous, and rarely delivers the quality that is engineered into a conventional prospective development. Project managers may find themselves under pressure to shortcut the reengineering process, but this will compromise validation. If the prototype is being taken forward as the final application, QA involvement from the outset is absolutely essential. Retrospective validation is discussed in more detail in Chapter 13.

TABLE 8.5
Pros and Cons of Prototyping

Advantages	Disadvantages
• User needs are better accommodated	• Limited identification of hazards
• The resulting system is easier to use	• System performance may be worse
• User problems are detected earlier	• System is harder to maintain
• Less effort to realize development of system	• The prototyping approach requires more experienced team members

Rapid Application Development

Rapid Application Development (RAD) emphasizes user involvement, prototyping, software reuse, the use of automated tools, and small development teams. The RAD life cycle, sometimes called a *spiral* life cycle, consists of four basic phases:

- Specification
- Design and development
- System build
- Acceptance testing

The specification phase and the design/development phase have much in common and are often merged. They make extensive use of Joint Application Development (JAD) workshops in which developers and prospective users work together to agree on the functionality of the planned computer system and to develop the prototype. It is vital that key users are present and actively contribute to the JAD workshops by giving an *authoritative* opinion. Otherwise, functional requirements captured may be inaccurate or incomplete, and the prioritization of their requirements may be inappropriate.

The system build phase is expected to extensively reuse existing software rather than develop new bespoke code. Reused software must be robust and documented; otherwise the benefits of the RAD process will be reduced by the need for a significant reengineering and revalidation exercise.

Successful acceptance testing is used as the basis for authorizing the computer system for installation and use. The focus of the acceptance-testing phase is user qualification and user training. Development testing should also be conducted as a matter of course. Time constraints imposed by RAD projects often lead to this activity being attenuated. It is important to understand that validation must involve comprehensive development testing.

Projects implementing RAD impose closely monitored timetables within which activities must be completed. The RAD philosophy is to complete as much of the assigned activities within their designated time constraints. Consequently, aspects of activities must be prioritized so that, if necessary, lesser aspects can be sacrificed. For instance, the inclusion of low priority functional requirements may be sacrificed or the amount of testing may be limited to enable the design/development and user acceptance activities to be completed within their respective timetables. Obviously, such stringent time management raises questions about the risk of a compromised validation. Limited testing, for instance, could be seen as failing to meet validation requirements. This aspect of RAD must be carefully managed in order to avoid a costly, retrospective validation.

In conclusion, while RAD projects offer many advantages, they also pose certain risks, and thus require careful, competent, realistic management to ensure that they deliver *validated* computer systems. RAD must not be used as an excuse to circumvent necessary life-cycle controls and documentation.

Extreme Programming

Extreme programming is a relatively new approach to accelerated development. It aims to deliver software faster than any other widely used approach. It comprises a number of key practices that, it is suggested, must be collectively applied for the approach to work.[27] Superficially, some extreme programming practices may appear to contravene conventional principles of good programming practice:

- Systems are planned to go through a series of frequent, small, incremental developments (i.e., an increment every 1 to 4 weeks).
- Projects and development have no independent quality oversight. Quality is a collective responsibility.

- Systems are designed to be as simple as possible. Extra complexity is removed upon exposure.
- Programmers write all the code in accordance with rules emphasizing communication through the code, rather than via documentation.
- The inchoate code must pass all its tests before further incremental development can take place. Programmers write their own unit test scripts. Customers write their own user test scripts.
- Systems are tuned rather than developed during final incremental development, leading to authorization for use in the live production environment.

Soundly tested small incremental developments and system releases should bring a high degree of assurance that the computer system is fit for purpose. The lack of formal specification and design documentation means that traditional GxP validation requirements cannot be satisfied. Instead, design information is documented within the software itself. Testing is conducted against test specifications rather than predefined design criteria. Thus, it is not a question of validation activities not actually being undertaken but rather a case of validation activities being conducted in a different way.

Another issue with conventional validation is how quality is managed and controlled. Extreme programming relies on collective responsibility for quality. This does not sit comfortably with current regulatory expectations for *independent* quality oversight, nor, many would allege, is it in accord with what is generally expected of human nature! Given the alternative approach to specifications and design described earlier, it may be hard for regulators to accept that extreme programming can delivery high-quality code without some sort of independent project verification. Although not called for by the extreme programming approach, a Quality and Compliance role (as discussed in Chapter 3) may well be appropriate. This may be needed at least until the regulators can understand how extreme programming might be able to work in practice without independent quality oversight.

In the short term it is unlikely that extreme programming will prove to be acceptable to the regulatory authorities. Nevertheless, it should not be dismissed as a future technique once it has been more widely demonstrated as fulfilling its potential. Regulators appreciate that there are strong cost-benefit drivers throughout industry to find ever-faster ways of implementing computer systems. It may well be that tools will emerge that would enable the disciplines of a more conventional, independent measure of oversight to be exerted without slowing down the development process.

REFERENCES

1. European Union Guide to Directive 91/356/EEC (1991), *European Commission Directive Laying Down the Principles of Good Manufacturing Practice for Medicinal Products for Human Use*.
2. ISO (2000), ISO 9001-3: Quality Management and Quality Assurance Standards — Part 3: *Guidelines for the Application of ISO 9001 to the Development, Supply and Maintenance of Software*, International Organization for Standardization, Geneva.
3. FDA (1997), *General Principles of Software Validation: Guidance for Industry*, Draft Guidance Version 1.1, June.
4. FDA (1987), *General Principles of Process Validation*, Food and Drug Administration, Center for Drug Evaluation and Research, Rockville, MD.
5. Kletz, T., Chung, P., and Shen-Orr, C. (1995), *Computer Control and Human Error*, Institution of Chemical Engineers, Rugby, U.K.
6. TGA (1990), *Australian Code of Good Manufacturing for Therapeutic Goods*, Medicinal Products, Part 1, Therapeutic Goods Administration, Woden, Australia.
7. U.S. Code of Federal Regulations Title 21, Part 211, *Current Good Manufacturing Practice for Finished Pharmaceuticals*.

8. European Union, *Annex 11 — Computerised Systems*, European Union Guide to Directive 91/356/EEC.
9. FDA (1982), *Input/Output Checking,* Compliance Policy Guides, Computerized Drug Processing, 7132a, Guide 7, Food and Drug Administration, Center for Drug Evaluation and Research, Rockville, MD.
10. Wingate, G.A.S. (1997), *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice*, Interpharm Press, Buffalo Grove, IL.
11. ISPE (1995), *Bulk Pharmaceutical Chemicals, Baseline: Pharmaceutical Engineering Guide for New Facilities*, Vol. 1 (first draft), International Society for Pharmaceutical Engineering, Tampa, FL.
12. FDA (1999), *Off-The-Shelf Software Use in Medical Devices*, FDA 1252, U.S. Department of Health and Human Services, Food and Drug Administration, Center for Devices and Radiological Health, September 9.
13. European Union, *Annex 15 — Qualification and Validation*, European Union Guide to Directive 91/356/EEC.
14. FDA (1983), *Guide to Inspection of Computerized Systems in Drug Processing (Blue Book)*, Reference Materials and Training Aids for Investigators, Food and Drug Administration, Center for Drug Evaluation and Research, Rockville, MD.
15. Johnson, I. (1995), *Validation of Finished Product Equipment*, ISPE European Computer System Validation Seminar, Zurich, September 20 and 21.
16. Blanchfield, J.R. (1998), *Food and Drink — Good Manufacturing Practice: A Guide to Its Responsible Management,* Institute of Food Science and Technology, London.
17. *Gold Sheet* (2000), 34 (5), May.
18. FDA (1997), *Hazard Analysis and Critical Point Principles and Application Guidelines*, National Advisory Committee on Microbiological Criteria for Foods, August.
19. Chung, P. and Broomfield, E. (1995), Hazard and Operability (HAZOP) Studies Applied to Computer-Controlled Process Plants, in *Computer Control and Human Error* (Ed. T. Kletz), Institution of Chemical Engineers, Rugby, U.K.
20. Lucas, P.R. and Wingate, G.A.S. (1996), Threats Analysis for Computer Systems, Special Issue on Computer Systems Validation, *Pharmaceutical Engineering*, 16 (3).
21. U.K. Ministry of Defence, Defence Standard 00-58: *A Guideline for HAZOP Studies in Systems which include a Programmable Electronic System*, MOD Directorate of Standardisation, Glasgow, U.K.
22. Andow, P. (1991), *Guidance on HAZOP Procedures for Computer Controlled Plants*, Her Majesty's Stationery Office, London.
23. FDA (1998), Guidance for FDA Reviewers and Industry: *Guidance for the Content of Pre-market Submissions for Software Contained in Medical Devices*, May 29.
24. Reid, C. (2001), *Effective Validation Strategies: Put GAMP Categories into Practice*, Business Intelligence Conference on Computer Systems Validation for cGMP in Pharmaceuticals, London, March 28 and 29.
25. Muth, O. (2001), *Risk Evaluation in Pharmaceutical Software Development*, Conference on Software Validation for Healthcare, Bonn, Germany, April 4–6.
26. Jones, C., Bloomfield, R.E., Froome, P.K.D., and Bishop, P.G. (2001), *Methods for Assessing the Safety Integrity of Safety-Related Software of Uncertain Pedigree (SOUP)*, U.K. Health and Safety Executive, Contract Research Report 337/2001.
27. Beck, K. (2000), *Extreme Programming*, Addison-Wesley, Reading, MA.
28. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), International Society for Pharmaceutical Engineering (www.ispe.org).

APPENDIX 8A
EXAMPLE CONTENTS FOR SUPPLIER PROJECT/QUALITY PLANS

Introduction

- Author/organization
- Authority
- Purpose
- Relationship with other documents
- Contractual status of document

Scope

- Project description

Project Plan

- Identify project activities
- Gantt charts
- Milestones and deliverables
- Mechanism to close project
- Risk management

Project Organization

- Personnel
- Training
- Nominated contacts
- Mechanism to escalate problems
- Project progress meetings
- Problem escalation process

Quality Plan

- Customer quality requirements
  - Customer procedures
  - Independent standards
  - Independent guidelines (e.g., GAMP)
  - Supplier quality assurance system
- Customer/supplier quality assurance system interface

Deliverables

- Identify types of document to be produced through project

References

Appendices

- Glossary
- Others

APPENDIX 8B EXAMPLE CHECKLIST FOR FUNCTIONAL SPECIFICATION

System Description

- Overview of the process with which the computer system will interact (e.g., laboratory, manufacturing, packaging, labeling, business)
- High-level description of the role of the computer system
- Operational boundaries and interface requirements
- Development tools: CASE tools, Computer Aided Design (CAD)
- List of constraints including working language affecting computer system

Equipment

- Computer hardware: Central Processing Unit (CPU), Random Access Memory (RAM), memory devices (hard disk, CD-ROM, etc.), storage devices, communication interfaces to equipment and other computer systems, operator terminals
- Computer software: operating system, communication drivers, network controllers, system software, configurable programs, application programs, databases
- Instrumentation: number, type, location; model numbers, software versions; tag numbers, data types, valid range
- Controlled elements: valves, heaters, motors and motor starters, relays, solenoids, etc.
- Power requirements with tolerable operating range

Interface

- Human Interface
- Number, size, and location of screens (mimic backgrounds and foregrounds)
- Input devices (keyboards, mouse, tracker ball, touch screen, and so on)
- Number, type, and location of printers (alarm, color, report)
- Interface features (information pages, visual/audible alarms, mimics/graphics)
- Security features (levels of access authority, passwords, key switches, biometrics, logging unauthorized access attempts, and so on)
- Process I/O Interface
- Operator intervention
- I/O specification for different devices
- Installed spare capacity, including fail-safe redundancy features
- Future expansion with any implications for reduced performance
- Plant external I/O signals (type, format, range, accuracy, timing)
- System-derived inputs (calculated variables: totalizers, analog-digital converters, interposing relays, frequency, period, validity)
- Communications Interface
- Protocols, buffer, cable specifications, line drivers, termination, loading, traffic density, automatic error correction

Process Control

- Sequence Control: text, flowcharts, and state-transition diagrams
- Continuous Control: automatic and control loops, database storage, scan frequencies, complex control schemes (cascade, ratio, predictive)

- Batch Control: batch initiation, process decomposition, continuous monitors, recipe handling (timers, volumes, weights, temperatures, pressures, download, and sequence interaction), logging
- Data Processing: derived values, data conversions, scaling, frequencies, periods, calculations, algorithms, validity checks, error correction

System Attributes

- Alarms: number and priorities, groups, response times, escalation, annunciation, presentation (including color and color changes), conditioning, acknowledgments, viewing, printing and storage capacity, segregation
- Trending: real-time and historical data, histograms, balance sheets, who can configure and view trends, printing and storage capacity
- Events: number, categories, notification, viewing, logging, printing and storage capacity
- Interlocks: hardwired interlocks, scan rates, suppression, logging, reporting, acknowledgment, computer system handshakes and watchdogs, avoid deadly embrace

Operational Environment

- Performance: response times (e.g., screen refresh rates, cycle times, and critical control response times), Mean-Time-To-Failure (MTTF), system remedial action, power failure recovery, startup, shutdown
- Redundancy: dual operation, segregated I/O on dedicated cards, peripherals, interfaces, internal networks, power supplies, hard disks
- Intrinsic Safety: equipment for zoned areas
- Operational Safety: failsafe mechanisms, error handling, database integrity, operator timeouts, fault tolerance, watchdogs, contingency plans, internal and legislative safety compliance
- Operational Procedures: operator commands, override control, process monitoring, parameter modification, load scheduling, startup and shutdown, fault-finding instructions, user manuals and documentation
- Training: formal courses, hands-on training, manuals
- Security: levels, means of access, parameter modification, program access, data security
- Data Integrity: archiving, buffer storage, device storage, data recovery, backup, restoration
- Environmental Conditions: earthing, filtering, loading, surge protection, temperature, humidity, vibration, electrical interference (ESD, EMI, RFI), fire, flood, hygiene
- Maintenance: spares, special handling practices and tools, cleaning, media backups and restoration, service contracts, service instructions, calibration schedules
- Expansion Philosophy: tuning variable demand, functional changes, spare capacity, performance, physical size restrictions

APPENDIX 8C
EXAMPLE SOFTWARE DESIGN STRUCTURE[28]

Introduction

Scope

System Description

- Module structure
- Interfaces

System Data

- Databases
- Files
- Records
- Data types

Module Descriptions

- Design
- Functionality
- Interfaces
- Subprogram overview
- Software module/unit data (i.e., databases, files, records, data)

Sub-Program Descriptions

- Operation
- Input/output parameters
- Side effects
- Language
- Programming standards

Interfaces

- Operation
- Timing
- Error handling
- Data transfer

Glossary

References

Appendices

APPENDIX 8D
EXAMPLE HARDWARE DESIGN STRUCTURE

Introduction

Scope (Overview)

Design Configuration

- Main computer
- Storage devices
- Peripherals
- Interconnections

Input/Output

- Digital
- Accuracy
- Analog
- Isolation
- Pulse
- Range
- Interface cards
- Timing

Environment

- Temperature
- Humidity
- External interfaces
- Physical security
- RFI, EMI, UV

Electrical Supplies

- Filtering
- Loading
- Earthing
- UPS

Glossary

References

Appendices

APPENDIX 8E
EXAMPLE HAZARD STUDY QUESTIONS (BASED ON VALIDATING AUTOMATED MANUFACTURING[10])

- ***Configuration of Computer System***
  - Is the system implemented according to the intent of its specification?
  - Are there sufficient I/Os to enable plant/process to be operated as intended?
  - Are all sensors and equipment working correctly as designated?
  - What is automatic? What is manual?
  - Does the design consider expansion requirements?
  - Does the design consider the possibility of performance degradation from probable causes (e.g., disk full)?
  - What is the integrity of the power supply?
  - Will the system be affected by electrical interference or poor earthing?
  - Will the system be affected by ambient temperature or humidity?
  - Will the system be affected by dust, contamination, or corrosive materials?
  - Have precautions been made for fire, flood, vibration, and shock?
  - Have precautions been made for environmental needs and hygiene standards?
  - Is the system intrinsically safe?
  - To which mode do instruments fail?
  - How does the system know if equipment is faulty (instrument, computer, and manufacturing equipment)?
  - Is there any redundancy built in to cover equipment failures?
  - What is the system response to and recovery from utility failure?
  - How are these facilities controlled?
  - Are there hardwired trips and interlocks? Challenge them.
  - How are redundant and standby systems tested?
  - Are off-line test or bureau systems used as a source of spares?
  - How can the validity of these spares be assured?
  - Do suppliers have necessary spares, and equipment?
  - Can suppliers backup and recreate the present software configuration, and restore it?
  - Is the equipment still supported by its original company?
  - Are parts still available?
  - Are other computer systems connected to this system?
  - What happens if the connection between systems is lost?
  - How are these connections controlled?
  - Is the maximum length of communication lines exceeded?

- ***Software Control Scheme***
  - Has the software been reviewed, is it backed up, and can it be restored after a fault?
  - Is the software documented, with all documents kept in a safe place?
  - Will the system recover to a safe state after a power failure?
  - Examine all sequence charts.
  - Examine summary software flowcharts.
  - Examine database structure and content.
  - Challenge software functionality for incorrect user input, corrupted data, and an incorrect decision.
  - What consequences will follow an erroneous operation?
  - How are hazardous situations notified to operations staff?
  - Can software recover automatically, and has this been tested?

- ***GMP-Related Functions***
  - Who is responsible for directing operations?
  - Does supplied documentation adequately cover abnormal plant/process states?
  - Are there safe states for operation (startup, shutdown, holds, normal running, emergency shutdown, maintenance)?
  - What happens to the pharmaceutical product in the event of a failure?
  - Can the pharmaceutical product be recovered in the event of a failure?
  - When is it safe to resume operation?
  - Is there an operating procedure to cover recovery after partial failure?
  - Is there an operating procedure to cover recovery after full failure?
  - Is there a contingency plan?
  - Challenge all reasonable planned scenarios.
  - Is the contingency plan periodically reviewed and updated?
  - Has the contingency plan ever been tested?
  - Is there a copy of all system parameters and settings?
  - Can in-process changes to equipment operating parameters be made?
  - How many product rejects are needed to halt equipment operation?
  - Is there product reject verification?
  - How are batch records controlled?
  - Investigate alarms associated with all failure modes. Are they useful?
  - What happens when large numbers of alarms are raised together?
  - Is there a hard copy of valid alarm settings?
  - Can an interlock be left in an incorrect state?
  - Identify and challenge event logging and trending.
  - Are regular (annual?) service reviews conducted?
  - Are trends in equipment performance and failures monitored?

- ***Operation of Computer System***
  - Is the plant/process operating philosophy fully defined and understood?
  - Are all plant/process operation phases defined (startup, shutdown, abort, recovery)?
  - Are operating procedures in place for the computer system?
  - Are procedures updated and personnel retained after changes to plant/process?
  - Was appropriate training given to all personnel associated with the system's operation?
  - Are standards and recommended company practices followed?
  - How does the operator know what equipment is under computer system control at any moment?
  - Can the operator accidentally change key parameters?
  - Can the computer system tolerate invalid operator input?
  - How does the operator know that a production problem has occurred?
  - How does the operator know what to do (procedures and training)?
  - Has allowance been made for color-blind operators?
  - Is the accuracy of data displayed consistent with control needs?
  - Are standard names, colors, units, symbols, and abbreviations defined?
  - Are alarms and messages recorded in more than one location in case of a printer failure?
  - How do we know if any equipment has failed during operation?
  - Is a fault reporting procedure in place?
  - Are system reliability and failure rates regularly reviewed for trends?
  - How frequently are relief instruments used and tested?
  - Can equipment be operated from more than one location? Could this lead to errors?
  - Is a service agreement in place?

  - Are response times defined?
  - Is equipment maintained and calibrated regularly?
  - Are maintenance and changes controlled by a procedure?
  - How are changes between summer and winter times controlled?

- ***Security***
  - Do procedures for physical access exist?
  - Can software be modified without authorization?
  - What authorization is needed to modify software?
  - Have access levels been used to define/enable different levels of access?
  - Are passwords/key locks used to define/enable different levels of access?
  - Are all changes to software automatically recorded?
  - Can alarms be suppressed, and readings taken off-line or out-of-scan?
  - Are there special security arrangements for critical items?

9 Coding, Configuration, and Build

CONTENTS

Once the computer system has been designed, it can be built. The supplier generally bears all the responsibility for the activities associated with this phase; these cover Software Programming, Source Code Review, and System Assembly. These activities may not involve the pharmaceutical or healthcare company at all, depending on the nature of the relationship with the supplier. In such circumstances Supplier Audits may be used to verify that the supplier has the required capability maturity for the task, through having suitable controls in place. This, however, will not be possible for Commercial Off-The-Shelf (COTS) software and hardware of unknown pedigree. The acceptability of such products should have been determined as part of the Design Review.

SOFTWARE PROGRAMMING

Programming is the lowest abstraction for the software development process. Software for the new application may be constructed either by programming in a native programming language like C^{++} or XML*, line by line, or by assembling together previously programmed software components from software libraries or other sources such as COTS software. Native language programming might involve the use of assembly language or micro-code for real time or other time-critical operations. Source code is then *translated* (commonly referred to as *compiled*) for use on the target computer system.

GMP regulatory authorities expect software to be programmed and maintained under version and change control.[1] They also encourage the use of good programming practices.[2,3] Such programming practices should be defined, and cover the following:

- Software structure (file organization: including comments, headers, and trailers)
- Naming conventions (for folders and directories, file names, functions, and variables)
- Revision convention (configuration/change control, with audit trails as appropriate)
- Code format (style including indentation, labels, and white space)
- Controls on the complexity of code
- Ensuring that there is no redundant or dead code
- Role of compilers (configuration switches and optimization)

The quality of software delivered by the programming work directly determines the effort that will be required to maintain it. Convoluted software often reflects a history of change, perhaps involving the integration of bits and pieces of code from other systems. Such software is hard to follow at the source code level not only for the author-developer but, much more importantly, for those who will be obliged to maintain it throughout its useful life. One of the commonest causes of this is poor documentation. Once written, there is not a lot that can be done with such "spaghetti" software. In the short term it may not be practical to rewrite it, in which case the supporting documentation should be strengthened through a detailed Source Code Review. If the software is expected to have a longer lifetime, and especially if portions of the code are intended for future products, then rewriting the code altogether may make the most sense from a cost-effectiveness standpoint. This often turns out to be the case with large business systems such as MRP II.

The selected programming language can also have a significant impact on the effectiveness of a piece of software. Data-driven languages are appropriate to data-oriented solutions, while formula-driven languages such as FORTRAN and COBOL tend to lead to algorithm-oriented solutions. A poor choice of a programming language could well impose an inappropriate solution on the task to be accomplished and the consequential deficiencies will often be deep seated. It is very important, therefore, that the programming language truly complements the software design.

Examples of how software readability and quality can be improved by addressing some of the topics discussed above are given in Figure 9.1. The use of structured indentation and intuitively named functions and variables according to defined conventions aids understanding immediately. The use of comment perhaps as a preamble to a subroutine helps further. Producing code that is more understandable may take some extra effort in the first instance, but in the long run this is richly repaid in terms of reduced effort to identify and eliminate innate defects during development. Consequently, the testing is accelerated and the ongoing maintenance during operation less burdensome for users and developers alike.

Programming Approach

The choice of programming practices should be tailored to the particular characteristics of the programming languages in use, and be directed toward mitigating their known deficiencies. For instance, the main weaknesses associated with conventional ladder logic include:[4]

* XML stands for Extensible Markup Language, the universal format for structured documents and data on the World Wide Web.

```
procedure A(var x: w);
begin b(y, n1);
b(x, n2); m(w[x]); y:=x; r(p[x])
end;
```

(A) Unstructured Code Fragment

```
procedure change_window (var nw: window);
        begin  border(current_window, no_highlight);
               border(nw, highlight);
               move_cursor(w[n]);
               current_window:=nw;
               resume(process[nw])
         end;
```

(B) Structured Code Fragment

FIGURE 9.1 Structuring Code.

- Poor facilities for structured or hierarchical program decomposition
- Limited facilities for software reuse
- Poor facilities for addressing and manipulating data structures
- Limited facilities for building complex sequences
- Cumbersome facilities for arithmetic operations

Some programming languages are less forgiving to developers and more prone to errors than others. For instance, error-prone features of programming languages such as BASIC, C^{++}, and RTL/2 include the following:[5]

- GOTO, Assigned GOTO, and GOSUB
- Floating-point numbers
- Pointers
- Parallelism (concurrent processing)
- Recursion
- Interrupts
- Dynamic allocation of memory

Recent research indicates that about 5% of programming code will normally contain logical errors,[6] and that these are more likely to arise from the programmer rather than from the language used.[7] Good programming standards, rigorously enforced by code inspection, can greatly diminish this error rate. The best in-house programming standards are those based on well-tried industry standards such as IEC 1131-3 and ISA-S88. There is much help available in the literature and on the Internet in this regard, and there is no excuse nowadays for sloppy and careless programming in the absence of defined standards.

Redundant Code ("Dead Code")

Redundant or "dead" code is program logic that cannot possibly execute because the program paths never permit those instructions to be reached. It is not uncommon to find up to 20% of the code redundant in this way when code from an earlier version is reworked to achieve a slightly different functional purpose. GMP regulatory authorities have expressed concern that redundant code might be unintentionally accessed during system operation and recommend its removal.[8] Redundant code can include the following:

- Superseded code from earlier software versions
- Residual code from system modifications
- Unused features of standard software packages

Source code that has been deliberately commented must not be regarded as dead code since the compiler ignores it, and can never therefore become executable instructions.

Wherever possible, redundant code should be removed and the software recompiled. Care, however, must be taken to distinguish rarely used code from redundant code, and not to mistakenly classify the former as the latter! Examples of rarely used code include

- Apparently unused modules in large configurable systems
- Diagnostic features and test programs that are intended to remain dormant until needed

In instances where software instructions become dead code because of program modifications, these should be removed from the program before recompilation and submission to the production process.

No executable code should be resident in vacant areas of code storage media (as happens sometimes in the form of firmware chips) in those computer systems intended to operate in areas subject to electrical disturbances. Transient faults induced by EMI, ESD, and RFI can cause inadvertent jumps to take place to these storage locations. Unused areas of storage should be initialized to contain logic patterns that, if erroneously accessed by the processor, will cause the system to revert to a known and predefined safe state. All overlays of reserved storage should be populated with similar logic patterns. A number of hardware and software system-level techniques, including "error-capturing instructions" and "capability checking," are available and are suitable for implementation within a variety of design constraints.[9]

Compilers

Many compilers offer configurable options that enable programmers to improve the quality of their code in various ways, either at compile time or through run time checks. Examples of the types of errors that can be exposed in this way include string underflow or overflow, and checking of array boundaries and ranges. All such innate compiler checks should be enabled, unless it is disadvantageous to leave them switched on for the compilation of the final version. After all, some of these aids may retard the compilation speed to an unacceptable degree, and this is especially acute with very large programs.

One of the most error-prone areas of compiler operation is code optimization.[5] This facility manipulates the software's structure in an attempt to improve its overall functional efficiency. Incorrect optimization could have a devastating effect on the software's desired functionality. Therefore, we strongly recommend that this option be disabled for GMP-critical software.

Recent Inspection Findings

- The firm has failed to put in place programming standards for the numerous (>100) source code blocks that have been developed and maintained by company personnel. [FDA 483, 2001]
- There was inadequate software version control. [FDA Warning Letter, 1998]
- Source code blocks contain change control history annotations at the beginning of the code for change history information for each source code program. The firm failed to ensure that these change history annotations are updated when programming changes have been made. [FDA 483, 2001]

- The computer system lacked adequate text descriptions of programs. [FDA Warning Letter, 2001]
- Sections of code lacked annotations (e.g., the meaning of variables), and contained "dead" or unused code. [FDA Warning Letter, 1998]
- Validation materials failed to include printouts of source code with customized source code configurations. [FDA 483, 1999]
- QA had reviewed and initialed each programming script, but the procedure was not documented. [FDA Warning Letter, 1999]
- System design documentation including program code was not maintained or updated. [FDA Warning Letter, 2001]
- Following recognition of the [programming] problem, no formal documented training was provided to key personnel to prevent its recurrence, e.g., training to programmers, software engineers, and quality assurance personnel. [FDA Warning Letter, 1998]

SOURCE CODE REVIEW

No one doubts the crucial operational dependence of computer systems on their software, and the importance of professionally developed software is widely appreciated. GMP regulatory authorities hold pharmaceutical and healthcare companies accountable for the "suitability" of computer systems, including software,[10] and expect them to take "all reasonable steps to ensure that it [software] has been produced in accordance with a system of Quality Assurance."[11] One GMP regulatory authority is quoted as stating that "there is no room in the pharmaceutical industry for magic boxes."[12]

Comprehensive software testing, in other words testing that exercises all the pathways through the code, is not a practical proposition except for the very smallest programs. It implies testing every possible logical state that the system can ever assume. Software is often documented using flowcharts that track decision points and processing states. A relatively simple flowchart is given in Figure 9.2. Any path through the software associated with the flowchart is capable of triggering an error or failure. And it is not just the pathway that is important — data input and data manipulation will influence whether or not an error or failure state is generated. Barry Boehm calculated the number of conditional pathways through the flowchart to be 10^{21}.[13] Exception handling for error/failure conditions introduces further complexity, with interrupts creating a "jump" to what would otherwise be a wholly unrelated part of the system. Assuming one individual test could be defined, executed, and documented each second (a somewhat optimistic assumption in real life), it would take longer than the estimated age of the universe to complete the testing! Indeed, even if batches of a thousand individual tests could be conducted concurrently, the time required to complete overall testing would only be reduced to the estimated age of the universe. It is therefore evident that full functional testing of every pathway is never possible, and much software today is more complex than the example given in Figure 9.2. Other techniques are therefore needed to complement functional testing and measure the quality achieved in software development.

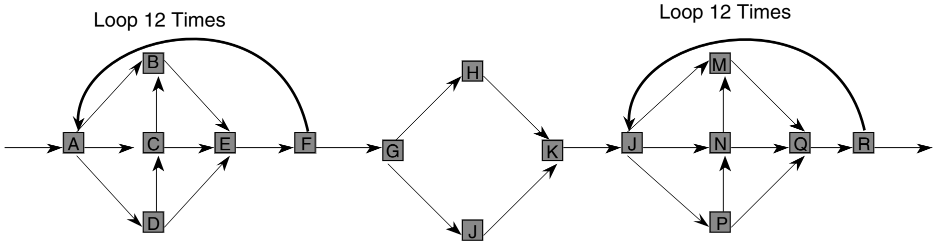


FIGURE 9.2 Practicalities of Comprehensive Testing.

Source Code Reviews (also known as Software Inspection) are a proven technique for improving software quality. These are intended to give a degree of assurance of the quality of code along the pathways that can never be functionally tested. We can use these, together with functional testing, to gain an overall measure of software quality.[14] It is astonishing that the limitations of functional testing are not widely appreciated. Many software companies and development teams blithely place complete reliance on functional testing as a measurement of quality without realizing the inadequacy of such measures. Quality must be *built into software* — it can *never* be solely tested in, nor can it be *measured by functional testing alone*. Pharmaceutical and healthcare companies must *not* rely on standard license agreements as mitigating the need for effective quality assurance systems, supervision including Source Code Reviews during Development, User Testing, and Supplier Audits. Most standard license agreements are nothing more than an abrogation of all responsibility by software developer organizations and can usually be succinctly summarized as "*As is, unsupported, and use at your own risk.*"

Review Criteria

Source Code Reviews have four basic objectives:

- Exposure of possible coding errors
- Determination of adherence to design specifications, including
  - Affirmation of process sequencing
  - I/O handling
  - Formulae and algorithms
  - Message and alarm handling
  - Configuration
- Determination of adherence to programming practices
  - (for example, headers, version control, change control)
- Identification of redundant and dead code

The GAMP Forum has responded to such concerns with a procedure for inspecting software embracing software design, adherence to coding standards, software logic, redundant code, and critical algorithms.[5] Source Code Reviews are particularly useful when verifying calculations that during the systems operation are being updated too quickly to check.

The Source Code Review must systematically cover all aspects of the software, with particular attention to GMP elements of functionality. The risk assessment process presented in Chapter 8 can be used to select which software will be subjected to the most detailed inspection (a threshold risk score will need to be set to determine when a detailed review is required). All configurations should be checked against specification.

Redundant bespoke (custom) programming is considered "dead" code and should be removed. The only exception is redundant code strategically introduced to try to protect the commercial confidentiality of proprietary software, usually by confusing disassemblers that might be used by unethical competitor organizations to reverse engineer the product.

COTS software functionality disabled by configuration is not redundant code in the truest sense, on the basis that the disabled software is intended to be enabled according to the need of a particular implementation. Examples of where functionality may be disabled without creating redundant code include library software (e.g., printer driver routines or statistical functions), built-in testing software, and embedded diagnostic instructions.

Annotated Software Listing

Listings of software subjected to detailed low-level walkthrough should be annotated with the reviewer's comments. Conventionally this has entailed handwritten comments on printed software listings, but there is no reason why electronic tools may not be used to support this activity assuming they are validated for this purpose.

Supplier Audits should always include the review of a sample of code to ensure compliance with quality system standards, though such a review will never pretend to assume the rigor of a formal Source Code Review. Where suppliers withhold software listings, access agreements should be established and refer to the availability of software and reference documentation in Chapter 11.

The process of reviewing source code typically consists of the following steps:

- Check adherence to programming practices (headers, version control, change control).
- Check I/O labels and other cross-reference information.
- Check any configuration setup values.
- Progressively check functional units for correct operation.
- Confirm overall process sequencing.

All critical I/O labels, cross-reference information, and configuration should be checked. Formulae and algorithms should be verified against design specification definitions. Where possible, manual confirmation of correct calculations should be undertaken for custom programmed formulae and algorithms. Message and alarm initiation and subsequent action should be traced to verify correct handling.

As the review progresses, a software listing should be marked up to record the findings made. Any deficiencies in the code should be clearly identified by an annotation. It is just as important to record if no deficiencies are identified. Figure 9.3 and Figure 9.4 provide some examples on how a software listing might be annotated. It should be recognized that the style of annotation would need to be adapted to fit different programming languages and structures.

Reporting

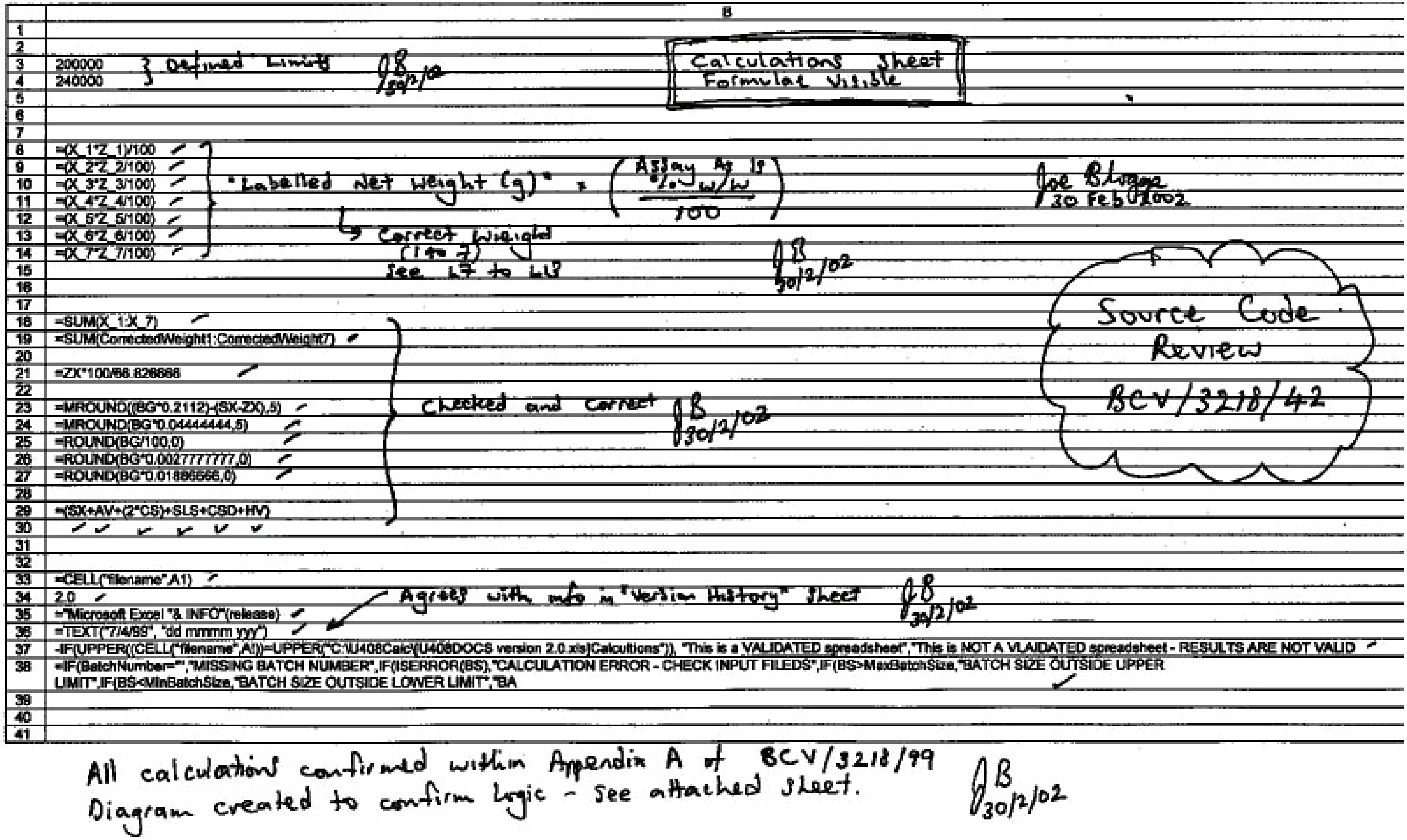
The outcome of the Source Code Review will be a report providing an overview of the review, together with a list of all observations noted and all actions that must be completed. Specific statements on software structure, programming practice, GMP-related functionality, information transfer with other portions of the system or other systems, error handling, redundant code, version control, and change control should be made before an overall conclusion on the suitability and maintainability of the software is drawn. A copy of annotated software listings should be retained with the report.

The report may identify some software modifications. How these modifications are to be followed through must be clearly defined. A major failing of many reports is the lack of follow-up of outstanding actions. Two classes of modification are defined here, for example:

Class A: Software change must be completed, software replaced, or supplementary controls introduced (e.g., procedural control or additional technical control) before the system can be released.

Class B: Software change does not have to be completed for the system to be released for use. These outstanding changes should be logged in the Validation Report, managed through change control, and subject to Periodic Review. It is important that these changes do not get overlooked.

Generally, widely distributed Off-The-Shelf (OTS) software is not considered to need Source Code Review if a reputable developer has produced it under an effective system of quality assurance and the product requires no more than an application parameter configuration.[15] In most computer systems, therefore, Source Code Reviews are limited to custom (bespoke) software and configuration within OTS software products.



	B
1	
2	
3	200000
4	240000
5	
6	
7	
8	=(X_1*Z_1)/100 ✓
9	=(X_2*Z_2/100) ✓
10	=(X_3*Z_3/100) ✓
11	=(X_4*Z_4/100) ✓
12	=(X_5*Z_5/100) ✓
13	=(X_6*Z_6/100) ✓
14	=(X_7*Z_7/100) ✓
15	
16	
17	
18	=SUM(X_1:X_7) ✓
19	=SUM(CorrectedWeight1:CorrectedWeight7) ✓
20	
21	=ZX*100/66.826666 ✓
22	
23	=MROUND((BG*0.2112)-(SX-ZX),5) ✓
24	=MROUND(BG*0.04444444,5) ✓
25	=ROUND(BG/100,0) ✓
26	=ROUND(BG*0.0027777777,0) ✓
27	=ROUND(BG*0.01886666,0) ✓
28	
29	=(SX+AV+(2*CS)+SLS+CSD+HV)
30	✓ ✓ ✓ ✓ ✓ ✓
31	
32	
33	=CELL("filename",A1) ✓
34	2.0 ✓
35	="Microsoft Excel "& INFO"(release) ✓
36	=TEXT("7/4/99", "dd mmmm yyy") ✓
37	-IF(UPPER((CELL("filename",A!))=UPPER("C:\U408Calc\[U408DOCS version 2.0.xls]Calcultions")), "This is a VALIDATED spreadsheet","This is NOT A VLAIDATED spreadsheet - RESULTS ARE NOT VALID ✓
38	=IF(BatchNumber="","MISSING BATCH NUMBER",IF(ISERROR(BS),"CALCULATION ERROR - CHECK INPUT FILEDS",IF(BS>MaxBatchSize,"BATCH SIZE OUTSIDE UPPER LIMIT",IF(BS<MinBatchSize,"BATCH SIZE OUTSIDE LOWER LIMIT","BA ✓
39	
40	
41	

} Defined Limits JB 30/2/02

Calculations Sheet
Formulae Visible

"Labelled Net Weight (g)" x (Assay As Is %w/w / 100)

↳ Correct Weight (1 to 7)
see L7 to L13 JB 30/2/02

Joe Bloggs
30 Feb 2002

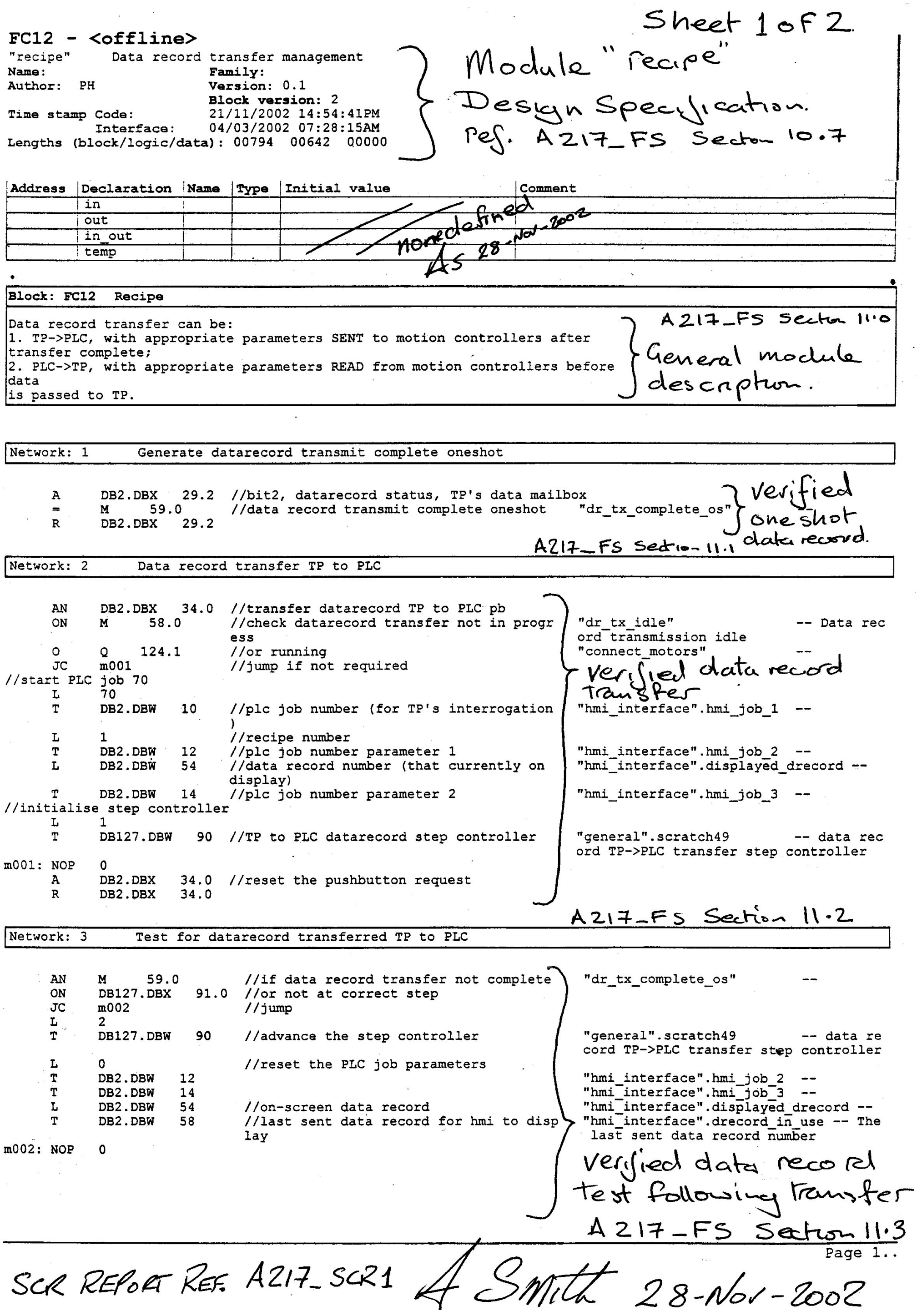
Source Code
Review
BCV/3218/42

Checked and Correct JB 30/2/02

Agrees with info in "Version History" sheet JB 30/2/02

All calculations confirmed within Appendix A of BCV/3218/99
Diagram created to confirm Logic - see attached sheet. JB 30/2/02

FIGURE 9.3 Example Annotated Software Listing.



Sheet 1 of 2

```
FC12 - <offline>
"recipe"     Data record transfer management
Name:                          Family:
Author:  PH                    Version: 0.1
                               Block version: 2
Time stamp Code:               21/11/2002 14:54:41PM
           Interface:          04/03/2002 07:28:15AM
Lengths (block/logic/data): 00794  00642  00000
```

Module "recipe" Design Specification. Ref. A217_FS Section 10.7

Address	Declaration	Name	Type	Initial value	Comment
	in				
	out				
	in_out				
	temp				

none defined AS 28-Nov-2002

```
Block: FC12   Recipe

Data record transfer can be:
1. TP->PLC, with appropriate parameters SENT to motion controllers after
transfer complete;
2. PLC->TP, with appropriate parameters READ from motion controllers before
data
is passed to TP.
```

A217_FS Section 11.0 General module description.

```
Network: 1      Generate datarecord transmit complete oneshot

     A      DB2.DBX   29.2  //bit2, datarecord status, TP's data mailbox
     =      M      59.0     //data record transmit complete oneshot    "dr_tx_complete_os"
     R      DB2.DBX   29.2
```

Verified one shot data record.

A217_FS Section 11.1

```
Network: 2      Data record transfer TP to PLC

     AN     DB2.DBX   34.0  //transfer datarecord TP to PLC pb
     ON     M      58.0     //check datarecord transfer not in progr   "dr_tx_idle"            -- Data rec
                            ess                                        ord transmission idle
     O      Q    124.1      //or running                               "connect_motors"        --
     JC     m001            //jump if not required
//start PLC job 70
     L      70
     T      DB2.DBW   10    //plc job number (for TP's interrogation   "hmi_interface".hmi_job_1  --
                            )
     L      1               //recipe number
     T      DB2.DBW   12    //plc job number parameter 1               "hmi_interface".hmi_job_2  --
     L      DB2.DBW   54    //data record number (that currently on    "hmi_interface".displayed_drecord --
                            display)
     T      DB2.DBW   14    //plc job number parameter 2               "hmi_interface".hmi_job_3  --
//initialise step controller
     L      1
     T      DB127.DBW   90  //TP to PLC datarecord step controller     "general".scratch49        -- data rec
                                                                       ord TP->PLC transfer step controller
m001: NOP   0
     A      DB2.DBX   34.0  //reset the pushbutton request
     R      DB2.DBX   34.0
```

Verified data record transfer

A217_FS Section 11.2

```
Network: 3      Test for datarecord transferred TP to PLC

     AN     M      59.0         //if data record transfer not complete  "dr_tx_complete_os"       --
     ON     DB127.DBX   91.0    //or not at correct step
     JC     m002                //jump
     L      2
     T      DB127.DBW   90      //advance the step controller           "general".scratch49        -- data re
                                                                        cord TP->PLC transfer step controller
     L      0                   //reset the PLC job parameters
     T      DB2.DBW   12                                                "hmi_interface".hmi_job_2  --
     T      DB2.DBW   14                                                "hmi_interface".hmi_job_3  --
     L      DB2.DBW   54        //on-screen data record                 "hmi_interface".displayed_drecord --
     T      DB2.DBW   58        //last sent data record for hmi to disp "hmi_interface".drecord_in_use -- The
                                lay                                      last sent data record number
m002: NOP   0
```

Verified data record test following transfer

A217_FS Section 11.3

Page 1..

SCR REPORT REF. A217_SCR1 A Smith 28-Nov-2002

FIGURE 9.4 Example Annotated Ladder Logic Listing.

Effectiveness

The effectiveness of Source Code Review is often questioned. Programmers alone should not inspect their own code because it is difficult to be objectively critical of one's own work. The objectivity of others and a willingness to accept criticism are key to any review process. Left to themselves, programmers' error detection rates on their own codes can be as low as 5%. Where a reviewer conducts the inspection in partnership with the software author, error detection rates can rise to 30% or more so long as it is not treated as a superficial, cursory activity. The time saved in taking corrective action on exposed errors, particularly the structural ones, in advance of testing usually more than justifies the involvement of a colleague.

Examples of real problems identified in Source Code Reviews include:

- Version and change control not implemented in so-called "industry standard" PLC and DCS software.
- Functions and procedures in MRP software not having a terminating statement so that the execution erroneously runs into the next routine.
- Incorrectly implemented calculations: moving averages in Supervisory Control and Data Acquisition (SCADA) systems, material mixing concentrations in DCS systems, flawed shelf-life date calculations in Laboratory Information Management Systems (LIMS).
- Duplicated error messages because cut-and-paste functions have been used carelessly.
- Interlocks and alarm signal inputs used by PLC software labeled as unused on electrical diagrams.

For a PLC or the configuration of a small Distributed Control System (DCS), the Source Code Reviews will typically require about 4 days' effort, split between an independent reviewer and the software programmer. The reward of identifying and correcting defects prior to Development Testing or User Qualification has proved time after time to more than compensate for the time and effort required to carry out the Review. It really is the most cost-effective way of building quality into software.

Access to Application Code

While rarely invoked, some GxP legislation requires reasonable regulator access to application-specific software, including any Source Code Review records.[14,16] For the purpose of regulatory GxP inspections, pharmaceutical and healthcare companies should therefore agree with their suppliers over possible access to application-specific software (say within 24 h). An example of the wording of such an agreement is given below:

> *[Supplier Name] hereby agrees to allow [Customer Name] or their representative, or a GxP Regulatory Authority access to view source code listings for [Product X] in hard copy and/or electronic format as requested. [Supplier Name] also agrees to provide technical assistance when requested to answer any questions raised during any such review. [Supplier Name] also agrees to store the original of each version of software supplied to [Customer Name] until it is replaced plus seven years. In the case of system retirement, the last version shall be stored to retirement plus seven years.*

GxP regulations require that access to the software and relevant associated documentation should be preserved for a number of years after the system or software has been retired (see Chapter 4 and Chapter 11 for more details). Software licenses do not entitle pharmaceutical and healthcare companies to ownership of the software products they have "purchased." All that has been purchased is a license, an official permission or legal right to use it for some period of time under defined conditions of use. Accordingly, some companies have established escrow (third party) accounts with suppliers to retain their access to software, but this is not mandatory. Access agreements directly with the software supplier for the purpose of regulatory inspections are an acceptable alternative. If the software supplier refuses to cooperate, this poses a dilemma. In such circumstances it is recommended that pharmaceutical and healthcare companies use other suppliers for future projects.[2,17]

RECENT INSPECTION FINDINGS

- Customized source code must be reviewed against requirements and the review results must be documented. [FDA 483, 1999]
- The firm did not review the software source code that operates the [computer system] to see if it met their user requirements before installation and operation. [FDA 483, 2001]
- It was confirmed the [software] listing was not reviewed or approved. [FDA 483, 2001]
- Validation materials failed to include documentation to establish that customized source code configurations had been reviewed. [FDA 483, 1999]
- There was no source (application) code review. [FDA 483, 2001]
- Configuration parameters must be reviewed against requirements and the review results must be documented. [FDA 483, 1999]
- There is no written procedure to describe the source (application) code review process that was performed for the XXXX computer system. [FDA 483, 2001]
- The firm has failed to perform a comprehensive review of all [software] to ensure appropriate programming standards have been followed. [FDA 483, 2001]
- Validation procedures governing source code reviews should avoid being guided by words such as “appropriate level” and “consistency.” [FDA EIR, 1999]
- Only a small fraction of each program’s code underwent detailed review. [FDA Warning Letter, 1998]
- To date only two of the 133 programs that comprise … have been subjected to code inspections under Standard Operating Procedures. Of these no defects were found in program … and multiple initializing problems were reported in … which is still undergoing review and code correction. [FDA Warning Letter, 1998]
- The selection of programs for code inspection under [Standard Operating Procedure] is not based on a statistical rationale. Your firm has implemented code inspections only on programs that are scheduled for code revisions for other reasons (enhancements …). [FDA Warning Letter, 1998]
- The firm failed to document review of source code blocks in … change control records. [FDA 483, 2001]
- No procedure for review of source code. No assurance that all lines of code and possibilities in source code are executed at least once. [FDA Warning Letter, 2002]

SYSTEM ASSEMBLY

Assembly should be conducted in accordance with procedures that recognize regulatory requirements and manufacturer’s recommendations. Any risk posed to pharmaceutical or healthcare processes by poor assembly must be minimized.[18] For instance, wiring and earthing practices must be safe.

Assembly should be conducted using preapproved procedures. The quality of assembly work, including software installation, should be monitored. Many organizations deploy visual inspection and diagnostic testing to confirm that the computer system’s hardware has been correctly assembled. Some companies tag assembled equipment that has passed such a quality check so that it can be easily identified.

Any assembly problems should be resolved before the system is released for Development Testing. If necessary, assembly procedures should be revised with any necessary corrections. Packaged computer systems do not need to be disassembled during Development Testing or User Qualification so long as assembled hardware units are sealed.

REFERENCES

1. FDA (1984), *CGMP Applicability to Hardware and Software*, Guide 11, Compliance Policy Guides, Computerized Drug Processing, 7132a, Guide 11, Food and Drug Administration, Center for Drug Evaluation and Research, Rockville, MD.
2. Trill, A.J. (1993), Computerised Systems and GMP — A U.K. Perspective, Part 1: Background, Standards and Methods; Part 2: Inspection Findings; Part 3: Best Practices and Topical Issues, *Pharmaceutical Technology International*, 5 (2): 12–26, 5 (3): 49–63, 5 (5): 17–30.
3. FDA (1987), *Software Development Activities*, Technical Report, Reference Materials, and Training Aids for Investigators, Food and Drug Administration, Center for Drug Evaluation and Research, Rockville, MD.
4. Lewis, R.W. (1995), *Programming Industrial Control Systems Using IEC 1131-3*, Control Engineering Series 50, Institution of Electrical Engineers, London.
5. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), International Society for Pharmaceutical Engineering (www.ispe.org).
6. Panko, R.R. (1998), What We Know about Spreadsheet Errors, *Journal of End User Computing*, 10 (2): 15–21.
7. Hatton, L. (1997), Unexpected (and Sometimes Unpleasant) Lessons from Data in Real Software Systems, *Safety and Reliability of Software Based Systems,* Springer-Verlag, Heidelberg, Germany.
8. Leveson, N. (1995), *Safeware: System Safety and Computers*, Addison-Wesley, Reading, MA.
9. Wingate, G.A.S. (1997), *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice*, Interpharm Press, Buffalo Grove, IL.
10. FDA (1985), *Vendor Responsibility*, Guide 12, Compliance Policy Guides, Computerised Drug Processing, 7132a, Guide 12, Food and Drug Administration, Center for Drug Evaluation and Research, Rockville, MD.
11. European Union (1993), *Annex 11 — Computerised Systems,* European Union Guide to Directive 91/356/EEC.
12. Fry, C.B. (1992), What We See That Makes Us Nervous, Guest Editorial, *Pharmaceutical Technology*, May/June: 10–11.
13. Boehm, B. (1970), *Some Information Processing Implications of Air Force Missions 1970–1980*, The Rand Corporation, Santa Monica, CA.
14. FDA (1987), *Source Code for Process Control Application Programs*, Compliance Policy Guides, Computerized Drug Processing, 7132a, Guide 15, Food and Drug Administration, Center for Drug Evaluation and Research, Rockville, MD.
15. Chapman, K. (1991), A History of Validation in the United States: Parts 1 and 2 — Validation of Computer-Related Systems, *Pharmaceutical Technology*, 15 (October): 82–96, 15 (November): 54–70.
16. U.S. Federal Food, Drug, and Cosmetic Act, Section 704, Factory Inspection.
17. Tetzlaff, R.F. (1992), GMP Documentation Requirements for Automated Systems: Parts 1, 2 and 3, *Pharmaceutical Technology*, 16 (3): 112–124, 16 (4): 60–72, 16 (5): 70–82.
18. European Union Guide to Directive 91/356/EEC (1991), *European Commission Directive Laying Down the Principles of Good Manufacturing Practice for Medicinal Products for Human Use*.
19. ACDM/PSI (1998), *Computer Systems Validation for Clinical Systems: A Practical Guide*, Version 1.1, December.

APPENDIX 9A CHECKLIST FOR SOFTWARE PRODUCTION, CONTROL, AND ISSUE[5]

Software Production

- Programming standards
- Command files
- Configuration control
- Change control
- Software structure

Software Structure

- Header
- Comments
- Named parameters
- Manageable module size
- No redundancy
- No dead development code
- Efficient algorithms

Software Headers

- Module/file name
- Constituent source file names
- Module version number
- Project name (and reference code/contract number)
- Customer company and application location
- Brief description of software
- Reference to command file
- Change history

Change History

- Change request number
- New version number
- Date of change
- Author of change
- Other source files affected

APPENDIX 9B
EXAMPLE PROGRAMMING STANDARDS[19]

Naming Conventions

Directories: It is recommended that files are stored in an organized folder structure relating to software architecture and or functions.

Index: An index file should be maintained (e.g., INDEX.TXT) for each directory. The index file should contain a list of all the programs/files in that directory with a short description of their contents/function.

File Names: File names should be descriptive and reflect the functions or content of the file. They should only contain alphanumeric characters (possibly plus the underscore character), and should always start with a letter rather than a number.

Extensions: For operating systems that support file extensions, a standard file extension naming convention should be used, e.g.,

- *filename*.DAT — ASCII data file
- *filename*.LOG — SAS log file
- *filename*.SAS — SAS program
- *filename*.SQL — SQL program
- *filename*.TXT — ASCII text file

Variables: Variable names should be intuitive and thereby reflect the contents of the variable. If it is difficult to select a relevant name, then a descriptive label should be used. Names made up of purely numeric characters should be avoided.

Program Documentation

All programs and subroutines should include documentation that provides a preamble to the source code in the form of a header comment block. The following information should be included:

Program Name: Name of program.

Platform: DOS, UNIX, VAX, Windows, etc.

Version: Version of software (e.g., 6.12 of the SAS package).

Author(s): Name of the programmer(s) and their affiliation.

Date: Program creation date.

Purpose: A description of what the program does and why it exists.

Parameters: Description of the parameters received from (input) or passed back to (output) the calling program.

Data Files: List any data sources for the program (e.g., ASCII files, ORACLE tables, permanent SAS data sets, etc.).

Programs Called: List any program calls that may be made external to the program.

Output: List any output files generated by the program.

Assumptions: List any assumptions upon which the program relies.

Restrictions: Describe any program restrictions.

Invocation: Describe how the program's execution is initiated.

Change History: This contains change control information for all modifications made to the program. Information should include date of change, name of programmer making modification, an outline description of the modification, and reason for the change. Some of this information need not be detailed if contained on references change control records.

Program code should be annotated with comments to reinforce the understanding of the code structure and its function. There should be at least one comment per main step, new idea, or use of an algorithm within the program. When a step or algorithm is complex, further comments should be added as appropriate through that section of code. Too much commenting should be avoided as this could hinder rather than aid an understanding of the code.

Program Layout

- Each source code statement should appear on a separate line.
- A blank line should be left between each logical section in the source code to aid readability.
- Blocks of source code statements representing nested routines should be indented so that these routines can be more easily identified. For example,

```
IF xxxx THEN DO

  statement;

  statement;

END;

ELSE DO

  statement;

  statement;

END;
```

- All variables should be declared and initialized at the beginning of the program. Default data types should not be used.
- All nonexecutable statements (e.g., variable declarations) should be grouped together in a block preferably at the beginning of the program.
- Complex mathematical expressions should be simplified by separating terms with spaces, or by breaking down the complex expression into a number of simpler expressions.
- Conditional branching structures should always bear a default clause to cater for situations outside the programmer's conception. This clause should cause the program to terminate gracefully. In this way the unexpected termination of the program in an undefined state can be engineered out and avoided.

General Practices

- It is good practice to arrange code into small reuseable modules. Once such modules have been validated, their reuse should be encouraged to improve quality and to reduce future validation efforts.
- Possible program input and execution errors should be predicted in advance and handled appropriately in the source code (e.g., division by zero).
- Avoidance of undesirable practices is also important to ensure the program does not process data in unexpected ways under unexpected conditions. Examples of bad practices to avoid include:
  - Commented-out code in final versions of programs
  - Hard-coded data changes in nonconversion programs

- Data processing sequences that vary and are difficult to repeat
- Bad practice examples carry much more weight as a teaching aid than good practice ones

Output Labeling

Output should be labeled with:

- The identity of the source program creating it, including version number
- The date and time generated
- The identity of the user
- The page number and total number of pages

APPENDIX 9C
EXAMPLE CHECKLIST FOR SOURCE CODE REVIEWS*[5]

Software Reviews

- Review formal issue of software
- Agreed and specified review participants
- Arrange review meeting
- Adequate prereview preparation time
- Conduct review
- Accurate and traceable review minutes
- Systematic coverage of software
  - Software design
  - Adherence to coding standards
  - Software logic
  - Redundant code
  - Critical algorithms
  - Alarms handling
  - Input/output interfaces
  - Data handling
- Agree corrective handling
- Assign corrective actions and completion dates
- Retain original reviewed software
  - Listings
  - Flow diagrams
- Incorporate changes
- Approve changes
- Issue software
- Retain review evidence

Review Follow-Up

- Ensure successful closure of review
- Escalate if required

* Regulatory authorities consider software a document and expect it to be treated as such within the quality system supervising its creation.

10 Development Testing

CONTENTS

Development Testing is the responsibility of the supplier. It includes establishing the Test Strategy, conducting Unit and Integration Testing, and conducting System Testing in preparation for User Qualification. Some organizations refer to System Testing as Factory Acceptance Testing. Development Testing is based on verifying the computer system's specification and design and

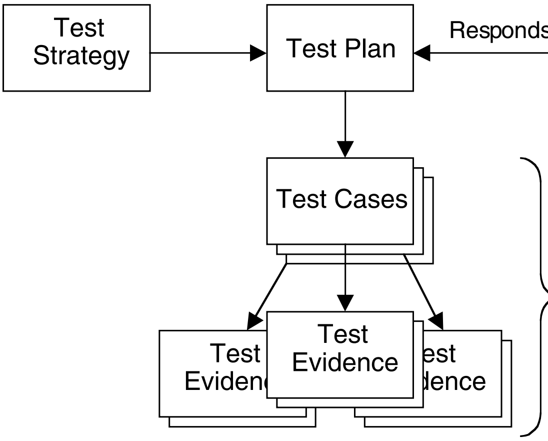


FIGURE 10.1 Testing Philosophy.

development documentation within the practical constraints of being at the supplier's premises. Comprehensive user testing is not usually possible under these circumstances.

Evidence of effective Development Testing can reduce the amount of subsequent User Qualification expected by GxP regulatory authorities. The pharmaceutical or healthcare company will often endeavor to include in its User Qualification as many tests as possible from Development Testing. It should also reduce the time needed to commission the computer system on the pharmaceutical or healthcare company's site, as qualification can focus on confirming an already established operational capability.

The supplier will normally invite the pharmaceutical or healthcare company to observe its own testing as part of a Predelivery Inspection. This is particularly important if the pharmaceutical or healthcare company is reducing the planned User Qualification based on the expectation of successful Development Testing. Many pharmaceutical and healthcare companies use Predelivery Inspection as an opportunity for informal operator training prior to the computer system's arrival on site. If specific training is required for User Qualification or the ongoing operation of the computer system, formal training is needed, and this should be documented in personnel training records.

TESTING STRATEGY

Testing must be carried out according to preapproved Test Plans and Test Specifications, and Test Reports prepared to collate the evidence of testing (i.e., raw data) as illustrated in Figure 10.1. Test Reports should be written to conclude each phase of testing and to authorize any subsequent phases of testing. Progression from one test phase to another should not occur without satisfactory resolution of any adverse test results.

TEST PLANS

Testing must include, but not necessarily be limited to, the activities listed below under the topics of Development Testing and User Qualification. However, the use of these qualification names is not compulsory. Due account must be taken of any test requirements identified by the Validation Plan, Supplier Audit, and Design Review. Testing must not be conducted against an unapproved specification.

Test Plans are used to define and justify the extent and approach to testing. Groups or individual test cases are identified together with any interdependencies. Test Plans may be embedded within Validation Plans, combined with Test Cases (to form what is commonly known as a test specification), or allowed to exist as separate documents. Test Plans must be reviewed and approved before the testing process they define begins. Test Plans and Test Cases are often referred to as protocols when applied to User Qualification.

TEST SPECIFICATIONS

Test Specifications collate a number of individual test cases. The value of preparing effective test cases should not be underestimated. Poor test cases will lead to a weaker measure of product quality than is possible from the activity and to an inconclusive overall result. These in turn will lead to delays while the uncertainty is considered; problem resolutions are determined and documented, usually with revised test specifications and repeated testing.

The level of detail required in Test Cases tends to vary considerably. Pharmaceutical or healthcare companies that want to use Development Testing to justify a reduction in the amount of User Qualification should review the test specifications as early as possible. Test instructions down to a keystroke level are not necessary if testers are trained and made familiar with the systems being tested. Any assumptions made regarding the capability and training of testers need to be documented in test specifications and supporting training records maintained.

The expected contents of individual test cases are described below.

Project Title/System Name

- Project number and system name to be defined in preapproved test specifications.
- Major systems should not use duplicate names.

Test Reference

- Unique test reference should be defined for each preapproved Test Case.
- Unique run number should be assigned during testing.
- Default run number should indicate the first test run unless retesting is done or a particular test requires multiple runs of the Test Cases.

Test Purpose

- Described a clear objective for each Test Case in the preapproved Test Specification.

Reference Documents and Test Prerequisites

- Test Case should carry a cross-reference to the part of the system specification that is being tested.
- Any prerequisites such as test equipment, calibration, test data, reference SOPs, user manuals, training, and sequences between different test scripts should be defined in the preapproved Test Specifications.

Test Method

- Define step-by-step test method.
- Identify data to be input for each step.
- Specify any screen dumps, reports, or observations to be collected as evidence at appropriate steps.
- Define associated acceptance criteria for individual steps as appropriate.
- Test Cases must not introduce new system specifications.

Test Results

- Register test case deviations in Project Compliance Issue Log.
- Cross-reference any Project Compliance Issues in test results.
- Confirm whether acceptance criteria for test method steps are met.

Test Outcome and Approval

- Define acceptance criteria for an overall successful Test Outcome.
- Annotate test outcome as appropriate during text execution.
- Insert signature after test execution to assign Test Outcome.
- Insert signature after test execution to confirm Test Outcome, noting confirmation as witness or review of test results.
- Name of signer and date of signature must accompany signatures.

Test specifications must be reviewed and approved before the testing they define begins. Test Cases can be written in such a way that test results are recorded directly on to an authorized copy of the test specification. Table 10.1 outlines an example Test Case.

Test Traceability

The Requirements Traceability Matrix (RTM) initially developed for the Design Review should be extended to track which tests cover which aspects of the specification.

Test Conditions

There are three basic types of testing: coverage testing, error-based testing, and fault-based testing. Tests should aim to expose errors rather than try to prove that they do not exist (we have seen in the previous chapter that proving errors do not exist is impossible). Testing must not be treated as debugging or snagging.

Coverage-Based Testing, as its name suggests, is concerned with establishing that all necessary aspects of the computer systems specification and design have been tested. As a general principle, all calls to routines, functions, and procedures should be exercised at least once during testing. In addition, all decision branches should also be exercised at least once. The use of an RTM can prove invaluable here, not only as a tool to identify tests but also to demonstrate afterward what coverage was achieved. Other useful tools include call trees.

Error-Based Testing focuses on error-prone test scenarios. It has been suggested that perhaps more than half of the functional tests conducted on a computer system should challenge its operational capabilities. Such testing includes:

- Boundary Values *(Guidewords: Minimum, Zero, Maximum)*
  Many problems arise when the design fails to take account of processing boundaries, such as data entry, maximum storage requirements, and maximum variables scanned at the highest scan frequency.
- Invalid Arguments *(Guidewords: Alphanumeric, Integer, Decimal)*
  Includes operator data entry, acknowledgments, state changes, open circuit instruments, instruments out of range, and instruments off-scan.
- Special Values *(Guidewords: Null-Entry, Function Keys)*
  Includes totally unexpected operator input and checking for undocumented function key shortcuts.

TABLE 10.1
Example Test Script

Project Title/System Name: *UV-Visible Chromatography System*		
Test Reference: *CS_TEST_04* **Run Number:** *01*	**Test Prerequisites:** *Test Reference CS_TEST_01 ("Log-On") has been successfully conducted*	**Reference Documents:** *User Manual CS/01* *Functional Specification CDS_N2_01*
Test Purpose: *Verify creation, operation, and reporting of an analytical method that performs spectral analysis of samples*		
Test Method:	**Acceptance Criteria (Expected Results):**	**Actual Results:**
Step 1: *Put ChemStation into "Advanced" mode. Load test assay method (select "File," select "Load Method," select "test_assay.m" from "\TEST\METHOD" directory on the test server, select "OK").*	Step 1: *None for setup*	Step 1: *Not applicable for setup*
Step 2: *Select "Instrument," select "Setup," select "Spectrophotometer." Enter following parameters: wavelength from "190" to "1100," integration time "0.5," all other values are left as default input.*	Step 2: *None for setup*	Step 2: *Not applicable for setup*
Step 3: *Load "Test Sample CSS05."*	Step 3: *None for setup*	Step 3: *Not applicable for setup*
Step 4: *Select "Run Sample." Print screen dump, initial/date, label and retain as evidence for this test.*	Step 4: *Result identifies sample material as hydrochloric sulfide*	Step 4: *Confirm UV result here*
Step 5: *Select "Close Run," select "Exit"*	Step 5: *None for shutdown*	Step 5: *Not applicable for shutdown*
Test Outcome (circle choice): PASS/REFER/FAIL		**Project Compliance Issues:**
Name of Tester: **Signature & Date:**	**Name of Checker:** **Signature & Date:**	

- Calculation Accuracy *(Guidewords: Precision, Exceptions)*
  Includes precision to a number of decimal places, underflow and overflow, division by zero, and other calculation exceptions.
- Performance *(Guidewords: Sequence, Timing, Volume of Data)*
  Includes execution of algorithms, task scheduling, system load, performance of simultaneous operations, data throughput, I/O scanning, and data refresh.
- Security and Access *(Guidewords: User Categories, Passwords)*
  Includes access controls for normal and privileged users, multiuser locking, and other security requirements.
- Error Handling and Recovery *(Guidewords: Messages, Alarms)*
  Includes software, hardware, and communication failure. Logging facilities are also included.

Fault-Based Testing focuses on the ability of tests to detect faults. This approach may artificially seed a number of faults in the software and then require the overall testing regime to reveal at least 95% of them. Seeding must be conducted without reference to existing test specifications. Validation practitioners do not commonly adopt fault-based testing although it provides a useful measure on how effectively testing has been conducted.

Test Execution and Test Evidence

Independence in testing is essential. No one can be relied upon to be wholly objective about his or her own work, and this is especially true in the highly creative activity of software development. Personnel who designed or developed the computer system under test should not conduct testing.

The collection of test evidence should concentrate on the main object of each Test Case. No test evidence should be mandated without good reason. In general it is not necessary to collect test evidence to demonstrate correct data entry or command keystrokes. Setup configuration should be defined in the Test Specification rather than treated as test evidence. Files used to support testing need to be archived.

Test evidence may be collated separately or attached to Test Cases. The GAMP Guide provides templates for collecting test evidence separately.[1] Table 10.1 provides an example of a Test Case that must be approved prior to testing but which can then be used to directly record testing. Whichever approach is used, a cross-reference should be made to and from separate test evidence and the Test Case it supports.

All raw data collected as test evidence should be initialed and dated. Observations made as test evidence should be documented as they occur with timings in addition to dates when appropriate. Supporting hard copy printouts, screen dumps, logs, photographs, certificates, charts, annotated drawings and listings, and reference documents must be identified with the tester's initials and dated at the time the evidence was produced. The use of ticks, crosses, "OK," or other abbreviations to indicate that actual results satisfied expected results should be avoided unless their meanings are specifically defined in the context of testing. It is better to faithfully record the actual results obtained.

Test Outcome

The outcome of each test is compared against acceptance criteria to ascertain whether the result fulfills the criteria without deviation. The concluding test outcomes are documented and approved as a "pass," "refer," or "fail."

- PASS — signifying that the Test Result meets the acceptance criteria as detailed in the test script in full without deviation from them in any way.

- REFER — signifying that the test result is ambiguous in that a deviation has occurred but the test still potentially fulfills the intent of the acceptance criteria. An example here might be a typographical Test Case error not affecting the integrity of testing. All referred test outcomes need to be registered in the Project Compliance Issue Log. Referred test outcomes must be resolved as either "pass" or "fail" before an overall conclusion can be drawn on the quality of the product being tested.
- FAIL — signifying that the Test Result does not fulfill the acceptance criteria.

Independent Checks

Test outcomes need independent verification for validated applications. There are two main ways to manage the checking of test evidence, and the meaning inferred from check signatures varies accordingly:

- ***Witness test results*** as they occur. This requires personnel to monitor testing as it progresses and sign test results/outcomes as each test is completed. This approach need only be used to document critical observations where physical test evidence such as printouts are not directly available from the computer system (e.g., audible alarm). The role of the witness can be restricted to GxP-related Test Cases.
- ***Review test results*** after testing is complete. This is often the cheaper and hence preferred method. It requires that sufficient evidence be collected to support the test result/outcome for each Test Case.

Independent checks should clearly document a review of corroborating evidence. It is this review that will give credence to the independent check if it were ever challenged during a regulatory inspection. Simply stating a PASS or FAIL test outcome without any test evidence is unlikely to satisfy regulatory inspection.

Test Failures

All test failures must be documented, reviewed, and analyzed to identify the origin of the failure. The person approving the Test Results must consider the consequences of failure on the significance of the Test Results already obtained. Single or multiple tests may be abandoned. If the analysis of a test failure results in an amendment to the Test Case, controlling specification, or software, then the relevant documentation must be amended and approved. Further testing requirements must be agreed upon in accordance with the relevant change control procedure. Retest of single, multiple, or all tests may be required. Deviations from the Test Case acceptance criteria, where there is no risk to GxP or safety, may be accepted with the approval of the User and QA. Such concessions must be recorded and justified in the Test Report. Managing deviations and the use of Project Compliance Issue Logs are discussed further in Chapter 4 and Chapter 6, respectively.

Test Reporting

The results of testing should be summarized in a Test Report that states:

- System identification (program, version configuration)
- Identification of Test Specifications
- Resolution to referred test outcomes, with justification as appropriate
- The actions taken to resolve test failures, with justification as appropriate
- Overall determination on whether testing satisfies acceptance criteria

The Test Report must not exclude any test conducted including those repeated for failed tests. It may be combined in a single document with test results. A successful overall testing outcome authorizes the computer system for use. Test Reports are not necessarily prepared by QA; however, they should be approved by QA.

Managing Changes during Testing

Changes to the system will likely be required during testing to correct inherent software defects exposed by test failures. It is important for the developer to manage such changes under careful change control. Supplier and User organizations should not apply undue pressure on developers to make and release changes too quickly such that change control might be compromised. After all, it is very tempting for developers under the pressure of unexpected project delays to carelessly correct one defect and in the process create another in an apparently unconnected function. Thus, when changes are made, the design must be carefully considered and the requirements for the regressions testing the system derived directly from this understanding. Then, and only then, can regression testing demonstrate that the change has not inadvertently created defects in other parts of the system.

Test Environment

Test environments can be quite complex depending on the size of the application and the need to provide configuration management of version upgrades. Small applications such as spreadsheets may be developed, tested, and released from a single environment. Larger applications generally warrant a segregated if not separate test environment.

For very large applications there are typically three working environments, as illustrated in Figure 10.2: Development, Test, and Holding. Software development for new and modified code is conducted in a dedicated development environment. When the software is ready for testing it is moved to the testing environment for unit, integration, and system testing. The testing environment may be a different physical installation or a segregated area in the development environment. Either way, strict configuration management must be observed. Only when testing has been successfully completed can the software be moved into the holding area as master source code. The holding area needs to be a highly protected separate environment to which access is restricted to those with authority to release approved software versions. If testing has been unsuccessful, the software is returned to the development environment for revision before subsequently coming back to the holding area for repeated testing until a successful outcome is achieved and the software is ready for release.

Recent Inspection Findings

- Test inputs are not always documented.
- Expected results are not always defined.
- Two comparisons done … did not state whether or not the results were acceptable.
- The procedure states that the application "validates" if computer and manual results "are the same." There is no definition of "same" with acceptable variation specified.
- Unused XXXXXX printouts were routinely discarded with no explanation. [FDA Warning Letter, 2000]
- Test results often consist of check marks only.
- The inspection found that data in numerous records were altered, erased, not recorded, recorded in pencil, or covered in white out material. Therefore there is not a complete record of all data secured in the course of each test. [FDA Warning Letter, 2000]
- Test results were found reported in pencil on uncontrolled pages.

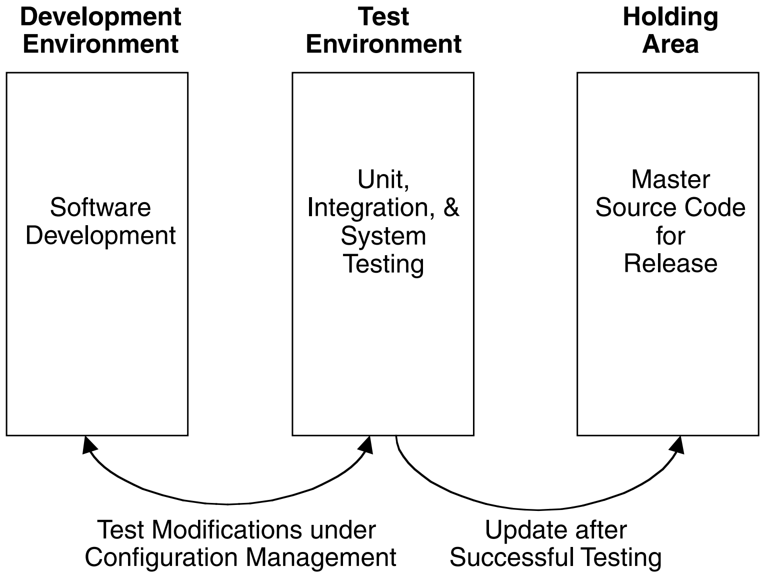

Development Environment
Test Environment
Holding Area
Software Development
Unit, Integration, & System Testing
Master Source Code for Release
Test Modifications under Configuration Management
Update after Successful Testing

FIGURE 10.2 Commercial Test Environment.

- Test documents included multiple sections of test forms, which were crossed out without initials, dates, or explanation.
- The procedure calls for the same individual who writes/revises the [software] program to validate the program.
- Test results lacked indication of review or approval.
- The test report generated from these activities lacked a document control number. [FDA 483, 2000]
- Firm failed to ensure that the supplier of the XXXX documented all the required test results to indicate the supplier's quality acceptance of the XXXX manufactured and delivered to your firm. [FDA Warning Letter, 2002]

UNIT AND INTEGRATION TESTING

Unit Testing (also known as *module testing*) is often done concurrently with coding and configuration, as program components are completed. Unit Testing should be extensive but not necessarily exhaustive, the aim being to develop a high degree of confidence in the essential functionality of modules.

Unit Testing must be accompanied by Integration Testing. Integration Testing exercises the interfaces between components and typically ensures that subsystems that have been developed separately work together correctly. Testing should ensure a high coverage of internal control flow paths, error handling, and recovery procedures — paths that are difficult to test in the context of functional (or "black box") testing, as we have seen in the previous chapter.

STRUCTURAL (WHITE BOX) TESTING

Together, Unit and Integration Testing are often referred to as Structural (or "White Box") Testing. Tests exercise the components and subsystems of the design in isolation, using known inputs to generate actual outputs that are then compared to expected outputs (see Figure 10.3). Coverage-based, error-based, and fault-based testing should be applied as described earlier.

It is important that pharmaceutical and healthcare companies have confidence in the Structural Testing as well as in the Functional Testing of the computer system. One complements the other, and together provide the measure of quality of the overall system. Records of Unit Testing and

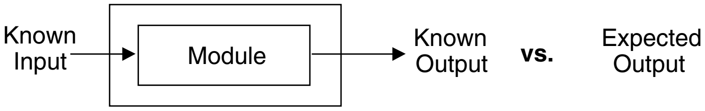


FIGURE 10.3 Structural "White Box" Testing.

Integration Testing (including test specifications and results) should be kept by the supplier and retained for inspection, if requested, by the pharmaceutical or healthcare company. Any test harnesses, emulations, and simulations used during testing must be specified and assurance in their capability demonstrated.

It is recommended that about 80% of the Development Testing effort be focused on Unit Testing and Integration Testing to establish the inherent structural correctness of the computer system. The remaining testing effort is applied to System Testing.

Acceptance Testing of COTS Software and Hardware

The System Developer should pay careful attention to the use of COTS software and associated necessary acceptance testing. Structural Testing is not required if sufficient confidence can be placed in Functional Testing. COTS products should be proven for use by commercial exposure and successful use in the marketplace so that further Structural Testing can be reckoned not to be required, as discussed in Chapter 8. Consequently, the following functional acceptance testing recommendations are made for COTS products (based on Jones et al.[2]):

- Test that the functions performed by the COTS software or hardware meet all specified requirements.
- The interfaces through which the user or other software invokes COTS functionality should be thoroughly tested.
- Test that all functions that are not required, and remain unused, cannot be invoked or do not adversely affect the required functions, for example, through erroneous inputs, interruptions, and misuse.
- Verify that all functions that are not required remain unused and those that are not access-protected do have procedural controls in place.
- All errors discovered and traced to a COTS product during testing must be reported to the vendor and the Design Review revisited as necessary.

In addition, Software Of Unknown Pedigree (SOUP) will require fault-based testing so that some indication of innate quality (albeit a very weak measure) can be derived. Fault-based testing will not always be possible for SOUP; this depends on access to its source code and the availability of supplementary design-related information such as user manuals. Unfortunately, the very nature of SOUP means a Supplier Audit, which is what is really needed in these circumstances, but is not possible. Where fault-based testing is not possible, the design may have to be modified to compensate for it. SOUP may have to be "wrapped" by other software that only allows valid data input. Alternatively, independent monitoring software may be implemented to identify any invalid SOUP operation. Wrapper software and independent monitoring software, of course, will require validation in their own right. These measures are a last resort and are far from desirable, but sometimes the lack of any viable alternative makes their adoption unavoidable.

Inspection Experience

Ian Johnson[3] recalls the instance of a PLC-controlled granulator that failed when challenged by operators deliberately entering inappropriate values for control parameters. Entry of zero for the

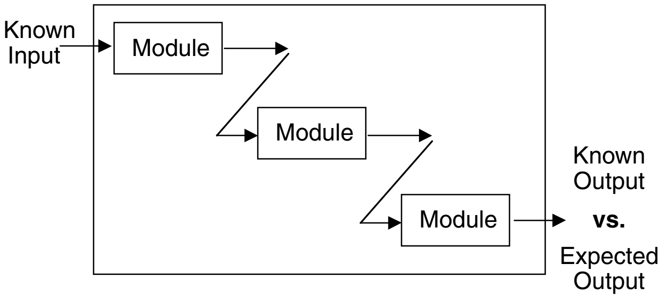


FIGURE 10.4 Functional "Black Box" Testing.

run duration or the stopping torque would cause the device to run indefinitely. Entry of zero revolutions per minute for the motor speed did not disable the motor as it should have done. Unfortunately, no memory was available to implement any warning messages or to provide some entry editing function or to reject an invalid value. As the granulator was entirely dependent on the PLC, the whole system was abandoned.

SYSTEM TESTING

System Testing is conducted by the supplier to verify that the computer system's intended and defined functionality has been achieved. Such Functional Testing is often referred to as "Black Box" Testing because it does not focus on the internal workings of a system (components and subsystems); rather, the focus is on the complete system as a single entity (see Figure 10.4).

System Testing by suppliers of COTS products is sometimes called *alpha testing* and is used as the basis for releasing a product to market. Some suppliers will also invoke *beta testing* whereby a selected band of trusted users is invited to evaluate COTS products before their general release. This is done with the full knowledge that inherent defects may well emerge, and the trusted users run that risk. In this way the supplier can verify the robustness of its products in the privacy of a smaller group of partners before it makes any necessary revisions prior to public exposure of the product in the wider market.

FUNCTIONAL (BLACK BOX) TESTING

Functional Testing is testing the system from a user's perspective — i.e., without knowledge of the internal architecture and structure of the system. Inventory checks are made by visual inspection, while functionality is verified by running the computer system. Test scenarios should include:

- Checking hardware components against equipment list
- Checking switch settings (e.g., interface card addressing)
- Checking any equipment calibration is calibrated as required
- Checking bespoke and COTS software versions loaded against configuration management plan
- Exercising inbuilt software diagnostic checks
- Verifying system operation against design intent
- Challenge testing against operating ranges (e.g., data entry and performance)
- Challenge testing security and access
- Verifying startup and shutdown routines
- Verifying data backup and recovery routines
- Verifying that communication interfaces are operating
- Verifying alarm and event status handling

Interface functionality is often tested using simulation utilities. This is to avoid the inconvenience of setting up associated equipment and instrumentation with the added burden of any calibration required. The use of simulators may entail additional validation requirements in regard to software tools as discussed in Chapter 5.

Tests not conducted as part of the System Testing must be included in User Qualification. Safety functions should include Functional Testing to ensure that the safety devices operate as intended in normal operating conditions and include exploring the consequences of a component failure as well as the effect this will have on the system. Calibration records must be kept to support User Qualification as required.[4,5]

Stress Testing

System Testing should include Stress Testing to verify that invalid conditions are managed in a controlled fashion and that these conditions do not lead to erroneous operation or catastrophic failure. There are basically two types of Stress Testing:

- Entering data outside the range of acceptability and ensuring that the data are flagged as erroneous.
- Burdening the system with an avalanche of transactions. The objective is to determine the maximum operational capacity at which the system can be run without danger of loss or corruption of data.

Automated testing tools can be used to great effect during System Testing and are discussed in detail in Chapter 5. The use of any automatic testing tools must be agreed upon with the pharmaceutical or healthcare company, preferably in the Supplier Project/Quality Plan.

Upgrade Compatibility

The upgrade path for superseded versions of the computer application also needs to be verified. Users expecting to upgrade existing applications should not experience problems. Upgrade tests should not be limited to functional testing but should also exercise data structures. An informed regression testing strategy needs to be employed.

Inspection Experience

Common testing problems observed by GMP regulatory authorities include the following:[3]

- Poor choice of test cases
- Failure to define the intent of tests
- Failure to document the test results

The success of Development Testing should be based on identifying and correcting deficiencies rather than on merely looking at the initial pass rate. After all, it is far more important to detect deficiencies now than be deluded into believing that the system is fully functional, only to be embarrassed later on during User Qualification or when the system is found to be noncompliant during live operation.

PREDELIVERY INSPECTION

Predelivery Inspections (PDI) are used to verify the system build against detailed hardware and software design, coding and configuration programming standards, hardware assembly practices and any relevant regulatory requirements, or industry guidance relating to these areas. Validation practitioners may be more familiar with the phrase "Midway Audit" in conjunction with GCP computer systems.[6]

Many pharmaceutical and healthcare companies find PDIs useful to help prompt their suppliers to ask for help and clarifications during Design and Development. Often suppliers have multiple concurrent projects, in which case work on individual projects tends to slip behind schedule and become rushed toward the end of the designated project timetable. Individual projects may need to be brought back on schedule and, if so, the pharmaceutical or healthcare company may be able to help by extending timescales, providing additional resources, or by clarifying requirements.

PDIs are based on visual assessments and are distinct from physical testing described earlier in this chapter. PDI typically covers observation/verification of the following (based on the *Baseline Pharmaceutical Engineering Guide*[7]):

- Drawings and layout diagrams
- Adoption of good programming practice
- Assembly checks as appropriate
- User interface functionality
- Unit, module, and integration test records

A PDI need not be a single event. In some situations the PDI may best be conducted in parts, examining various elements of a system as they are completed. The scheduling and scope of PDIs should be carefully considered to maximize their benefit.

It should be recognized that there will be situations, especially on smaller projects, where the cost of attending the PDI may outweigh the benefits and risks in terms of schedule. In these cases, the inspection can be postponed until delivery on-site; this is a business cost-benefit decision. An example where a single PDI might be appropriate on a large project is instructive. This might be where a project team is sequentially rolling out a number of similar applications, and a PDI on the first application may be all that is needed depending on the differences between similar applications. PDIs are also not appropriate for COTS products because by definition they are already released to market and so development and testing are complete.

Not many pharmaceutical and healthcare companies currently conduct PDIs, although the concept has been identified as good practice for some time. This is because PDIs are often hard to justify, especially when project budgets are tight; they are often considered as only desirable, not essential. Experience has shown, however, that they have proved very useful and effective, giving early warning of potential problems and helping to build a partnership with the suppliers. It is important to avoid situations where the supplier wants to release a system for delivery (for cash flow reasons), while the pharmaceutical or healthcare company is equally keen to accept delivery (and get on with the project). It is recommended that projects do not wait until the User Qualification stage to fix known problems that are more easily corrected before installation of the computer system at the pharmaceutical or healthcare company's site.

REFERENCES

1. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), International Society for Pharmaceutical Engineering (www.ispe.org).
2. Jones, C., Bloomfield, R.E., Froome, P.K.D., and Bishop, P.G. (2001), *Methods for Assessing the Safety Integrity of Safety-Related Software of Uncertain Pedigree (SOUP)*, U.K. Health and Safety Executive, Contract Research Report 337/2001.
3. Wingate, G.A.S. (1997), *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice*, Interpharm Press, Buffalo Grove, IL.

4. European Union (1993), European Union Guide to Directive 91/356/EEC.
5. U.S. Code of Federal Regulations Title 21, Part 211, *Current Good Manufacturing Practice for Finished Pharmaceuticals*.
6. Stokes, T. (2001), Validating Computer Systems, Part 4 — *Applied Clinical Trials*, 10(2).
7. ISPE (2001), *Baseline Pharmaceutical Engineering Guide: Qualification & Commissioning*, International Society of Pharmaceutical Engineering, Tampa, FL.

APPENDIX 10A
EXAMPLE TEST PLAN[1]

Introduction

Scope (Overview)

Test Plan

- Specific areas not tested
- Test procedure explanation
- Action in the event of failure
- Logical grouping of tests
- How to record test results

Test Requirements

- Personnel
- Hardware
- Software
- Test harness
- Test data sets
- Referenced documents

Test Procedure

- Unique test reference
- Cross-reference to specification
- Step-by-step method
- Expected results (acceptance criteria)

Test Results

- Raw data
- Retention of results
- Method of accepting completed tests

Glossary

References

Appendices

APPENDIX 10B
EXAMPLE TEST STRUCTURE[1]

Unique Reference

Objective

- Single sentence

Resources Requirements

- Specific to tests

Step-by-Step Procedure

- Repeatable procedure
- No unrecorded prerequisite requirements
  - Information
  - Experience

Acceptance Criteria

- Smart
  - Specific
  - Measurable
  - Achievable
  - Realistic
  - Timed

Testing Requirements

- Personnel
- Hardware
- Software
- Test harness
- Test data sets
- Referenced documents

Bottom Line Test Result

- Pass/fail outcome

Observations

- Additional information
- Acceptance concession

11 User Qualification and Authorization to Use

CONTENTS

The purpose of the User Qualification stage is to verify the operability of a computer system. Authorization to use the computer system after User Qualification is documented through a Validation Report. User Qualification is sometimes known as User Acceptance Testing (UAT), but differs from Development Testing in that it is performed under the supervision of the user organization. Development Testing does not require any user involvement, and indeed for Commercial Off-The-Shelf (COTS) systems users in general are seldom consulted. Care must be taken when planning computer systems not to unnecessarily duplicate Development Testing during User Qualification. It is often also prudent to involve future maintenance and support representatives within the User Qualification activities. User Qualification should abide by any prerequisites that are required, in readiness for operation and maintenance of the computer system.

QUALIFICATION

Qualification is the responsibility of the pharmaceutical and healthcare company, although suppliers often assist it. This phase consists of four sequential activities, as illustrated in Figure 11.1: Site Preparation, Installation Qualification (IQ), Operational Qualification (OQ), and Performance Qualification (PQ). IQ, OQ, and PQ should be applied to computer systems as indicated by key regulatory guidance.[1–3]

The relationship between qualification and system specifications is indicated in Figure 11.2 and Figure 11.3. Site Preparation ensures that the setup requirements for the computer system are complete; IQ verifies the installation, configuration, and calibration of delivered equipment to the Software and Hardware Design; OQ verifies the operational capability to the system specification; and PQ verifies the robust and dependable operation of the computer system. The inclusion or exclusion of tests between these qualification activities is usually based on convenience.

The use of the term "qualification" terminology may sometimes confuse those who are familiar with established process/equipment/facility validation practices. As the FDA has conceded, there is no consensus on the use of testing terminology, especially for user site testing.[4] For the purposes of this book, the term "qualification" is used to embrace any user testing that is conducted outside the developer's controlled environment. This testing should take place at the user's site with the actual hardware and software that will be part of the installed system configuration. Testing is accomplished through either actual or simulated use of the software being tested, within the context in which it is intended to function.

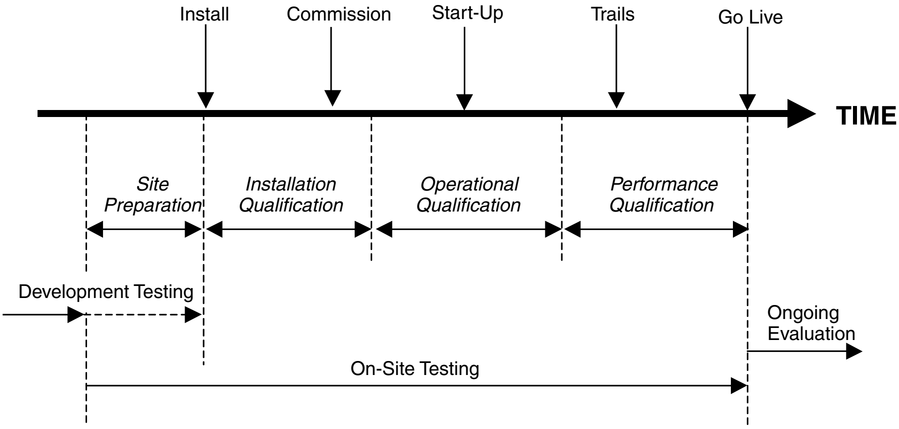


FIGURE 11.1 Qualification Time Line.

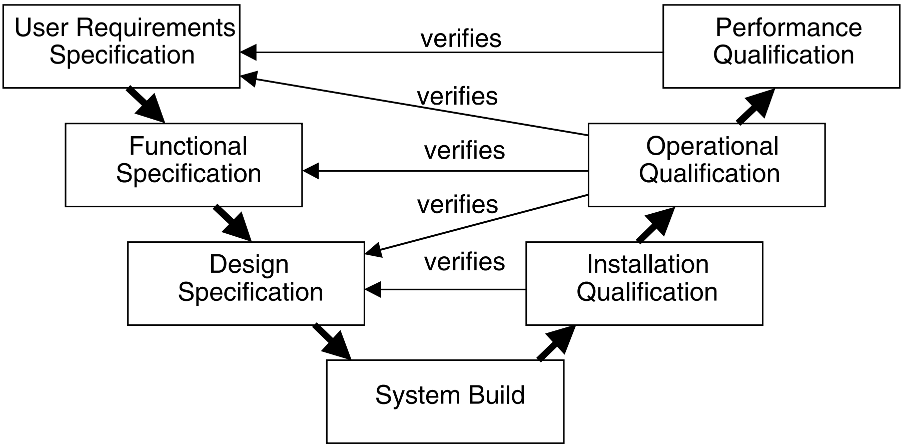

User Requirements Specification
Performance Qualification
verifies
verifies
Functional Specification
verifies
Operational Qualification
verifies
Design Specification
verifies
Installation Qualification
System Build

FIGURE 11.2 Verifying System Specifications.

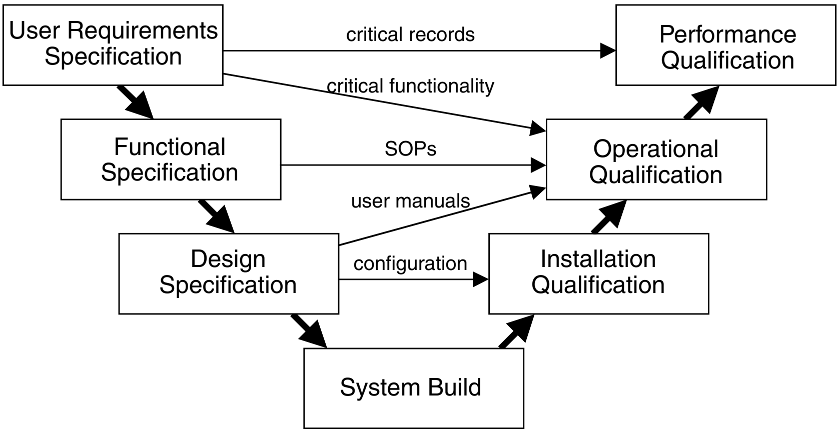


FIGURE 11.3 Supporting Test Requirements.

Test Documentation

Qualification should follow the same principles that were outlined for the computer system's Development Testing and discussed in Chapter 10. Test specifications (also known as qualification protocols) must be written, reviewed, and approved before testing begins. It is especially important that the qualification meets the so-called *S.M.A.R.T.* criteria:[5]

S*pecific:* test objectives address documented requirements.
M*easurable:* test acceptance criteria are objective, not subjective.
A*chievable:* test acceptance criteria are realistic.
R*ecorded:* test outcome evidence is signed off and, where available, raw data is attached.
T*raceable:* test records, including subsequent actions, can be traced to defined system functional requirements (it does what it is supposed to do).

Many consultancy firms offer pharmaceutical and healthcare companies access to their standard qualification protocols for a fee. However, such test specifications should be adapted to reflect the specific build configuration of the system being tested.

Test specifications in theory can be written during system development. In practice, however, while they may be drafted during development, they often need details confirmed with information that is only available after system development is complete.

User Qualification can begin once test specifications have been approved. Figure 11.4 outlines the test management process. For a specific function to be tested, it is necessary to have a test method and a known system build configuration. Test results should be recorded for all test methods executed. The outcome of the test should satisfy predefined acceptance criteria, in which case the testing may proceed to the next test. All test failures must be recorded and their cause diagnosed beyond doubt. It may be necessary to abandon that test, but this does not necessarily mean that the overall testing activity has to cease there and then. Where testing continues after the failure of an individual test, a rationale should be prepared and approved to record the justification for this decision to proceed. Examples of where there may be a clear justification to proceed include a limited hardware failure, or isolated software failures, or even a software failure with a limited impact. In some instances the software itself may be defect-free in that respect, but the apparent failure is actually due to an incorrect test execution process, in which case the test may be repeated. In other instances the individual test may be abandoned, but the overall testing continues with the next logical test. Where tests are repeated, for whatever reason, the original test results should be retained as well as the retest results. The most important factor throughout is *never* to ignore a test failure that *could* point to a fundamental design flaw. Not to do so is to deceive oneself, and such action is bound to end in tears. Such failures must be explored to allay suspicion before much other testing ensues.

Test failures will normally require a root cause fix. Some tests might fail on a cosmetic technicality such as an incidental typographic error. In this situation the necessary amendments can be marked up on an existing copy of the test method, taking care not to obscure the original text. The reason for making the amendment and the person effecting it should be clearly identified, together with the time the amendment was made. Other tests might trigger a failure because a test method is clearly in error. In these situations, it may be acceptable to annotate a fresh clean copy of the test method and rerun the test. Again, the reason for making the amendment and person effecting it should be clearly identified, together with the time the amendment was made.

Hopefully, most tests will uncover technical system deficiencies rather than test method inaccuracies. Technical deficiencies should be corrected and system documentation updated to reflect any change made. It may be appropriate to increment the system build version under configuration management. New test methods may need to be prepared to test any changes made. If a new system

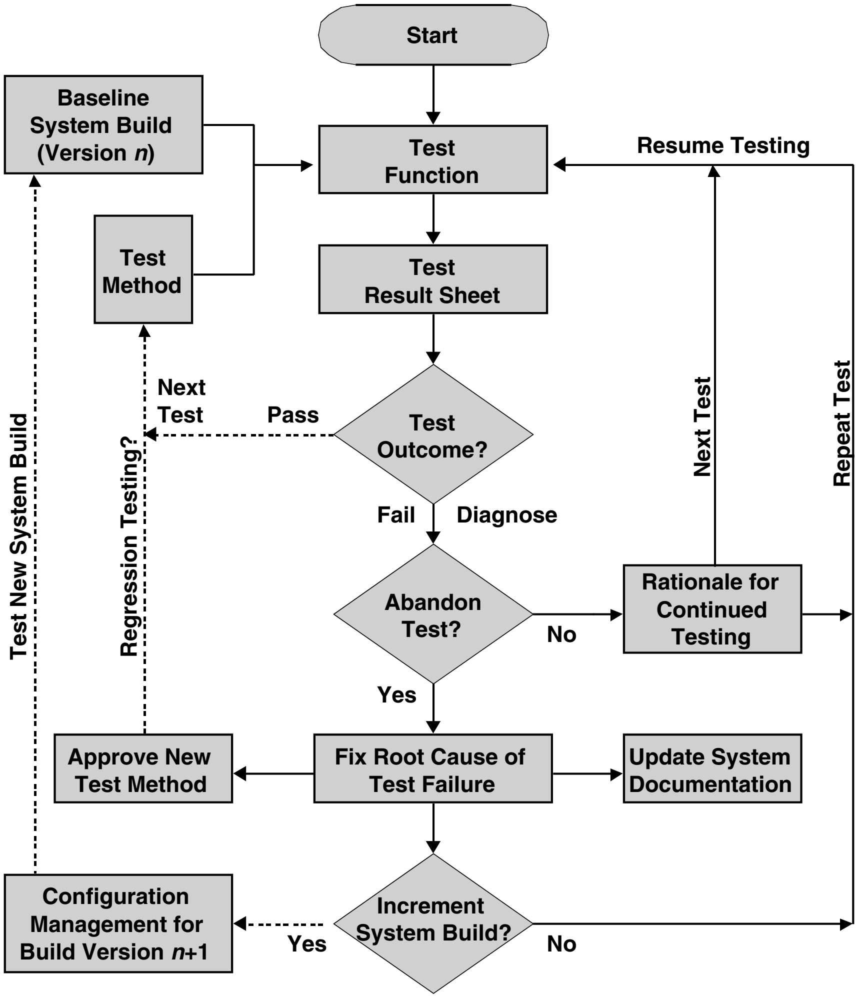


FIGURE 11.4 Test Management.

build is created, then overall testing should be reviewed to determine where a comprehensive retest is required or whether relevant regression testing will be sufficient.

A test report should be prepared to complete each qualification activity (IQ, OQ, and PQ), summarizing the outcome of testing. Any failed tests, retests, and concessions to accept software despite tests on it having failed must be discussed. Not every test has to be passed without reservation in order to allow the next qualification activity to begin, so long as any permission to proceed is justified in the reports and corrective actions to resolve any problems initiated. Each report will typically conclude with a statement authorizing progression to the next qualification activity.

Design Reviews should be revisited as appropriate to consider errors discovered during Qualification. All errors identified in a COTS product should be reported to the supplier and a response sought. If no satisfactory response is forthcoming, the seriousness of the failure should be assessed and the ensuing decision, with any mitigating further actions, recorded.

The Requirements Traceability Matrix (RTM) should be updated with details of test specifications and test reports. It should be possible to track a user requirement through Functional Specification, Design, System Build, Development Testing, and User Qualification.

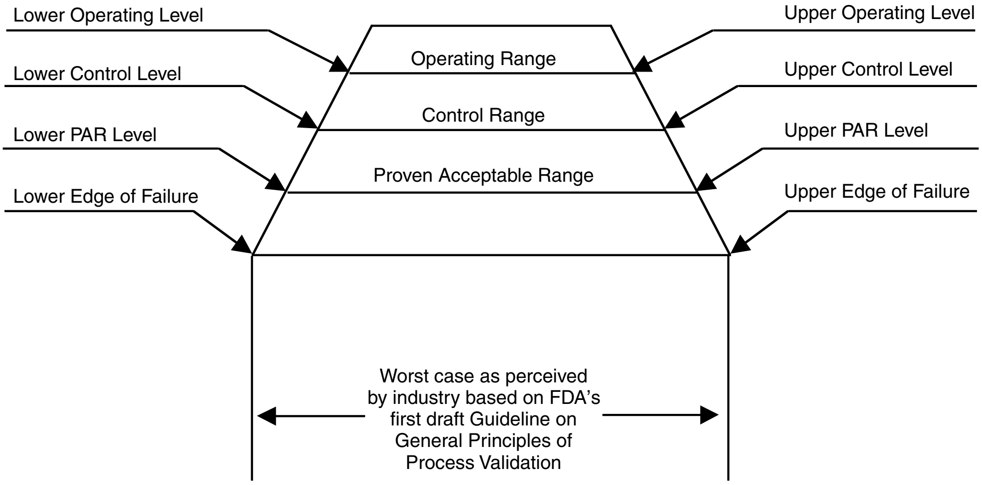


FIGURE 11.5 PMA Stress Testing Model.

Stress Testing

Testing must include worst case scenarios, sometimes referred to as stress testing. The U.S. Pharmaceutical Manufacturers Association has promoted the model illustrated in Figure 11.5 to explain the boundaries that should be exercised. It is not sufficient just to test a computer system within its anticipated normal operating range. Instead, testing should verify correct operation across a proven acceptable range. This range should exceed the control range. Processing outside the control range will cause an alarm or error to be generated. It is important that the system does not fail when it should be alarm or error handling. Testing to the point of physical failure (destructive testing) is not required and indeed should be avoided. If such severe testing is required, it should generally be conducted using simulation techniques.

Test Environment

It is becoming common to have separate development, QA, and live environments within which different levels of testing can be conducted. Development and QA environments are what is termed *off-line*, that is independent of the day-by-day operating processes. The live environment is, in contrast, *operational*. The aim is to progress testing through each environment such that:

- Development testing takes place in the off-line development environment.
- User acceptance testing occurs off-line in the QA environment.
- On-line user acceptance takes place in the live environment.

The management of development testing and controls for associated test environments are discussed in Chapter 10.

It is vital that the QA and live environments are equivalent so that test results between the two can be regarded as equivalent. Without such equivalence there is no assurance that a satisfactory test outcome in one environment will be replicated in the other environment. The QA environment should therefore be subjected to IQ demonstrating that this is, from a testing standpoint, equivalent to the intended live environment. Transport mechanisms used to move or replicate the application from one environment to another should be validated.

OQ is normally conducted in the controlled off-line QA environment. Alternatively, OQ may be conducted with the final system installed *in situ*, prior to its release for use in the live environment. Unlike OQ, PQ must *always* be conducted in the live environment.

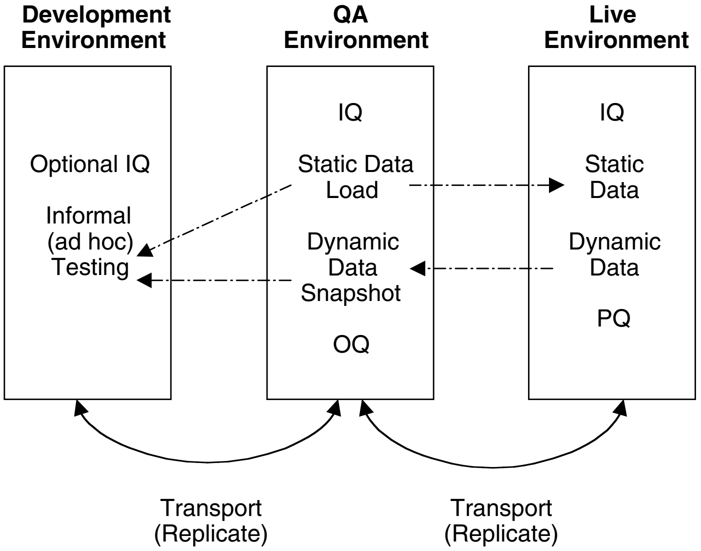


FIGURE 11.6 Test Environments.

It is vital that the QA environment is maintained under strict configuration management. There should be no software development in the QA environment. Software should be prepared in the development environment and then, when completed, transported to the QA environment. Source Code Reviews should be conducted in the QA environment. If this approach is taken, strict configuration management and change control within the development environment is not required, and it should facilitate faster software development.

Testing operations are rarely as sequential between the various test environments as the illustration in Figure 11.6 might imply. It is quite normal for testing to iterate backward through the test environments when tests fail to deliver the expected results or when testing is conducted on an incremental enhancement to an existing system. In particular, the live environment may be used to provide "snapshot" dynamic data for the QA environment, rather than having to laboriously load dummy dynamic data. Similarly, the configuration for the development environment may take for its basis the IQ from the QA environment, which is equivalent to the live environment.

Training should be conducted whenever possible within the QA environments. It is likely training will involve setting up case study situations with supporting dummy records. If the live operating environment is used for training, then care must be taken to restore any records added, modified, or deleted as a result of the training course exercises. Such data manipulation for training purposes, however, is not without risk of human error and the possible impact that could have in the live environment.

Leverage Development Testing

The scope and depth of User Qualification can be reduced if reliance can be placed on the adequacy of the supplier's Development Testing. Commercially available software that has been successfully tested by its supplier does not require the same level of user testing by the pharmaceutical or healthcare company.[3] Supplier Audit and Predelivery Inspection can be used to provide confidence and evidence in taking this approach.

Table 11.1 shows how the focus of the testing changes as Development Testing and User Qualification progresses. Inadequate Development Testing means that additional User Qualification will be expected in compensation. For example, a lack of structural (white box) testing during system development would require more rigorous user testing later on. Structural testing may not

TABLE 11.1
Changing Focus of Testing through Project Life Cycle

	Development Testing		User Qualification		
	COTS Vendor	**System Integrator**	**IQ**	**OQ**	**PQ**
Test Scope	Whole product (hardware and/or software)	Customization associated with COTS products	Hardware platform, data load, interfaces to other systems	Complete integrated system as it is intended to be used	Complete system in operational environment
Focus	Release certification of product as fit for purpose	Any COTS product configuration, new bespoke (custom) hardware and software	Introduction of computer system into working environment	GxP critical processes	GxP data and records, operational performance
Test Strategy	Comprehensive testing of product (white box)	Test user functionality (black box), including stress testing	Check completeness, confirm interfaces work	Check user functionality, challenge testing on process level	Confirm user functionality in operational environment

be possible, especially for COTS products, so comprehensive functional (black box) testing should be considered with significant stress testing.

Parallel Operation

Computer systems replacing manual ways of working should be at least as effective as the older manual process. If they are not, then they should not be authorized for use. It is for this reason that some regulations call for manual ways of working to be run in parallel with the replacement computer system, until the hoped-for improved effectiveness is demonstrated. In practice, a backout strategy for the replacement new computer system is usually developed with procedures as necessary, so that if testing demonstrates that the transition will not be successful, the *status quo ante* can be restored. Operations can return to the original ways of working, be they manual or automated. It always makes good business sense to have a contingency plan.

Running the legacy system, manual or automated, in parallel with the new system for the period of the process PQ is often not a practical option. In such circumstances processes, such as additional data checks and report verification, should be temporarily operated in parallel with the computer system until the completion of PQ.

Beta Testing

As indicated earlier, some pharmaceutical and healthcare companies agree to conduct beta testing for suppliers. Beta testing involves customers taking delivery of a system prior to its general release and then using it in its intended operating environment and reporting any problems experienced back to the supplier. The advantage to the customer is early access to a system or application. The disadvantage to the customer is that there may be yet unknown high-impact defects. Beta systems can therefore not be considered as "standard" or fully tested, as we explained earlier. More information on standard systems can be found in Chapter 8. Pharmaceutical and healthcare companies must never use beta-ware as part of a validated computer system.

Recent Inspection Findings

- The firm's software programs have not been qualified and/or validated. [FDA Warning Letter, 1999]
- Failure to exercise appropriate controls over and to routinely calibrate, inspect, or check automatic, mechanical, or electronic equipment used in the manufacturing, processing, and packaging of a drug product according to a written program designed to assure proper performance (21 CFR 211.68) in that the installation qualification (IQ), operational qualification (OQ), or performance qualification (PQ) performed for the [redacted] was not performed. [FDA Warning Letter, 2002]
- Completed IQ/OQ/PQ data not available for XXXX computer system server. [FDA 483, 2002]
- No documentation detailing IQ, OQ, and PQ of XXXX system. [FDA 483, 2001]
- Failure to perform/maintain computer validation in that there was no validation protocol to show how the system was tested and what were the expected outcomes, and there was no documentation to identify the operator performing each significant step, date completed, whether expected outcomes were met, and management review. [FDA Warning Letter, 2000]
- There was no documentation to assure that the system operated properly as intended by the vendor and performed according to the firm's intended user requirements. [FDA 483, 1999]
- The XXXX form that documents approval to migrate the program to the production environment was not signed off by Quality Control. [FDA 483, 2002]

- The firm failed to define or describe the use of the various development, test, and production environments. [FDA 483, 2001]
- The test report generated from these activities was not approved by the Quality Unit. [FDA 483 2000]
- Installation Qualification (IQ), Operational Qualification (OQ), Performance Qualification (PQ) not performed. [FDA Warning Letter, 2002]
- Firm did not maintain or refer to the location of software testing procedures. [FDA Warning Letter, 2002]

PREQUALIFICATION ACTIVITIES

The physical site of the computer system should be prepared. Some organizations treat such site preparation as part of Commissioning.

Site Preparations

The suitability of the operating environment for the computer system to be deployed[6] needs checking against that defined in the system's specification. The physical location should be compliant with any original vendor or system integrator's recommendations. The placement of the computer system, including the building of any special rooms or housing, associated wiring, and power supply voltages, must be confirmed as adequate and in line with preapproved Engineering Line Diagrams (ELDs).

Instrumentation must be accessible to facilitate operations and be covered by maintenance and calibration schedules.[7] Loop checks should be made for instrumentation. Inputs and outputs must be checked to provide strong assurance of accuracy.

Environmental requirements outlined in the Hardware Design, such as temperature, humidity, vibration, dust, EMI, RFI, and ESD, should also be checked in comparison with their acceptable bounds. Once these checks are complete, *in situ* qualification of the computer system can begin.

Commissioning

The physical installation of a computer system, often known as Commissioning, should be conducted according to preapproved procedures. Commissioning records should document fulfillment of any relevant vendor/supplier installation recommendations. Commissioning activities include:

- Interface card addressing checks
- Field wiring checks (loop testing)
- Input/output continuity testing
- Calibration and tuning of instrumentation

Computer hardware will require electrical earths and signal earths for intrinsic and nonintrinsic safety to be achieved. Wiring diagrams should be available as appropriate to site-specific installations.

Commissioning often involves an element of "snagging" to address any unforeseen issues and fix any installation errors. It should be possible to repeat installation instructions if this is a more appropriate corrective action. Verification of the installation is documented through a process of Installation Qualification.

Calibration

Instrumentation should have its predelivery calibration verified and any remaining calibration set. Calibration should be conducted with at least two known values.

The following advice is based on the *ICH Good Manufacturing Guide for Active Pharmaceutical Ingredients*:[3]

- Control, weighing, measuring, monitoring, and test equipment and instrumentation that is critical for assuring the quality of pharmaceutical and healthcare products should be calibrated according to written procedures and an established schedule.
- Calibrations should be performed using standards traceable to certified standards if these exist.
- Records of these calibrations should be maintained.
- The current calibration status of critical equipment/instrumentation should be known and verifiable.
- Equipment/instruments that do not meet calibration criteria should not be used.
- Deviations from approved standards of calibration on critical equipment/instruments should be investigated. This is to determine if these deviations affect the quality of the pharmaceutical or healthcare products manufactured using this equipment since the last successful calibration.

The *GAMP Good Practice Guide for Calibration Management*[8] further suggests:

- A calibration master list for instruments should be established.
- All instrumentation should be assigned and tagged with a unique number.
- The calibration method should be defined in approved procedures.
- Calibration measuring standards should be more accurate than the required accuracy of the equipment being calibrated.
- Each measuring standard should be traceable to a nationally or internationally recognized standard where one exists.
- Electronic systems used to manage calibration should fulfill appropriate electronic record/signature requirements.
- There should be documentary evidence that all personnel involved in the calibration process are trained and competent.

The contents for a Calibration Master List are suggested below:[8]

- Asset
- TAG
- Device description, manufacturer, and serial number
- Device range (must satisfy process requirements)
- Device accuracy (must satisfy process requirements)
- Process range required
- Process accuracy required
- Calibration range required (to satisfy process requirements)
- Calibration frequency (e.g., 6 months, 12 months)
- Device criticality (process critical, product critical, or noncritical)

Calibration certificates should be prepared where they are not provided by third parties and retained as a regulatory record. Many pharmaceutical and healthcare companies are installing computer systems to manage calibration master lists and calibration certificates that support resource scheduling. Such computer systems should be validated. An example calibration certificate is shown in Table 11.2.

TABLE 11.2
Example Calibration Certificate

Calibration Test Sheet	Electronic Temperature Transmitter
Department	Complies with Procedure Number:
Service	Temperature Element Serial Number:
	Temperature Transmitter Serial Number:
Location/Use	Control Loop/Tag Number:
	Instrument Range:
	Critical Device □ Noncritical Device □

Electronic Temperature Transmitter	
Manufacturer:	Type and Model:
Process Range __________to___________	Device Accuracy ___________________ ± ˚C
Calibrated Range ________to___________	Specified Process Accuracy ___________ ± ˚C

Calibration					
Standard RTD Serial Number	Standard RTD Temperature	Signal Output (mA)	Temp. Output Equiv. (˚C)	Error (˚C)	Pass/Fail

Post Adjustment Calibration					
Standard RTD Serial Number	Standard RTD Temperature	Signal Output (mA)	Temp. Output Equiv. (˚C)	Error (˚C)	Pass/Fail

Test Equipment Details				
Equipment	Manufacturer	Model Number	Serial Number	Certificate Number
Digital Multimeter				
Standard Reference				
Standard RTD				
Standard RTD				
Standard RTD				

Conclusion
The combination of the above Test Equipment is able to calibrate a device to an accuracy of ………….˚C
Comments/Observations:

Test Performed and Recorded by:	Name:	Signature:	Date:
Checked by:	Name:	Signature:	Date:

Self-calibrating features should not be relied upon to the exclusion of any external performance check. The frequency of periodic checks on self-calibrating features will depend on how often the features are used, scale, criticality, and tolerance. Typically annual checks should be conducted on self-calibrating features.

Recent Inspection Findings

- Failure to assure [computer] equipment is routinely calibrated, inspected or checked according to a written program design to assure proper performance. [FDA Warning Letter, 2000]
- Procedures for calibration of various instruments lacked some or all of the following information: persons responsible for the calibration; specifications or limits; action taken if a test fails; and a periodic review by management. [FDA Warning Letter, 2001]
- No QA program for calibration and maintenance of the XXXX system. [FDA 483, 2002]
- There is no documentation that equipment calibration was performed when scheduled in your firm's procedures. [FDA Warning Letter, 2001]
- Your procedures for calibration are incomplete, for instance no predetermined acceptance criteria. [FDA Warning Letter, 2002]
- Failure to maintain calibration checks and inspections. [FDA Warning Letter, 2002]
- Inadequate SOP for review and evaluation of calibration reports from outside contractors. [FDA 483, 2001]
- No procedure for corrective and preventative action when equipment outside calibration range. [FDA 483, 2001]

DATA LOAD

The reliance that can be placed in a computer system is fundamentally determined by the integrity of the data it processes. It must be recognized that data accuracy is absolutely vital in the business context. However well an application works, it will be fundamentally undermined if the data it processes is dubious. Data load is a key task that must be adequately managed to satisfy business and regulatory needs. Loading of data can be broken down into five basic steps: data sourcing, data mapping, data collection, data entry, and data verification.

Data Sourcing

Data sourcing consists of defining, in existing systems or documentation, the master reference (prime source) for the data entities required to support the new system. In some instances data may need to be created because they do not already exist electronically.

A top-level definition of static data as fixed, and dynamic data as subject to changes, is not necessarily as clear as it sounds. Most data actually change in practice, but it is the *frequency* of change that is important when considering what is static and dynamic data. It should be possible to check static data against a master reference to verify that it is correct. No such check can typically be done for dynamic data because by its nature it changes frequently, so a check can only be made against its last known value. Examples of static and dynamic data include recipes and supplier details. Examples of dynamic GxP data include date of manufacture, batch number, notification of deviation, planned change, analytical results, and batch release.

Data Mapping

Data mapping is the process of identifying and documenting, for every field being populated in the new system, where the data is to be found in existing systems (or documents). The mapping of each field will be classified as follows:

Simple: There is an obvious legacy field equivalent, or lack of equivalent, to the new system field.

Complex: There is information in the legacy environment but, before it is suitable for entry into the new system, the field length or format needs to be changed. Perhaps the field needs to be transformed, several fields need to be combined, a field in the legacy system needs to be split to feed several fields in the new system, or there may be a combination of all or some of these.

Data mapping should consider any electronic record implications such as maintaining audit trails during data migration. Electronic record requirements are discussed in more detail in Chapter 15.

Data Collection

The method of data collection is affected by the approach taken to loading data into the new system (i.e., electronic or manual). The criteria used to decide whether to load manually or electronically include:

- Whether a standard program exists for the data transfer of the particular business object in the new system
- The availability of the data in electronic form
- The number of records that need to be transferred
- The feasibility within the constraints of the project (e.g., time, available resources with the appropriate skill sets)
- Expected error rates

Data Entry

Data entry needs to be verified as accurate against master references (system sources and/or documents). Data from different sources may need to be aggregated during migration, or perhaps some reformatting might be required (e.g., field lengths). The manipulations need to be verified as having been conducted correctly. Checks are also required for transcription errors. Transcription error checks should be conducted as indicated below for dynamic data. The creation of backup copies of the original data will be regularly scheduled, following defined procedures, to provide a fallback position in the event of problems. A further (sanity) check is often beneficial at this stage to double-check that there have been no misinterpretations of the business object/field information.

Manual data entry errors might run at a 0.5% error rate but must be expected to be much higher. If spreadsheets are used as a medium to transfer data, then error rates typically in the range of 20 to 40% should be expected. Where critical data are being entered manually, there should be an additional check on the accuracy of the entry.[3,6,7] This can be done by a second operator or by the system itself.

Data Verification

While all GxP data should be checked for correctness, it may be possible to justify a sample check for other data categories if a business case can be made to justify the omission of checks on all records. Some regulators require a double check for GxP data entry. Such checks should immediately follow the data entry and precede any further processing of the data. Where this is not possible, checking must be conducted as soon as possible and a risk assessment performed to address the potential consequences of erroneous data input.

Data entry and data checking should be considered as separate activities. Each activity must be traceable to the individuals carrying out the activity and the date on which it was performed.

Individuals who perform data checking must be trained in data accuracy as a minimum requirement. Additional training may be necessary as appropriate to the level of checking being performed.

Recent Inspection Findings

- Input data validation methods not always defined. [FDA Warning Letter]
- Validation not conducted after XXXX data was migrated to new server. [FDA 483, 2002]

INSTALLATION QUALIFICATION

IQ provides documented verification that a computer system has been installed according to written and preapproved specifications.[9]

The integration of the computer system (hardware, software, and instrumentation) must be confirmed in readiness for the subsequent OQ activity. Some practitioners have referred to this as the *testing of static attributes* of the computer system. The importance of completing the IQ before commencing the OQ can be illustrated by a recent incident in which a pharmaceutical company had over 35% of the instrumentation for a multiproduct plant but did not have available calibration certificates. There were various reasons for this, but none were recorded. Some instruments were no longer used, some had extended recalibration periods, and some had been undergoing calibration for several weeks. The net effect was that the computer system under qualification was clearly not in a controlled state suitable for the OQ, and in consequence, it was not ready for use.

Scope of Testing

IQ should focus on the installation of the hardware platform, the loading of data, and the setting up the interfaces to other systems. This will include the following:

- Inventory Checks
- Operational Environment Checks
- Diagnostics Checks
- Documentation Availability

IQ testing should embrace the test environments as discussed earlier in this chapter. Appendix 11A and Appendix 11B provide checklists that may be used in the development of an IQ protocol.

Inventory Checks

The FDA and other regulatory authorities require that all major items of equipment be uniquely identified. All the specified components of the system should be present and correct including printers, Visual Display Units (VDUs) and touch screens, keyboards, and computer cards. The identifying serial numbers and model numbers of all the major items must be recorded. The question as to whether units of equipment need to be dismantled in order to check their component details is often raised. If a unit is sealed in such a way that dismantling would invalidate the manufacturer's equipment warranty, then disassembly should not be attempted; it is not required in these circumstances. The IQ should simply check the unique identity of the sealed unit. Processing boards that are clip-fastened into slots in a rack should have their serial numbers recorded, along with their slot position within the rack. It is worth checking with suppliers in advance of delivery whether their equipment does in fact have unique identifiers.

The correct versions of software must be installed and appropriate backup copies made. The correct versions of firmware must also be checked for their presence. This may include a physical inspection of an Electronically Programmable Read Only Memory (EPROM) to read its label. The configuration of databases and the content of any library information should also be checked. The last three generations of backup should be retained. The storage medium for the software must be labeled with the software reference name and version. Facilities should exist to store the backup files in a separate and secure place.[7] Fireproof cabinets or rooms should be used wherever possible.

Operational Environment Checks

Operational environment checks should include those on power supplies, ambient temperature and humidity, vibration and dust levels, Electro-Magnetic Interference (EMI), Radio Frequency Interference (RFI), and Electrostatic Discharges (ESD) as relevant to the needs of the computer system. This list of operational environment requirements is by no means exhaustive, and may be extended or even reduced depending on what is known about the system. EMI and RFI might be tested with the localized use of mobile or cell telephones, walkie-talkie communications receivers/transmitters, arc welding equipment, and electronic drills. The aim is to test the vulnerability of the computer system to interference in situations that must be considered as normal working conditions.

Diagnostic Checks

Diagnostic checks are normally conducted as a part of the IQ. Such checks include those of the built-in system configuration, conducting system loading tests, and checking timer accuracy. Software drivers, such as communication protocols, will also require testing.

Documentation Availability

All documentation furnished by the supplier should be available. User manuals, as-built drawings, instrument calibration records, and procedures for operation and maintenance (including calibration schedules) of the system should all be checked to verify that they are suitable. Supplier documentation should be reviewed for accuracy in its specifications of the various versions of software used and approved as fit for purpose. It is recommended that checks are made to verify that contingency plans, SOPs, and any Service Level Agreements (SLAs) are also in place. Any specific competencies supposed to be acquired before the IQ/OQ/PQ through training should also have been achieved — these records should be checked.

Recent Inspection Findings

- Proper installation and verification of functionality was not performed for software version loaded. [FDA Warning Letter, 1999]
- The Installation Qualification (IQ) protocol stipulated that all required software be installed, but the protocol did not state what software was required. [FDA 483, 2002]
- Software used "out of the box" without deviation report or investigation into configuration error. [FDA 483, 2002]
- Headquarters has failed, despite deviations and problem reports, to establish adequate control of software configuration settings, installation qualification, and validation. [FDA 483, 2002]

OPERATIONAL QUALIFICATION

Operational Qualification (OQ) provides documented verification that a computer system operates according to written and preapproved specifications throughout all its specified operating ranges.[9]

OQ should only commence after the successful completion of the IQ. In short it comprises user acceptance testing, for it is necessary to demonstrate that the computer system operates in accordance with the Functional (Design) Specification. Individual tests should reference appropriate Functional Specifications. Testing should be designed to demonstrate that operations will function as specified under normal operating conditions and, where appropriate, under realistic stress conditions.

An OQ Summary Report should be issued on completion of OQ activities. Simpler computerized systems may combine the IQ and OQ stages of validation into a single activity and document this accordingly. More complex computerized systems may be divided into subsystems and subjected to separate OQ. These exercises should then be complemented by a collective OQ demonstrating that the fully integration system functions as intended.

Scope of Testing

OQ should focus on GxP-critical processes. It should:

- Confirm that critical functionality works, including hazard controls.
- Verify that disabled functionality cannot be accessed.
- Check the execution of decision branches and sequences.
- Check important calculations and algorithms.
- Check security controls — system access and user authority checks.
- Check alarm and message handling — all important error messages designed into the system should be checked to ensure that they appear as intended under their relevant error conditions (it may be wholly impractical to check *all* the error messages).
- Confirm the creation and maintenance of audit trails for electronic records.
- Confirm the integrity of electronic signatures including, where appropriate, the use of biometrics.

Additional tests demanded or recommended as a result of the findings of the Supplier Audit, Source Code Review, or Design Review activities should also be included. Appendix 11A and Appendix 11C provide checklists that can aid in the development of an OQ protocol.

Test Reduction

The OQ may be based on a repetition of a chosen sample of the Development Testing tests in order to reduce the amount of OQ testing conducted.[6] As discussed earlier, this is only permissible where extensive Development Testing has been successfully conducted (i.e., without significant defects emerging) and recorded. The suitability of such documentation must be reviewed and approved by QA for this purpose. The test sample for OQ must include, but not be limited to, those tests originally conducted as emulations and simulations. Simulation and emulation specifically for Qualification should be avoided.[5] If the repeated tests of the chosen sample do not meet their acceptance criteria (i.e., if fresh system defects emerge), then the causes of such failures must be thoroughly investigated and an extended sample of tests repeated if confidence in the new system is not to be fatally undermined. The advantage in this approach is that commissioning time on the pharmaceutical and healthcare company's site is reduced, and the system can become fully operational sooner, provided all is well. It might be argued that repeating the supplier's Development Testing does not contribute to an increasing level of assurance of the fitness for purpose of the system. However, practical experience suggests that crucial deficiencies are often discovered in systems even at this late stage in the life cycle. This is very worrying, for obvious reasons — it implies that much of the preceding effort to confirm the innate quality of the system has missed its target. Here are just a few examples of such late-stage failures:

- Backup copies of the application software did not work.
- A computer system froze when too many concurrent messages were generated.
- The operator of a control system would never become aware of concurrent alarm messages as the graphic pages bearing them had banners that only permitted the display of the latest-generated alarm.
- When "on" and "off" buttons were pressed simultaneously, the computerized system initiated an equipment operation.
- Computer software was able to trigger the controlled equipment into operation despite the fact that the hardwired fail-safe lockout device had been activated.

Verifying SOPs

Operations personnel must be able to use all operating procedures *before* the computer system is cleared for live use. User Standard Operating Procedures (SOPs) can be used to confirm system functionality. Any competencies required to conduct these tests, including training on user SOPs, should be given and recorded before testing begins.

System Release

Computerized systems are often released into the live environment following completion of OQ. An interim Validation Report or an alternative document such as a System Release Note should be prepared, reviewed, and approved in order to authorize the use of the system. The interim report should address all aspects of the Validation Plan up to and including the OQ. Several draft Validation Reports of this kind may be required in order to phase the rollout of components of the overall system or where a phased rollout is planned to multiple sites.

Recent Inspection Findings

- No testing of the [computer] system after installation at the operating site. Operating sites are part of the overall system and lack of their qualification means the system validation is incomplete. [FDA 483]
- Testing was not conducted to insure that each system configured could handle high sampling rates. Validation of the system did not include critical system tests such as volume, stress, performance, boundary, and compatibility. [FDA Warning Letter, 2000]
- There was no assurance that complete functional testing has been performed. [FDA Warning Letter, 2001]
- Regarding the recent functional [Y2K program update] testing conducted on XXXXXX:
  1. General test plans lack a document control number and lack approval by the Quality Unit.
  2. Detailed test plans lack a document control number and lack approval by the Quality Unit.
  3. Test Scripts lack indication of review or approval.
  4. The report generated from these activities lacked a document control number, was not approved by the Quality Unit. Additionally, this report commits to correct errors identified in XXXXXX during this testing. The original commitment in this report is for corrective actions to be delivered by March 31, 1998. Subsequently this plan was updated to have corrections delivered by March 31, 1999. The firm produced no report, which addresses the corrections made in response to this report. [FDA 483, 2000]
- Validation is incomplete … e.g., does not call for testing of the [computer] system under worst case (e.g., full capacity) conditions, and lacks testing provisions to show correct functioning of software. [FDA Warning Letter, 1999]

- Software testing has not been conducted simulating worst case conditions.
- The alarm system and its backup for the XXXX are not challenged to demonstrate that they would function as intended. [FDA Warning Letter, 2000]
- Testing has not included test cases to assess the password security system. [FDA 483, 2001]
- Inadequate qualification in that no power failure simulations were performed as required by the firm's protocol. [FDA 483, 2002]
- Your firm failed to properly maintain electronic files containing data secured in the course of tests. [FDA Warning Letter, 1999]
- There was no testing of error conditions such as division by zero, inappropriate negative values, values outside acceptable ranges, etc. [FDA 483]
- Testing of special values (input of zero or null) and testing of invalid inputs … are not documented.
- The procedure does not call for error condition testing.
- Alarm system is unable to store more than XX transgressions, and these transgressions are not recorded. [FDA Warning Letter, 2000]

PERFORMANCE QUALIFICATION

Verifying whether or not a computer system is fit for its intended purpose often means designing tests that are directly related to the manufacture of drug products. PQ therefore provides documented verification that a computer system is capable of performing or controlling the activities of the processes required to perform control, according to written and preapproved specifications, while operating in its specified operating environment.[9]

PQ should only commence after the successful completion of the OQ stage. It comprises product performance and/or process performance qualification. At this stage, the pharmaceutical or healthcare company must demonstrate that the completed installation ("as-built") of the computer system at the site is operating in accordance with the intent of the URS. PQ is sometimes also referred to as a part of Process Validation, where the computer system supports a production process. A fundamental condition within PQ is that changes may be made to the computer system during testing. If the need for change emerges as a result of test failures, PQ must be repeated in its entirety. The underlying principle here is that the change may have disrupted system stability and reproducibility.

Scope of Testing

Performance Qualification should focus on GxP data and records and operational performance. It must prove that:

- GxP records are correct.
- Automated processes are reproducible.

The degree of testing will also be influenced by the amount of OQ testing already conducted. Appendix 11A and 11D provide checklists that can be used to assist the development of a PQ protocol.

Product Performance Qualification

Product PQ is a quality control activity that aims to verify the correct generation of GxP records. A matrix approach might be required to cover the practical range of acceptable variations. Some examples of product PQ tests are:

- Creation of batch reports (startup, sequencing, and closeout of consecutive batch processes)
- Data/analysis checks of custom user reports
- Structure and content checks for label variants
- Checks of presentation details on product packaging variants

Batch reports for PQ include batch records (e.g., those relating to key manufacturing steps such as media fills, cleaning), product release (i.e., sentencing), packaging (including labeling), and product batch distribution records (for batch tracking and recall). The PQ for multiproduct applications should cover necessary variants. The PQ exercise should test the system's operation in handling a minimum of three production batches or five for biological applications. The number of consecutive batches required, however, is not fixed and will depend on the process being validated.

The content and format of batch records must be defined within the system specification. Automated batch records must provide an accurate reproduction of master data[7] and deliver a level of assurance equivalent to a double manual check, bearing in mind that manual checks can identify and record unexpected observations.[7,10] Computer systems releasing batches must be designed to demand an authorization for each batch, and the identity of responsible person giving this must be recorded against the batches.[6,7,11] All batch records require quality control inspection and approval prior to release and distribution of the product.[7] The identity of operators entering or confirming data should be recorded. Authority to change data and the reasons for such changes should be recorded in an audit trail. Similar requirements apply to labeling and packaging.

Process Performance Qualification

Process PQ is a quality assurance activity that aims to verify that the automated process is reproducible. Process PQ is sometimes referred to as *Post Implementation Review* and is based on performance monitoring rather than testing. Examples of some process PQ topics are:

- Demonstrating that the correct functionality of the system is not disrupted during acceptable daily, calendar, and seasonal operating environment variations (e.g., variations in power supply, temperature, humidity, vibration, dust, EMI, RFI, and ESD)
- Demonstrating that an acceptable level of service continuity is achieved (e.g., availability, failure on demand, and reliability)
- Demonstrating the effectiveness of SOPs and training courses
- Demonstrating that the users are being adequately supported (e.g., through a reduction in the rate of enquiries received from them, with a decreasing number of outstanding responses/resolutions to their questions)

Variations in temperature and humidity might be monitored over a period of time using a portable chart recorder as part of the PQ. Vulnerabilities to electrostatic discharge (ESD), vibration, and dust are more difficult to measure. All that may be possible in this context is to periodically review whether these have affected live operations in any way. If this is the case, it should be clearly stated and the causes followed up as part of the ongoing support program for maintaining validation.

Service organizations should set up processes to collect and analyze operational performance data. Examples of performance charts are shown in Chapter 12. Performance metrics to be tracked and acceptable service levels to be met should be specified in Service Level Agreements. Performance charts might include monitoring the training and help desk activity as indicted in the bullet points above.

Recent Inspection Findings

See Performance Monitoring in Chapter 12.

AUTHORIZATION TO USE

Pharmaceutical and healthcare products should not be released to market when the processes and equipment used to manufacture them have not been properly validated. This includes necessary validation of computer systems. Annex 11 of the *European Guide to GMP* imposes specific rules regarding the validation of computerized systems,[6] when these are used for recording certification and batch release.[12] The only possible exception to this rule should be when all of the following criteria are met:

- The pharmaceutical medicines and healthcare products (e.g., medical devices) concerned are for life-threatening diseases or situations.
- There is no equivalent pharmaceutical or healthcare product available in the marketplace.
- The supply of available treatments or medicines has fallen to a critically low level.

In such extreme situations justifications for releasing pharmaceutical and healthcare products to market under these most exceptional conditions must be fully documented by responsible personnel, approved by senior management, and agreed in advance with relevant regulatory authorities.

Validation Report

Validation Reports are prepared in response to Validation Plans. Their purpose is to provide to management a review of the success of the validation exercise and any concessions made during it. The objective of the report is to seek their endorsement of the completion and acceptance of the validation conducted. Validation Reports may also document failed validation and instruct design modifications and further testing. The FDA and other regulatory authorities may request a translation if the original document has been drafted in a language other than English, so that their inspectors can scrutinize the document themselves during an inspection.

Validation Reports should be prepared by the person instructed and authorized by management to do so in the Validation Plan or in another relevant procedure. Where this is not the case, the authority under which the report is written should be stated.

It is recommended that Validation Reports follow the same structure as their corresponding Validation Plans so that the two documents can be read side by side and the course of the validation followed step by step. Figure 11.7 illustrates this relationship. A summary for each phase of the validation exercise should be prepared. Details of test outcomes, test certificates, documentation, etc., should be included. Test environments should be described in outline, and any test prerequisites discussed in case they qualify, or even undermine, the overall validation conclusion reached.

The GAMP Guide suggests that the Validation Report should include the following information regarding each phase of validation:[9]

- Reference to the controlling specification for the phase
- Confirmation that all tests or verification were executed and witnessed (if applicable) by suitably qualified and authorized personnel. This includes all supplier factory testing and site acceptance testing
- Details of any supporting resources involved — names, job titles, and qualifications
- Locale and environment for any testing

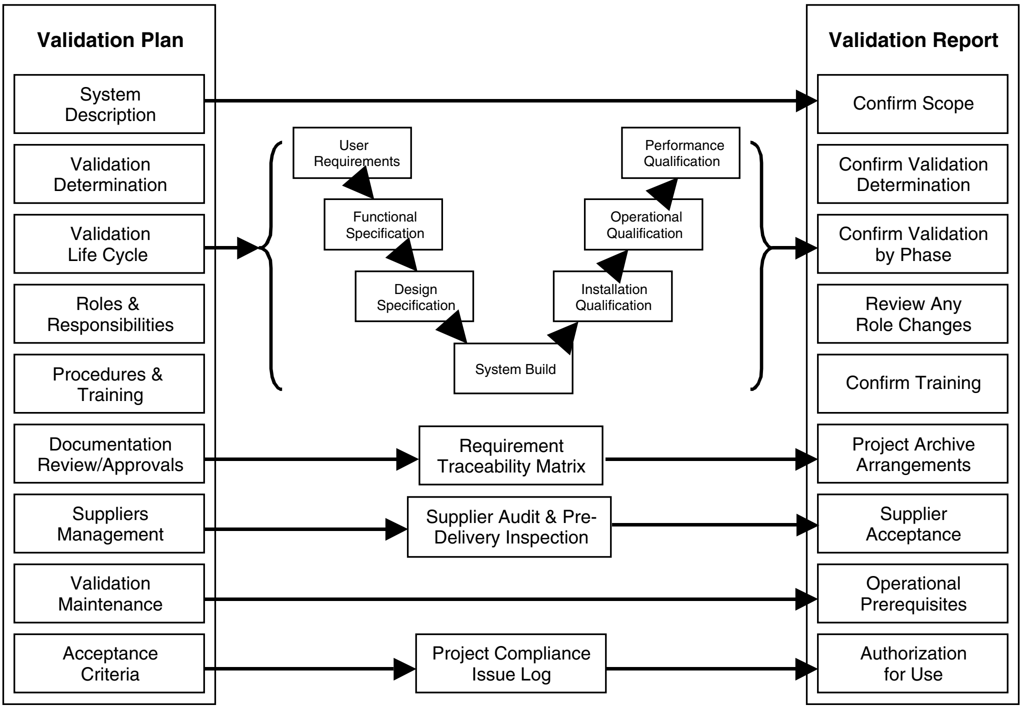


FIGURE 11.7 Relationship between Validation Plans and Reports.

- Confirmation of the dates over which the phases occurred, with explanations of delays and actions taken to resolve them
- Confirmation that all tests and activities were subjected to regular project team and QA reviews, with reference to supporting evidence

Each phase of validation should have a clear unambiguous statement drawing a conclusion on the validation status that the evidence provided is reckoned to justify. The overall validation conclusion should then come as no surprise, provided each phase has satisfied its predetermined acceptance criteria.

The breakdown of results should be summarized. The GAMP Guide recommends that a tabular format be used. The report should refer to the original test records and test specification documents. The summary table should contain, as a minimum, those tests that resulted in failure or deviation.

Any deviations and interventions to the pharmaceutical or healthcare company's Validation Plan or Supplier's Project/Quality Plan must be recorded, their impact on validation assessed, and their true cause investigated. Deviations and interventions may include changes to SOPs during validation, concessions on the acceptability of unexpected test results, or modifications to the life-cycle model to make it more appropriate.

Validation Reports should also identify each and every issue not resolved by a corrective action during the project. Table 11.3 provides an example of part of a Project Compliance Issue Log which can be used within a Validation Report. The table provides details of the variance, why it occurred, and how it was resolved. It also furnishes a written justification for situations where a corrective action is not possible or appropriate. Similarly, suppliers may supply a report summarizing their own validation work, which can also be referenced by the Validation Report.

The Validation Report authorizing use of the computer system should not be issued until all operation and maintenance requirements, including document management, calibration, maintenance, change control, security, etc., have been put in place.

TABLE 11.3
Example of Part of a Project Compliance Issues Log

Issue No.	Author and Date Identified	Description	Resolution	Justification	Status
98	E. Thomas October 10, 2003	IQ Test Failure — wrong version of application software loaded	No Action Annotate correction to test record and accept test result against original version observed	Test script had typo — correct version of software was loaded actually correctly as required	Closed
99	S. Pattison October 22, 2003	OQ Test Failure — standard reports would not print when requested	Change Control Reference 37 Printer setup error Reconfigure printer and retest	Not Applicable	Closed
100	G. Smith October 22, 2003	OQ Test Failure — system does not save updated records	No Action Software error identified and confirmed by vendor	Function is not used, no impact elsewhere	Closed

It is essential that the validation status of the system does not become compromised. Revalidation will be required if validation controls are not being implemented. The costs of revalidation can be in excess of five times that of ensuring validation controls were available and used in the first place; see Chapter 12 and Chapter 17 for a further discussion. Management must ensure that their organization's investment in validation is not effectively jettisoned.

QA must approve Validation Reports. For European pharmaceutical and healthcare companies this is likely to be a responsibility of the registered Qualified Person.[13]

Validation Summary Report

Validation Summary Reports are usually prepared to accompany Validation Master Plans, although this is not necessarily always the case. They provide an executive summary of the Validation Report and need to be approved by QA. Details of deviations should not be included; the report simply provides a walk through the succession of project stages, identifying key deliverables. The GAMP Guide suggests the following contents:[9]

- Provide the mapping of the activities performed against those expected in the Validation (Master) Plan.
- Provide a summary of the validation activities undertaken.
- Provide reference to evidence that these activities are in compliance with the stated requirements.
- Confirm that the project documentation has been reviewed and approved as required by appropriate personnel.
- Confirm training has been provided and documented as planned.
- Confirm that documentation has been created to show that all the records related to validation have been securely stored.
- Specify the approach to maintaining the validated status during the operational phase.
- Confirm all project compliance issues that were logged during the project have been resolved satisfactorily.
- Report that the project has been successfully completed.

It is sometimes necessary to modify the original intent of a computer system or validation strategy to some degree in order to achieve an acceptable outcome. The Validation Summary Report should highlight and justify such changes of direction. As for Validation Reports, Validation Summary Reports should be made available to the FDA in English.

Validation Certificate

The concept of a Validation Summary Report can be taken a stage further in the form of a Validation Certificate. Such certificates consist of a one-page summary statement defining any constraints on the use of the computer system. An example of Validation Certificate is shown in Table 11.4. Validation Certificates are sometimes displayed alongside the computer system itself, where the system is a single discrete item. Certificates for distributed systems do not normally make sense since there are too many potential points of use alongside which to display such a certificate. Validation Determination Statements (described earlier in Chapter 6) can be presented to an inspector with reciprocal Validation Certification as the very highest level of evidence of validation. If Validation Certificates, they should be approved by QA.

Recent Inspection Findings

- Failure to establish and maintain procedures for final acceptance. [FDA Warning Letter, 1999]
- No Validation Report was written following execution of validation protocol. [FDA 483, 2002]
- Incomplete Validation Report. [FDA 483, 2001]
- Failure to perform/maintain computer validation in that there was no documentation to show if the validation was reviewed prior to software implementation. [FDA Warning Letter, 2000]
- The inspection reports that the documents reviewed did not define the system as being validated but was a qualification document. [FDA Warning Letter, 2001]
- Validation Report approved although deviations were not adequately investigated. [FDA 483, 2002]
- Password Master List made globally available in Validation Report. [FDA 483, 2002]
- The validation of the computer system used to control the XXXX process is incomplete. Your proposed corrective actions for deficiencies 2, 3, and 4 regarding validation appear satisfactory except that the validations will not be completed until the end of March, 2001 and imply that you will continue to use the unvalidated computer systems and equipment cleaning methods until them. [FDA Warning Letter, 2000]
- The firm has failed to generate validation summary reports for the overall program throughout its software life cycle. [FDA 483, 2001]
- The validation summary should include items such as how the system is tested, expected outcomes, whether outcomes were met, worst case scenarios, etc. [FDA Warning Letter, April 2000]
- Computer enhancement was identified as needed to correct labeling deviations but this enhancement was still not implemented over one year later. [FDA 483, 2002]

TABLE 11.4
Example Format of a Validation Certificate

System Name	Electronic Batch Record System
Controlling Specification Reference	EBRS/FS/03
Validation Plan Reference	EBRS/VP/02

FINAL SYSTEM VALIDATION APPROVAL

The signatories below have reviewed the validation package for the *[name of the supplier (vendor), and name of system]* computer system. The review included the assessment of the phase reports listed below, including details of the execution of approved test scripts, test phase conclusions based on test phase acceptance criteria, and resolution of items listed in issues log. The determined validated status is derived as a culmination of this review process.

Key Validation Package Documentation	Document Reference	Acceptance Criteria Satisfied (Yes/No)
Supplier Audit	EBRS/SA/01	Yes
Design Review	EBRS/DR/02	Yes
Source Code Review	EBRS/SCR/01	Yes
Predelivery Inspection	EBRS/PDI/01	Yes
Installation Qualification — Peripherals	EBRS/IQ1/01	Yes
Installation Qualification — QA Test Environment	EBRS/IQ2/01	Yes
Installation Qualification — Production Environment	EBRS/IQ3/01	Yes
Operational Qualification — User Functionality	EBRS/OQ1/03	Yes
Operational Qualification — Interfaces	EBRS/OQ2/02	Yes
Operational Qualification — Security	EBRS/OQ3/01	Yes
Performance Qualification	EBRS/PQ/01	Yes
Project Issues Log	EBRS/PIL/12	Yes
Validation Report	EBRS/VR/01	Yes

VALIDATION STATUS DECLARATION

In consequence, we determine that the *[name of system]* has been validated in accordance with requirements of its Validation Plan, and we authorize its use by suitably trained and qualified personnel. We affirm that this system must be maintained in order to preserve its validated status.

APPROVAL DATE

[must be entered after approval signatories below have been added, but prior to first date of use]

Each individual signing below approves the validation status of the [name of system] computer system.

Name	Job Title	Signature	Date
[System Owner/User]			
[Quality and Compliance]			

REFERENCES

1. European Union, *Annex 15 — Qualification and Validation*, European Union Guide to Directive 91/356/EEC.
2. FDA (1995), *Glossary of Computerized System and Software Development Terminology*, August.
3. ICH (2000), *Good Manufacturing Practice Guide for Active Pharmaceutical Ingredients*, ICH Harmonised Tripartite Guideline, November 10.
4. FDA (2002), *General Principles of Software Validation; Final Guidance for Industry and FDA Staff*, U.S. Food and Drug Administration, Rockville, MD.
5. Wingate, G.A.S. (1997), *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice*, Interpharm Press, Buffalo Grove, IL.
6. European Union (1993), *Annex 11 — Computerised Systems*, European Union Guide to Directive 91/356/EEC.
7. U.S. Code of Federal Regulations Title 21: Part 211, *Current Good Manufacturing Practice for Finished Pharmaceuticals*.
8. ISPE (2002), “Calibration Management,” *GAMP Good Practice Guide*.
9. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (www.ispe.org).
10. TGA (1990), *Australian Code of Good Manufacturing for Therapeutic Goods*, Medicinal Products, Part 1, Therapeutic Goods Administration, Woden, Australia.
11. FDA (1982), *Identification of “Persons” on Batch Production and Control Records*, Compliance Policy Guides, Computerized Drug Processing, 7132a, Guide 8, Food and Drug Administration, Center for Drug Evaluation and Research, Rockville, MD.
12. European Union, *Annex 16 — Certification by a Qualified Person and Batch Release*, European Union Guide to Directive 91/356/EEC.
13. Article 12 of EU Directive 75/319/EEC and Article 29 of EU Directive 81/851/EEC.

APPENDIX 11A EXAMPLE QUALIFICATION PROTOCOL STRUCTURE (BASED ON THE GAMP GUIDE[9])

Introduction

Test Plan

- Specific areas that have not been tested, with justification for this test procedure explanation
- Action in event of failure
- Logical grouping of tests
- How to record test results

Test Requirements

- Personnel
- Hardware
- Software (including configuration)
- Test harness
- Test data sets
- Referenced documents

Test Prerequisites

- Relevant documents must be available
- Test system defined
- Critical instruments must be calibrated

Testing Philosophy

- Witness and tester must be agreed upon by customer
- Test results must be countersigned by both witness and tester

Test Procedure Format

- Unique test references
- Controlling specification reference (cross-reference)
- Title of test
- Prerequisites
- Test description
- Acceptance criteria
- Data to be recorded
- Further actions

Test Procedure Execution

- Endorse the outcome as pass or fail
- Attach raw data
- Report unexpected incidents and noncompliances
- Failed tests may be completed or abandoned
- A change or a repair may trigger a fresh set of tests to verify the patch

Test Results File

- Test progress section
- Passed test section
- Failed test section
- Test incident section
- Review report section
- Working copies of test scripts
- Test result sheets and raw data

Test Evidence

- Raw data
- Retention of test results
- Method of accepting completion of tests

Glossary

References

APPENDIX 11B
EXAMPLE INSTALLATION QUALIFICATION CONTENTS

Scope

- Visual check on hardware
- Power-up and power-down
- Inventory of software installed (with versions)
- System diagnostic testing
- Verify acceptable operating environment (e.g., power supply, EMI, RFI)
- Computer clock accuracy testing
- Check that all the SOPs are in place
- Check that the documentation has been produced and are available, including the User Manuals
- Confirm that training has been conducted

APPENDIX 11C
EXAMPLE OPERATIONAL QUALIFICATION CONTENTS

SCOPE

- Startup and shutdown of application
- Confirm user functionality (trace the test results back to the user requirements)
  - Correct execution of decision branches and sequences
  - Correct display and report of information
  - Challenge user functionality with invalid inputs
- Verify deselected or disabled functionality cannot be accessed or reenabled
- Check application-specific calculations and algorithms
- Check security controls — system access and user authority
- Check alarm and message handling — all error messages
- Verify that trips and interlocks work as intended
- Check creation and maintenance of audit trails for electronic records
- Verify integrity of electronic signatures
- Ensure backup, media storage arrangements, and restore processes exist and have been tested
- Ensure archive, retention, and retrieval processes exist
- Check for existence of business continuity plans, including recovery after a catastophe
- Verify battery backup and UPS cut-in upon a power failure

APPENDIX 11D
EXAMPLE PERFORMANCE QUALIFICATION CONTENTS

Scope of Product PQ

- Check batch reports
  - Production records against plant logbooks for inconsistencies
- Check data accuracy and analysis for custom user reports
  - Cycle counting
  - Period ending cycles
  - Inventory reconciliation
  - Release processes
- Check label variants
  - Structure
  - Content
- Check product packaging variants
  - Presentation details

Scope of Process PQ

- Operability during daily, calendar, and seasonal operating variations
  - Environmental (e.g., variations in power supply, temperature, humidity, vibration, dust, EMI, RFI, and ESD)
  - Peak user loading
- Acceptable level of service continuity is maintained
  - System availability (planned and unplanned downtime)
  - Access denial on demand
  - Security breach attempts
  - Data performance (e.g., network, database, disk)
- Effectiveness of SOPs and training
  - Suitability of SOPs (be concerned if an avalanche of change requests has appeared!)
  - Competency assessment scores for recipients of training
- User support
  - Reduction in number of enquiries received from users
  - Number of outstanding responses/resolutions to user enquiries decreasing
  - Monitor upheld change requests

APPENDIX 11E
EXAMPLE CONTENTS FOR A VALIDATION REPORT

Introduction

- Author/organization
- Authority
- Purpose
- Relationship with other documents (e.g., Validation Plans)
- Contractual status of document

System Description

- Confirmation of the identification of the system scope and boundaries (e.g., hardware, software, operating system, network)
- Confirm constraints and assumptions, exclusions and justifications

Validation Determination

- Confirm rationale behind validation requirement (may be reference to Validation Determination Statement)
- Confirm rationale updated as necessary to address any changes in system scope

Validation Life Cycle

- Confirm completion of life cycle phase by phase
  - Identification of Specification documentation
  - Summary of key findings and corrective actions from the Design Review
  - Summary of key findings and corrective actions from the Source Code Review.
  - Summary of Test Results including any Test Failures with corrective actions from Test Reports. Summary should cover IQ, OQ, and PQ
  - Confirmation that all Operation and Maintenance Prerequisites are in place
- Review Project Compliance Issues Log and satisfactory resolution of items

Role and Responsibilities

- Review any role changes
- Provide additional CVs (qualifications and experience) as appropriate

Procedures and Training

- Confirm training in SOPs delivered
- Confirm Training Records updated

Document Review and Approvals

- Lists all validation documentation produced that should be readily available for inspection
- Identify RTM where developed
- Confirm project document archive arrangements

Supplier and Subcontractor Management

- Summary of key findings and corrective actions from any Supplier Audit Reports
- Summary of key findings and corrective actions from any Predelivery Inspections

Support Program for Maintaining Validation

- Description of how the validation status will be maintained

Conclusion

- A clear statement that the Validation Plan has been successfully executed with a review of any outstanding actions or restrictions on use of system; all deviations from the Validation Plan must be justified or resolved

References

Appendices

- Glossary
- Others

12 Operation and Maintenance

CONTENTS

The operation and maintenance of computer systems can be far more demanding than system development. Over the lifetime of a computer system, more money and effort are typically put into operation and maintenance than the original project implementation, and good maintenance can substantially extend the useful life of what are more and more expensive assets. Consequently, the operation and maintenance of computer systems should be a high profile role. Pharmaceutical and healthcare companies who ignore this are more likely to be forced to replace systems earlier than they need to because their systems have degraded faster as a result of change than they needed to. Degrading system documentation and functionality will also affect the ongoing level of compliance.

This chapter reviews key operation and maintenance activities from a quality and compliance perspective:

- Performance monitoring
- Repair and preventative maintenance
- Upgrades, bug fixes, and patches
- Data maintenance
- Backup and restoration
- Archive and retrieval
- Business continuity planning
- Security
- Contracts and Service Level Agreements (SLAs)
- User procedures
- Periodic review and revalidation

Reliable operation does not indicate that a computer system is compliant, although such evidence can be used to support validation. Regulatory authorities uncovering operational issues concerning a computer system during an inspection are likely to follow up with a detailed inspection of system validation. Such inspections are often referred to as "for cause" and are discussed in detail in Chapter 16.

PERFORMANCE MONITORING

The performance of computer systems should be monitored to establish evidence that they deliver service levels required. The intent is also to anticipate any performance problems and initiate corrective action as appropriate. Performance monitoring can be seen as an extension to process performance qualification. A key step is the identification of appropriate performance parameters to monitor.

Performance Parameters

Depending on the risks associated with an application, the type of computer systems, and the operating environment, the following system conditions might be checked:

Servers/Workstations/PCs

- CPU utilization
- Cache memory utilization
- Disk capacity utilization
- Interactive response time
- Number of transactions per time unit
- Average job waiting time
- Print queue times
- I/O load
- System alarm/error messages
- Condition/readiness of business continuity measures
- Trip count for Uninterruptable Power Supplies (UPS)

Network

- Availability of components (e.g., server and routers)
- Network loading (e.g., number of collisions)

Applications

- Monitoring application error/alarm messages
- Response times

Procedures should exist which describe monitoring activities, data collection, and analysis. Operational observations are typically recorded in logbooks with the time and date, comment, and signature of the person making the observation. Some logbooks also have entries noting any corrective action (perhaps the reference to a change request) against the observation.

Statistical analysis such as Statistical Process Control (SPC) may be used to derive performance parameters as well as track and trend for alert/alarm conditions. Automated monitoring tools may be available to assist in the collection of relevant data. A record of any such tools used should be maintained and any validation requirements considered.

Status Notification

The notification requirements of out-of-specification results will vary depending on the criticality of the deviation. Some deviations may need immediate attention such as alerts identifying the loss of availability of I/O cards or peripheral devices. Other observations such as the above-recommended disk utilization will gather information to be used by periodic reviews. All parameter deviations should be diagnosed and any corrective action progressed through change control.

The mechanism employed to notify the status of monitored parameters should be carefully considered. The timeliness of communication should be commensurate with the degree of GxP risk particular parameters pose. All deviations on GxP parameters affecting product quality must be reported to QA. Example notification mechanisms include:

- Audible or visual alarms
- Message on the system console
- Printed lists or logs
- Pager message to system operators
- E-mail to system operator
- E-mail to external services
- Periodic review

Procedures and controls must be established to ensure status notification is appropriately handled. For instance, distribution details must be maintained to ensure e-mails are received by the right people. Validation of specific notification mechanisms may be appropriate.

Monitoring Plan

A Monitoring Plan should be developed to identify parameters to be monitored, specify the warning limits, and frequency of observation. The time intervals and warning limits for monitored performance parameters must be adequate to take corrective timely action where appropriate. Regulatory expectations will be invoked when certain phrases are used to describe monitoring intervals. Frequent typically indicates hourly or daily. Regular typically indicates weekly or monthly. Periodic typically indicates quarterly, annually, or biannually.

Some firms use Reliability Centered Maintenance (RCM) as part of their preventative maintenance strategy.

The GAMP 4 Guide[1] suggests a tabular format for Monitoring Plans (see Table 12.1). The structure of the table includes identification of the monitored parameter with warning limit, frequency of observation, monitoring tool, notification mechanism, when and where results are documented, and the retention period for these results. Monitoring records should be maintained and retained for appropriate predefined retention periods in a safe and secure location.

Recent Inspection Findings

- No investigation was conducted to determine the cause of missing data and no corrective measures were implemented to prevent the reoccurrence of this event. [FDA Warning Letter, 1999]
- Not all critical alarm reports describe the investigation, provide an assignable cause for the alarm, or describe the corrective actions are performed, conclusions and final recommendations. [FDA 483, 2001]
- No corrective/preventative action taken to prevent software errors due to the buildup of temporary files in computers used to control *[computer system]*. [FDA 483, 2001]
- No controls or corrective action after frequent XXXX software errors causing computer lockup. [FDA 483, 2001]

TABLE 12.1
Example of Monitoring Plan for Server-Based LIMS

Monitored Parameter	Warning Limit	Frequency of Observation	Monitoring Tool	Notification Mechanism	Where Monitoring Records Are Documented	Retention Period
CPU Utilization	Average over 25% in 24-h period	Every 10 min	System procedure	System console	File with 24-h CPU statistics	6 months
Disk Filling Grade	Over 90%	Hourly	System procedures	E-mail to system operator	E-mail directory	30 days
System Error Message	Error count increased by severe system error (defined in the tool)	Every second	Tool “CheckSys”	Message to operator pager with error number	According to SOP “Problem Management”	According to appropriate GxP regulations
Critical Batch Jobs • All Monitor Jobs • Fullbackup.com • Dircheck.com • Check print_queues.com • Stop_database.com • LIMS	If batch job is lost	Every 10 min	System procedure	E-mail to system operator Automatic restart of batch jobs	E-mail directory	30 days
Critical Processes • LIMS • Pathworks • Oracle • Perfect Disk • UCX • DECnet • Security Audit	If process is not running	Every minute	Tool “CheckSys”	E-mail to system operator	E-mail directory	30 days

- Personnel will receive their XXXX via an e-mail that has been sent from an e-mail distribution list. The firm has failed to implement controls to document that these distribution lists are maintained and updated with the current approved list of users. [FDA 483, 2001]
- A computer terminal used in the production area for XXXX was observed to be operating constantly in alarm mode. [FDA Warning Letter, 1999]
- Trending or systems perspective analysis of XXXX for XXXX is not being performed. [FDA Warning Letter, 1999]

REPAIR AND PREVENTATIVE MAINTENANCE

Routine repair and maintenance activities should be embodied in approved SOPs. Instrumentation, computer hardware elements, and communication network components should all be covered. The following areas should be addressed:

- Scheduling Maintenance
- Scheduling Calibration
- Recommended Spares Holding
- Documentation

SCHEDULING

The frequency of maintenance should be defined in these SOPs and, unless otherwise justified, should comply with the OEM's recommendations. Maintenance frequencies may be determined by recalibration requirements and reliability-centered preventive maintenance calculations. Advice can be sought from supplier organizations, but it should not be solely relied on because it is highly unlikely that they fully understand the precise nature of the pharmaceutical or healthcare application. Justifications for periodic inspection intervals should be recorded, remembering that they can be modified in the light of operational experience. Any change to recalibrations periods or preventive maintenance intervals, however, must be controlled.

Repair and maintenance operations should not present any hazard to the pharmaceutical or healthcare product.[2] Defective elements of computer systems (including instrumentation and analytical laboratory equipment) should, if possible, be removed from their place of use (production area or laboratory bench), or at least be clearly labeled as defective. It is unlikely, unfortunately, that the precise time of failure will be known. This often leaves operations staff with a dilemma of what to do with the drug products that might or might not have been made when the computer system was defective. Indeed, was there an initial partial failure and a period of degraded operation before any fault was recognized? No specific guidance can be given except to consider the merits of each situation, case by case, and ensure that a quality check is performed on product batches made during the period when the computer system is suspected of malfunction or failure. It is best to play safe when considering the number of product batches that are potentially substandard and assume worst-case scenarios.

CALIBRATION

Calibrated equipment should be labeled at the time of each calibration with the date of the calibration and next calibration due date. This label facilitates a visual inspection of equipment to check whether it is approaching its next calibration date or is overdue. The label should also include as a minimum the initials of the engineer who conducted the calibration. Some companies also include space for a full signature and printed name, but this should not prove necessary if initials are legible and can be traced to the appropriate engineer. Where labels are used, however, care must be taken to apply them to a clean dry area so that they do not fall off. Labels should be considered aides-mémoire, with the master record being kept elsewhere (perhaps handwritten in a plant logbook or a calibration certificate in an engineering management system) in case the labels become detached. Calibration procedures must be agreed on and wherever appropriate must refer to national calibration standards.

The *GAMP Good Practice Guide for Calibration Management*[3] adds the following regulatory expectations:

- Each instrument should have a permanent master history record.
- All instrumentation should be assigned and tagged with a unique number.
- The calibration method should be defined in approved procedures.
- Calibration frequency and process limits should be defined for each instrument.
- There should be a means of readily determining the calibration status of instrumentation.
- Calibration records should be maintained.
- Calibration measuring standards should be more accurate than the required accuracy of the equipment being calibrated.
- Each measuring standard should be traceable to a nationally, or internationally, recognized standard where one exists.
- All instruments used should be fit for purpose.
- There should be documentary evidence that all personnel involved in the calibration process are trained and competent.
- A documented change management process should be established.
- Electronic systems used to manage calibration should fulfill appropriate electronic record/signature requirements.

A nonconformance investigation should be conducted when a product quality-critical instrument is found out of calibration or fails a recalibration. The investigation process should include the following steps:[3]

- Previous calibration labels/tags should be removed where applicable.
- An "out of calibration" label should be attached to the instrument.
- The failure of the instrument should be logged and this information made readily available.
- A nonconformance report should be raised for the failed instrument before any adjustments are made.
- The action to repair, adjust, or replace the instrument should be followed by a complete calibration.
- The QA department should be informed to investigate the potential need for return or recall of manufactured/packaged product.
- The nonconformance report should be completed, approved, filed, and made retrievable for future reference.

Spares Holding

A review should be conducted on the ready availability of spare parts. The availability of some spare parts may be restricted. Special arrangements should be considered if alternative ways of working are not possible while a computer system awaits repair. There may be a link here to Business Continuity Planning, discussed later in this chapter.

Spare parts should be stored in accordance with manufacturer recommendations. Model numbers should be clearly identified on spare parts. Version numbers for spare parts containing software or firmware should also be recorded so that the correct part is retrieved when required.

Care should be taken when considering the use of equivalent parts for superseded items. The assumption that the change is “like for like” is not always valid. A medical device company operating in the U.K., for instance, once bought replacement CPU boards for its legacy analytical computer systems. The original boards had a 50-Hz clock but the replacements came from the U.S. with a 60-Hz clock. Unfortunately, it was a time-critical application and the problem was only discovered after a computer system had been repaired and put back into operation. Another medical device company in the U.S. recalled a workstation associated with a medical system because a so-called equivalent Visual Display Unit reversed the left/right perspective of medical image data. This image reversal could potentially have led to erroneous medical diagnosis. Not all “like for like” changes are as dangerous as these examples, but they do illustrate the point not to assume there will be no impact of change. Hence the recommendation that evidence of equivalence needs be collected (e.g., supplier documentation or supplementary user qualification) and retained.

Documentation

Maintenance and repair documentation may be requested during an inspection by a GMP regulator. Documentation for maintenance activities must include a description of the operations performed, who conducted the maintenance and when, and the results confirming that the maintenance work was completed satisfactorily. Calibration certificates should be retained. Repair records, meanwhile, should include a description of the problem, corrective action taken, acceptance testing criteria, and the results confirming that the repair work has restored the computer system to an operational state. Repair logbooks can be used to record nonroutine repair and maintenance work.

Records should be kept regardless of whether or not the work was conducted by a contractor service supplier. If such engineering support is provided by an external agency using its own procedures, then those procedures must be subjected to approval by the pharmaceutical or healthcare company before they are used. Repair logbooks should note visits form external staff, recording their names, the date, and the summary of work conducted so that additional information held by the supplier can be traced in the future if necessary. It is important that service arrangements defining when suppliers are used by pharmaceutical or healthcare companies to conduct maintenance and repair work are formally agreed upon. Such agreements are often embedded in contracts called Service Level Agreements (SLAs). The GAMP Forum promotes the development of a Maintenance Plan to define roles and responsibilities.[1] It is unacceptable to the GMP regulatory authorities not to have documentary evidence demonstrating management control of these activities.

Recent Inspection Findings

- No documented maintenance procedures. [FDA 483, 2002]
- Failure to perform/maintain computer validation in that there was no documentation to show if problems were experienced during the process, and how they were solved. [FDA Warning Letter, 2000]
- No calibration was performed prior to [system] use. [FDA Warning Letter, 2000]
- Your firm does not have a quality assurance program in place to calibrate and maintain … equipment according to manufacturer’s specifications. [FDA Warning Letter, 2000]
- Calibration records not displayed on or near equipment and not readily available. [FDA 483, 2001]

UPGRADES, BUG FIXES, AND PATCHES

This section concentrates on software upgrades, bug fixes, and patches. It is important to appreciate some basic practicalities of what happens in real life when considering compliance activities.

Why Upgrade?

When upgrading software it is prudent to establish why the upgrade is necessary. Practitioners usually cite one or more of the reasons below:

- Vendors do not support earlier version.
- Upgrading establishes common operating environment between new and existing systems.
- Are you hoping the upgrade will fix bugs in the existing product you have already bought?
- Are you wanting to use new features promoted as part of the upgrade?
- Do you really need the new features offered as part of the upgrade?
- How many known bugs are associated with these new features?

User licenses can give suppliers the right to withdraw support for their products as soon as an upgrade becomes commercially available. This effectively forces users to upgrade immediately. The latest PIC/S computer validation guidance recommends that unsupported computer systems should be withdrawn from service.[4]

Most suppliers will support their respective hardware and software for at least the three latest versions. If an entirely new product supersedes an existing product, there is usually some period of grace to migrate to the new product. Some suppliers, however, have deliberately built in discontinuity into their product upgrades. This aspect should be carefully considered. Upgrading software may also necessitate upgrading hardware, disk size, and processor. Equally, upgrades to hardware may require a supporting upgrade to software.

In order to maintain a common operating environment, the existing systems need to be upgraded. The networked computer systems in many organizations are moving toward the use of a standardized desktop configuration. It can be very difficult to run two or more versions of the same software product across the network.

If the case for an upgrade is based on a new feature, then check when the new feature will be delivered. Quite often the scope of a new release is cut back to meet shipping dates. Remember too that new features will have their own bugs. Try to use market-tested software. Do not feel the urge to upgrade to be at the leading edge unless there is a compelling business case. Pharmaceutical and healthcare companies should consider waiting until the software has developed some kind of track record. A typical waiting period might be 6 months for a widely used piece of software. Where a pharmaceutical or healthcare company consciously decides to be an early adopter, then additional Development Testing and User Qualification is likely to be required to establish confidence in the software.

Bug Fixes and Patches

Software firms knowingly release their products with residual bugs. Remember that is it not practical to test every single aspect of the software's design (see Chapter 9). Patches to large software products like MRP II may contain many hundreds of bug fixes. Such large patches should not come as surprise; remember that on average, commercial programs have about 14 to 17 bugs with various degrees of severity per thousand lines of software. MRP II products can have many millions of lines of code.

Programmers typically rely more on actual program code rather than documentation when trying to understand how software works in order to implement a change. It is easy to miss potential impacts of changes on seemingly unrelated areas of software when relying on the personal knowledge and understanding of individuals rather than approved design documents. Programmers also often take the opportunity when making a change to make further modifications that are not specifically authorized or defined in advance. Not too surprisingly, up to one in five bug fixes in complex software can lead to the introduction of a further new bug, the so-called software death cycle.

The adoption of good practices such as those defined by GxP validation should improve software quality. Original document sets need to be reviewed after a number of changes have been implemented to see if a new baseline set of documents needs to be generated.

Pharmaceutical and healthcare companies should evaluate whether or not to immediately take a patch when it first becomes available. Patches should only be taken if they support the bug fixes needed. Unless there is a driving operational requirement to apply the patch, it is recommended that companies wait and evaluate the experience of other firms applying the patch just in case the patch includes new bugs that make the situation worse rather than better. It may also be more effective to implement a number of patches together rather than individually.

Major upgrades may be required to implement specific bug fixes. Upgrades tend to be feature-focused, not quality-focused, in an attempt to attract new users. If a specific bug fix is required, check that it will be included; if it is critical to many customers, there is a good chance it will have been addressed. Suppliers typically prioritize bugs, especially for large applications, in an attempt to fix all critical bugs for a new release.

Installation and Validation

When a major upgrade is being planned it is worthwhile considering bringing forward the next scheduled periodic review to determine whether any revalidation can be combined with the upgrade effort. Revalidation is discussed in detail later in this chapter. Patches and bug fixes, meanwhile, are typically managed based on a Change Control and an Installation Qualification (IQ). In either case the scope of change needs to be understood prior to installation and validation. Supplier release notes should be consulted.

Some Operational Qualification (OQ) activity may be required to verify the upgrade — confirming that old and new functionality are available and that they work. In addition to directly testing the change, sufficient regression testing should be conducted to demonstrate that the portions of the system not involved in the change were not adversely impacted. Existing OQ test scripts may be suitable for reuse with the savings that it brings. The amount of OQ testing will depend on the complexity and criticality of the computer system and the supplier's own release management of the new version. If the supplier has conducted rigorous testing, then the pharmaceutical and healthcare company's OQ can be limited to a selection of functional tests confirming key operations. Do not assume, however, that supplier activities have been conducted and suitably documented without supporting evidence (e.g., from an earlier Supplier Audit).

Before installing an upgrade, patch, or bug fix a backout strategy should be defined with approved procedures as appropriate. If the installation is in trouble, users will be keen to return to the original computer system while the upgrade, patch, or bug fix is reevaluated. It is often not practical to rollback and reinstall the original hardware or software once an upgrade has been conducted, even when the upgrade brings severe problems. The cost to an organization rolling back to an original installation often far outweighs the money back for the purchase price of the upgrade. The message is clear in regard to implementing upgrades: do not implement automatically; look before you leap.

Upgrade Considerations

When deciding whether or not to upgrade it is important to take account of the following issues:

- New version functionality should be downward compatible with the previous version(s).
- New versions should be able to process data migrated from the previous version(s).

Suppliers usually make sure their products are backward compatible so that legacy systems can be seamlessly replaced by new systems. Suppliers typically develop their upgrades for use on the same hardware platform. Full compatibility, however, is more than this. The new product must have all the functionality that the old product had. New functions can be added, but previous functions must not be removed. For example, newer versions of word processing software typically can read formatted text documents written on older versions.

With every software upgrade, either of the application or an operating system, the validity of previously recorded data files should also be checked. This can be achieved by comparing the data derived from a legacy system with the data derived from the system upgrade.

Beta Software

Many software vendors distribute early versions of their software, called beta versions, usually free of charge to interested customers. This software is still under test and must not be used to support regulated pharmaceutical and healthcare operations. Users of beta software are supposed to help the software vendor by reporting bugs they discover. The software vendor makes no promises to fix user-discovered bugs before final release of the product concerned. For the likes of Microsoft it has been suggested that 90% of the bugs reported against beta software are already known by the vendor. Cynics have suggested that beta testing is a marketing ploy to make potential customers think of themselves as stakeholders in the success of the new product release. No formal testing is done by 15% of software firms; instead, they rely entirely on beta testing before releasing their products to market.

Emergency Changes

Exceptional circumstances may require changes to be made very rapidly (e.g., deployment of new virus protection software). Due to time constraints at the time when the emergency change is made, it may be necessary to review and complete documentation retrospectively and therefore proceed while accepting a degree of risk. If emergency changes are allowed to occur in this way, the process must be defined in an approved procedure. The use of this procedure should be monitored to ensure it is not abused by being deployed for nonemergency changes.

Figure 12.1 depicts the so-called emergency change process in which changes are made to software; it is recompiled and deployed into use before associated documentation (detailed design and, where appropriate, functional specifications) are updated. Testing is often not conducted to preapproved test specifications; rather, test reports rely entirely on collating supporting evidence generated and observations made during testing.

Wherever possible the emergency change scenarios should be avoided, but in the real world, emergency changes cannot be completely irradiated. In an emergency situation there is but one thing that matters: getting the system up and running as soon as possible. The structure of the software can degrade quickly as fix is made upon fix because of the resulting increased complexity and lag in system documentation catching up with emergency changes. If emergency changes are not managed properly, future maintenance becomes more and more difficult. If it is used at all, preventative maintenance activities should be planned to repair any structural degradation incurred.

Availability of Software and Reference Documentation

All custom (bespoke) software source code must be available for regulatory inspection (e.g., OECD recommendation in the *GLP Consensus Document*[5]). Relevant COTS product reference documentation should also be available for inspection, recognizing that proprietary COTS source code is usually only available at the supplier's premises, and access may not be available for regulatory inspection.

Copies of retained software must be stored in safe and secure areas, protected within fireproof safes. Where access to software is restricted, formal access agreements should be established, e.g.,

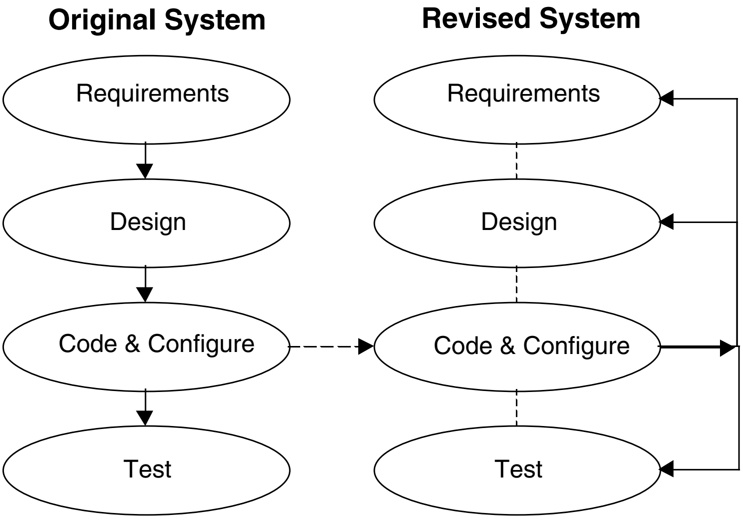


FIGURE 12.1 Emergency Change Process.

escrow accounts. Responsibility for the maintenance of the copied software and keeping reference documentation up to date, as well as its duration of storage, must be agreed upon.

Prioritizing Changes

The risk assessment process presented in Chapter 8 can be used to assist scheduling change requests. Without such an approach, prioritizing changes can become a cumbersome activity, and in extreme circumstances use vital resources that would be better focused on implementing change. Care must be taken when applying the risk assessment process because the data associated with a change could alter whether or not its associated function is critical. For instance, using an active ingredient without an associated batch number is more significant than using a pencil without an associated batch number.

Recent Inspection Findings

- After software version XXXXXX was loaded, *[it]* was not tested to assure that the essential functions would properly operate. [FDA Warning Letter, 1999]
- The firm did not monitor and keep track of changes to hardware, application, or operating system software. [FDA 483, 1999]
- The version software change was not properly validated prior to its use. [FDA Warning Letter, 1999]
- The program was not controlled by revision numbers to discriminate one revision from another. [FDA Warning Letter, 2001]
- The … program has undergone six code modifications. Each of these code modifications was implemented after a Software and Test Case Review Checklist was completed … However, none of these six code reviews detected the … problem … which led to the current recall. [FDA Warning Letter, 1998]
- There were no written Standard Operating Procedures for hardware and software change control and software revision control. [FDA 2001]
- Although the firm has in place change control for program code changes, the Quality Unit has failed to put in place procedures to ensure that the system design control documentation XXXX is updated as appropriate when program code changes have been

made. Design control documentation has not been updated since the initial release *[3 years ago].* [FDA 483, 2002]

- There was no validation data to show that the data acquisition system gave accurate and reliable results after the firm made several hardware and software upgrades. [FDA 483]
- The firm did not keep track of changes to operating system. [FDA 483]
- Software used "out of the box" without deviation report or investigation into configuration error. [FDA 483, 2002]

DATA MAINTENANCE

Data Life Cycle

Data maintenance is required throughout the data life cycle (see Figure 12.2, based on GERM[7]). Data may be captured by a manual or automated input. User procedures are required for manual data input and their effectiveness should be audited. Software supporting automated data input such as that used for data acquisition by instrumentation or data migration tools requires validation. Checks should include confirming any necessary calibration has been conducted and if interfaces are working correctly as validated.

It is important to appreciate that some data may be transient and will never be stored to durable media, while other transient data may be processed to derive data before being stored. Both transient and stored data must be protected from unauthorized, inadvertent, or malicious modification. It is expected that a register of authorized users, identification codes, and scope of authority of individuals to input or change data is maintained. Some computer systems "lock-down" data, denying all write-access. Security arrangement is discussed in detail elsewhere in this chapter.

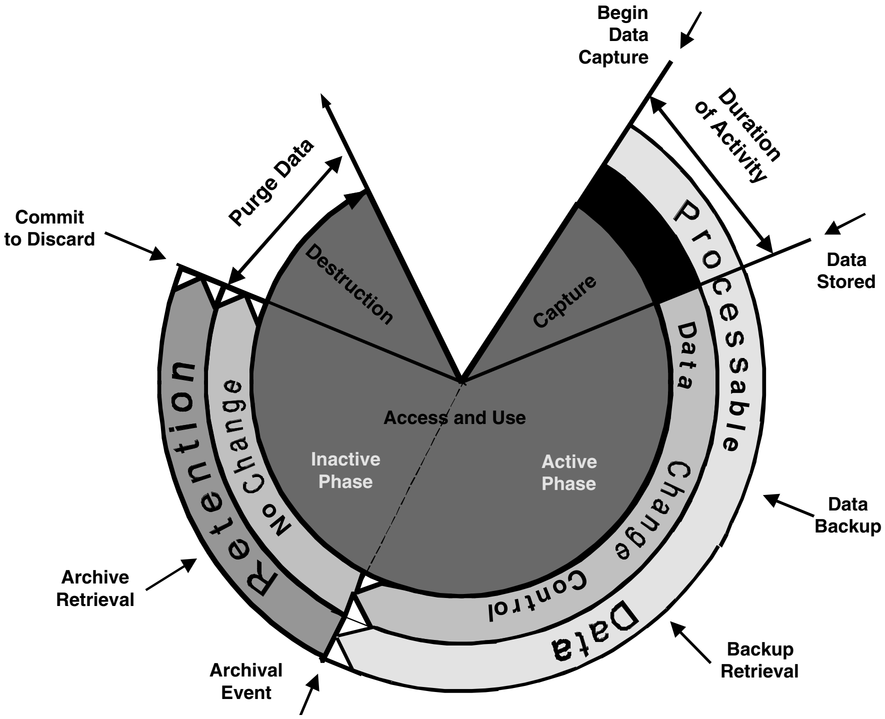


FIGURE 12.2 Data Life Cycle.

Authorized changes to stored data must be managed under change control. Data changes should be approved before they are implemented and data entry checked to confirm accuracy. Some regulatory authorities require a second verifying check for critical data entry and changes. Examples of data requiring such a second check include manufacturing formula and laboratory data. The second check may be conducted by an authorized person with logged name and identification, with timestamp, via a computer keyboard. For other computer systems featuring direct data capture linked to databases and intelligent peripherals (e.g., in a dispensary), the second check may be part of the validated computer system functionality.[4] Built-in checks might include boundary checks that data are within valid range, or authority checks to verify that the person making the change has specific authority to do so for the data item concerned.

Periodic backups may be required to avoid memory shortages and degraded performance. Restoration processes need to be verified as part of validation. Backup and restoration routines may also be used to support archiving and retrieval of data. Before archiving is undertaken, it is important to consider where data need to be retained and if so for how long. The aim should be only to keep critical data and to discard and purge the rest when no longer needed to support the operation of the computer system. Periodic data archiving requirements should be scheduled and conducted in accordance with defined procedures. Archiving and retrieval requirements are discussed in detail later in this chapter.

Audit Trails

Audit trail information supporting change control records should be maintained with or as part of their respective change control records. Audit trail information should include who made the data change, nature of the change, and date/time the change was made. Audit trail information may be maintained in paper, electronic, or hybrid form. Whatever medium is chosen, audit trail information must be preserved in conjunction with their corresponding data. Security arrangements should be equivalent to those protecting master data. Audit trails should be available in human readable form for the purpose of inspection.

Retention of Raw Data

Raw data must be retained for a period of time as defined by GxP requirements. Data may be migrated for storage to another system as long as accurate and complete copies are maintained and the transfer process has been validated. Raw data should only be disposed of or destroyed in accordance with defined procedures and authorization from local management.

Recent Inspection Findings

- Your firm has no SOP for maintaining data. [FDA Warning Letter, 2000]
- No control over changes operators can make to processing data. [FDA 483, 2002]
- Firm failed to maintain all laboratory original data … even though this option was available. [FDA 483, 2001]
- Failure to have appropriate controls over computer or related systems to assure that changes in records are instituted only by authorized personnel. [FDA Warning Letter, 2000]
- The [system] audit trail switch was intentionally disabled, and prevented the act of recording analytical data that was modified or edited. [FDA 483, 1999]
- There were no restrictions on who could create, rename, or delete data. [FDA 483, 1999]
- Audit trails not maintained for raw data files. [FDA 483, 2002]

- There was a lack of a secure system to prevent unauthorized entry in restricted data systems. Data edit authorizations were available to all unauthorized users, not only the system administrator. [FDA Warning Letter, 2000]
- The software does not secure data from alterations, losses, or erasures. The software allows for overwriting of original data. [FDA Warning Letter, 1999]
- When the capacity of the floppy disk is filled, the original data is not retained as a permanent record. Rather, the data on the floppy disk is overwritten and/or deleted. [FDA 483, 2001]
- Files corresponding to missing data were routinely deleted from the hard-drive and were not backed up. [FDA Warning Letter, 2000]
- Records did not contain documentation of second individual's review and verification of the original data. [FDA Warning Letter, 2000]
- The equipment's computer used for filling operations, which retains equipment errors that occur during filling operations, lacked the capacity to retain electronic data. After every 15th filling operation, the information was overwritten due to the storage capacity of the equipment's hard drive. [FDA Warning Letter, 2001]
- The firm did not have sufficient security controls in place to prevent [users] from editing or modifying data. [FDA 483, 1999]
- Failure to establish appropriate procedures to assure that computerized processing control systems and data storage systems are secure and managed to assure integrity of processes and data that could affect conformance to specifications. [FDA, 2001]
- No record to document that the Quality Unit reviews process operation data in computer system's data historian. [FDA 483, 2001]
- No procedure detailing file management for files stored/retrieved from network server. [FDA 483, 2001]
- No procedure governing XXXX data file management for file stored on server. [FDA 483, 2001]
- Raw data was not properly recorded or reviewed, changes in raw data were not initialed or dated. [FDA Warning Letter, 2000]
- Corrections to raw data were noted to be obscured with white correction fluid or improperly voided (no initials, date, reason or explanation of change). [FDA Warning Letter, 2000]
- Raw data was lost. [FDA Warning Letter, 2000]
- Data … [*from microbiological testing*] was entered into the Laboratory Information Management System (LIMS) prior to the documented review of the data. This is a concern to us especially because our investigators observed the Responsible Pharmacist releasing product based only on the computer data. Therefore, it is conceivable that product is released to the market prior to a second review of the raw data. [FDA Warning Letter, 1999]
- Your current practice of submitting [floppy] disks to different contractors and receiving [floppy] disks from various locations does not address how an audit trail was maintained. [FDA Warning Letter, 1999]
- There has been no formal evaluation performed in order to assure that the measurements that are printed as the permanent record is an accurate reflection of the data obtained via the floppy disk. [FDA 483, 2001]

BACKUPS AND RESTORATION

GxP regulations require pharmaceutical and healthcare companies to maintain backups of software programs including configuration, data input, and operational data in accordance with defined procedures. Installation disks for COTS software should also be kept for backup purposes. Backups

TABLE 12.2
Backup and Restoration Options

Strategy	Description	Pros	Cons	Cost
Traditional backup to tape	Manual process of copying data from hard disk to tape and transporting to secure facility	Simple-to-implement technology, multiple price-point devices/software available	Manual transportation and storage prone to risk and error; potentially long lead-time to restoration; not always practical given available "windows" of processing time	Low
Backup to electronic tape vault	Copying data from disk to a remote tape system via a WAN link	Data is accessible in shorter timeframe, services becoming standardized, WAN link process falling, and exposure to risk/errors in manual methods reduced	WAN links can introduce latency into backup process; depending on vault provider, storage may be difficult to restore; data restoration times potentially lengthy	Medium to High
Disk monitoring	Copying data written to one disk or array of disks to a second disk or array of disks via a WAN link	Instantaneous restoration of access to data possible (depending on WAN link availability and synchronicity of primary and mirrored arrays)	WAN links can introduce latency into production system operations; some mirroring systems reduce production system performance; logic errors may be replicated from original to mirrored data sets	High

provide a means of recovering computer systems and restoring GxP records from loss, corruption, physical damage, and unauthorized change. Without backups and a restoration capability, most companies cannot recover from a major disaster regardless of other preparations they have made.

Strategy

Options for backup and restoration are summarized in Table 12.2. Pros and cons must be balanced to meet the company requirements. More than one strategy for backup and restoration may be deployed as appropriate. The strategic approach to be adopted should include consideration of the following topics:

- Common policies/procedures/systems that will facilitate a consistent backup/restore approach to different applications and infrastructure can help simplify managing recovery.
- Standardized desktop configuration should reduce the variability to be managed during recovery.
- Adopting a thin client computing architecture concentrates recovery processes on a few key servers, thus reducing overall workload and numbers of personnel involved.
- WORM media (write-once, read-many) offers high security and integrity for backups.

Scheduling

The scheduling requirements for different computer systems will vary and the needs of individual systems must be assessed. Many organizations perform backups at intervals of between 1 and 60 days, although the frequency will vary depending on the criticality of the computer system, rate of change affecting the computer system, and the longevity of the associated storage media. A register of backup activity for each computer system must be kept. It is strongly recommended that backup activities are automated through networked storage devices.

Procedure

A procedure should be established for conducting backups and restoration. The procedure should cover:

- Type of backup: full or incremental
- Frequency of backup (daily, weekly, or monthly, depending on the computer system concerned)
- Number of separate backup copies (usually two, one stored remotely)
- Labeling of storage media with backup reference
- Storage location for backups (local, and remote if critical)
- Number of backup generations retained
- Documentation (electronic or paper) to be retained to provide a history of the backups and restorations for the live system
- Recycling of storage media for reuse

It is generally recommended that three backup copies are kept, one for each of the last three backups. This system is sometimes referred to as grandfather–father–son backups. Each backup should be verified before it is stored in a secure location,[8] preferably a fireproof safe. Environmental controls in storage area should be carefully considered to avoid unnecessary degradation of backup media as a consequence of excessive heat, cold, and humidity.

Any change to the backup procedure must be carefully considered and any necessary reciprocal modification to the restoration procedures made. There have been several instances where incorrect backup procedures have not been tested and subsequently backups could not be restored.

Storage Media

The appropriate backup media can vary; examples include diskettes, cartridge tapes, removable disk cartridges, or remote-networked host computers. The retention responsibilities for backups are the same as for other documentation and records. Stored backups should be checked for accessibility, durability, and accuracy at a frequency appropriate for the storage medium.[2,9] Beware of wear-out of media when purposely overwritten for reuse. Different media have different life spans. CD-ROMs, for instance, typically have a 10-year lifetime but tapes have a much shorter lifetime.

Recent Inspection Findings

- There were no written Standard Operating Procedures for backup. [FDA 2001]
- There is no established written procedure that describes the steps taken to backup the XXXX disks to ensure data recovery in the event of disk loss or file corruption. [FDA 483, 2002]
- Backup tapes were never restored and verified. [FDA 483, 1999]
- Backup tapes were stored off-site in an employee's home. [FDA 483, 1999]
- There was no documentation to demonstrate that the WAN was capable of properly performing backup and recovery of data. [FDA 483, 1999]
- Firm's procedures did not specify the frequency of backing up raw data files. [FDA 483, 2002]
- Data cannot be backed up due to a malfunctioning floppy drive. [FDA 483, 2003]

ARCHIVING AND RETRIEVAL

Archiving should not be confused with taking backups. Backups of data and software can be loaded to return the computer system back to a known operational state. Backups are usually taken on a daily or weekly basis and backup copies retained for a number of months. In contrast, archive records need to be accessible for a number of years, perhaps to people who were not involved in any way with their generation.

Archiving Requirements

GxP data, records, and documentation including computer validation should be archived. Internal audit reports from self-inspections monitoring a pharmaceutical or healthcare company's compliance with its own quality management system do not have to be retained once corrective actions have been completed, so long as evidence of those corrective actions is kept (e.g., change control records). Supplier audit reports and periodic reviews are not internal audits and should be retained.

The integrity of archived records is dependent on the validation of the systems from which they were taken and the validation of systems used for archiving and retention of those records. Chapter 13 and Chapter 15 discuss special requirements for regulated electronic records/signatures and long-term archiving solutions, respectively. Standard Operating Procedures for archiving and retrieval of software and data must be specified, tested, and approved before the computer system is approved for use.

Retention Requirements

Retention periods for data, records, and documentation are the same regardless of the medium (electronic or paper).[9] R&D records should be generally archived for 30 years although in specific circumstances longer periods may be appropriate. The retention time for validation documentation relating to a drug product's manufacture is as at least 1 year after the product's expiry date. The retention time for validation documentation relating to a drug product exempted from expiry dates varies depending on whether it is supplied to the U.S. or to Europe. For the U.S., it is at least 3 years after the last batch has been distributed,[9] while for Europe documentation must be retained for at least 5 years from its certification.[2] The U.K.'s IQA Pharmaceutical Quality Group suggests that all documentation be retained for a period of at least 5 years from the last date of supply.[10] An effective solution for many organizations has been to store their documents for a period of 7 years after the effective expiry date of a drug product or as long as the computer system is used, whichever is longer.

Storage Requirements

Archives, like backups, should be stored at a separate and secure location.[2] Critical documentation, records, and data should be kept in a fireproof safe. In some cases it is acceptable to print copies of electronic records for archiving, but advice should be sought from regulatory authorities. Clinical trial data are often stored on microfiche or other electronic medium. It should not be possible to alter such electronic copies so that they could be interpreted as master records.[11]

Temperature and humidity may have a bigger impact than in the case of backups because of the extended duration of storage. The storage environment should be periodically evaluated to confirm stable storage conditions exist. Environment data should be recorded and maintained. Some firms use automated monitoring systems for this purpose.

Retained media are likely to require at least one refresh during their retention period. Different media have different life spans, and manufacturer's recommended refresh intervals vary. CD ROMs for instance typically have a 10-year life span and a 5-year refresh recommendation. DAT usage should not exceed 20 times for read/write operations and are typically considered to have a 5-year life span without copy. Tapes, meanwhile, may be accessed perhaps up to 100 times but require retensioning. It is recommended that a new copy of a tape be made every 12 months. The process of data migration is discussed in Chapter 11. Data migration will be required not only as part of normal media management but also when media become obsolete during the retention period. Long-term preservation issues for archives are discussed in Chapter 13.

Retrieval Requirements

Archive information required by regulators, including those stored electronically, must be accessible at their site of use during an authorized inspection. It should be possible to give inspectors, if requested, a true paper copy (accurate and complete) of master documentation regardless of whether the original's medium was magnetic, electronic, optical, or paper within 24 h of the request. Longer retrieval periods of up to 48 h may be agreed to for information that is stored remotely from the site being inspected. True copies must be legible and properly registered as copies. Where large volumes of information are archived, the use of manual or automated supporting indexes is recommended to ease retrieval. Software applications, scripts, or queries used for manipulating or extracting data should be validated and maintained for the duration of the retention period.

It is vital that retained records are not compromised. Unlike backups that, by their nature, are routinely superseded by newer copies, archives are irreplaceable historical records. The content and meaning of archived information must not be inadvertently or maliciously changed. Consequently, access to retained records should be read-only. After each use the storage media should be given an integrity test to verify that it has not been corrupted or damaged. Logs of archive access should record media retrieved, returned, and the success of subsequent integrity testing documented. Storage media must not be misplaced or lost.

Recent Inspection Findings

- There were no written Standard Operating Procedures for archival. [FDA 2001]
- It was not demonstrated that electronic copies of XXXXXX could be stored and retrieved for the duration of the record retention period. [FDA Warning Letter, 1999]

BUSINESS CONTINUITY PLANNING

Business Continuity Plans define how significant unplanned disruption to business operations (sometimes referred to as disasters) can be managed to enable the system recovery and business to resume. Disruptions may occur as a result of loss of data or outage of all or part of the computer system's functionality. The range of circumstances causing disruption can range from accidental deletion of a single data file to the loss of an entire data center from, for instance, fire.

Business Continuity Plans are sometimes referred to as Disaster Recovery Plans or Contingency Plans. There are two basic scenarios:

- Suspend business operations until the computer system is restored.
- Use alternative means to continue business operations until the computer system is restored.

Suspending business operations may entail scrapping work in progress or continuing work in progress to completion using alternative means. It may be possible to use alternative means to support business operations for some time before final suspension awaiting restoration of the original computer system. The duration to which alternative means can be supported will depend on the overhead to operate them including the effort to retrospectively enter interim operational data into the original computer system to bring it up to date.

Procedures and Plans

Procedures and plans supporting business continuity must be specified, tested, and approved before the system is approved for use. Topics for consideration should include catastrophic hardware and software failures, fire/flood/lightning strikes, and security breaches. Procedures need to address:[8]

- Specification of the minimum replacement hardware and software requirements and their source
- Specification of the time frame within which the replacement system should be in production, based on business considerations
- Implementation of the replacement system
- Steps to revalidate the system to the required standard
- Steps to restore the data so that process activities may be resumed as soon as possible

The procedures and plans employed should be retested periodically and all relevant personnel should be aware of their existence. A copy of the procedures should be maintained off-site.

Regulators are interested in business continuity as a means of securing the supply of drug products to the user community. The requirement for Business Continuity Plans covering computer systems is defined in EU GMP Annex 11 (the FDA has similar requirements).

> *There should be available adequate alternative arrangements for systems which need to be operated in the event of a breakdown. The time to bring the alternative arrangements into use should be related to the possible urgency of the need to use them. For example, information required to effect a recall must be available at short notice. The procedures to be followed if the system breaks down should be defined and validated. Any failures and remedial actions taken should be recorded. [Clause 15 and 16, EU GMP Annex 11]*

There are seven basic tasks to be completed for business continuity planning:

- Identify assets and/or business functions that are vital to the support of critical business functions.
- Assess interdependencies between critical computer systems/applications.
- Identify vulnerable points of failure and make changes to reduce or mitigate them.
- Select recovery strategy to meet appropriate timeframes for restoration.
- Develop business continuity plan.
- Prepare procedural instructions and conduct training.
- Verify business continuity plan through verification exercise.

Major threats are identified in Table 12.3 with suggested controls to support continuity of business operations. Leading disaster scenarios in one survey were system malfunction (44%), human error (32%), software malfunction (14%), computer viruses (7%), and natural disasters (3%).[12] Plan for general disaster scenarios; it is too easy to get bogged down trying to identify every conceivable catastrophic situation. It is also important to remember that threats are relative. Water extinguishers to suppress a fire, for instance, should not be treated as bringing a new threat of water damage.

Verification is not normally possible through comprehensive testing. Some companies may claim that they can test computer systems in isolation, accepting the disruption this often involves. Testing disaster scenarios, by their nature, are catastrophic and not to be knowingly invoked. Simulation provides a much more practical approach. Simulation exercises are based on rehearsals whereby teams walk through what they would do in a disaster scenario, using procedures and possibly some support systems. Simulations can prove useful training events. The approach to verifying business continuity planning will depend on the particular opportunities and constraints affecting a company.

Redundant Systems and Commercial Hot Sites

In the event of a disaster, dedicated redundant systems at a separate locality which must be far enough distant not to have been affected by the disaster are brought on-line. Users are either relocated to the backup facility or are provided remote access to the backup system via some sort

TABLE 12.3
Threats and Controls for Business Continuity Planning

Threats	Controls
Water damage (e.g., leaky pipes and floods)	Water detection to provide early warning of leaks and other water hazards (e.g., condensation)
Fire/heat damage (e.g., arson, equipment overheating, lightning strikes)	Detection of preignition gases, smoke, and other indicators of impending fire to enable proactive response that will ensure health and safety of personnel and prevent loss of data and equipment to fire Suppression of fires (e.g., sprinkler systems, gaseous extinguishing systems, using noncombustible materials in facility, restrict storage of combustible consumables such as paper) Use fireproof cases, cabinets, and safes
Power failure	Continuity of electrical power in the presence of an electrical outage (e.g., use of an uninterruptable power supply — UPS) or surge (e.g., electrical conditioning)
Network failure	Network backup and restoration facilities at local and intersite level; restoration of communications external to company
System malfunction (software, hardware, human error)	Detection of contamination levels (dust, food and drink, production materials) that can accumulate in equipment and lead to system malfunction Monitor hours worked by individuals and/or mundane nature of work that might result in loss of concentration and hence introduction of human errors (data errors and user operation errors)
Malicious/accidental damage (e.g., hackers)	Logical firewalls and user access systems requiring combination of physical and logical password elements Physical security of corporate computing, data centers, and telecommunications facilities
Other factors (forced evacuation for environmental hazards, aircraft crashes)	Provision of and training in evacuation procedures and safe areas

of preestablished network connection. User applications typically have a target time for restoration of redundant systems and commercial hot sites of within 1 to 2 h and 7.5 h, respectively.

Besides being the most reliable method of recovery with minimal business disruption, redundancy also tends to be the most expensive. A commercial hot site, for this reason, is often a more acceptable alternative from a cost perspective, provided a slightly longer recovery window is acceptable to the business.

Service Bureaus

Some companies elect to back up systems against failure by contracting with a service bureau for emergency recovery. Essentially it is an insurance policy whereby the pharmaceutical or healthcare company leases a standby system. User terminals and printers are installed in the client offices with network connection to the service bureau that may be at the service supplier's premises or a mobile facility that is driven onto site. User applications typically have a target time to restoration within 24 h. The problem with commercial mobile facilities is that their service providers often require up to 48 h to guarantee deployment.

This approach to business continuity planning requires:

- The interdependency between critical and noncritical applications to be understood so that when the service bureau is invoked it can operate independently, or that other critical cosystems are also restored
- The most recent application versions are restored with current data

This solution can be very complex where there are several applications involved, as each application typically requires its own service bureau. Many companies are not considering the use of Internet and intranet linking to support restoration.

Backup Agreement

This approach involves a site being provided with a backup by a partner organization. This does not mandate a redundant system but more often utilization of spare computing capacity at the partner organization. User applications typically have a target time to restoration within 24 h. Practical problems include maintaining current system versions on partner organizations and finding a mutually convenient time to test the backup facility. Maintaining the partnership can be complex. Another issue is how to ensure that the partner's computer systems are not themselves brought into the disaster scenario by placing too high a demand on their computer systems when the backup is invoked.

Cold Sites

Cold sites involve preparing an alternate backup system. Company-owned cold sites have the drawback of being expensive to outfit. Such an investment can, however, be used for off-site storage and training when not activated. An alternative is to employ a commercial cold site that might be shared between a number of client companies. As with service bureaus, cold sites may be mobile facilities that are driven to a client's site. The risk with cold sites is that because they are shared it is possible that they may not be available if a disaster has already hit one of the sharing parties. User applications typically have a target time to restoration of between 24 and 72 h. Longer than 72 h typically means that the business has come to a complete stop.

Manual Ways of Working

Define manual ways of working for application during system outage. Remember that on restoration some reprocessing of data input to the original or backup system (catch up) will be required and this must be planned for. Manual records made during the outage, even once input into the restored system, must be retained.

Software Licenses

Loss of software support for aging versions of business critical systems can create significant business continuity and regulatory risks. Pharmaceutical and healthcare companies should provide a definitive statement on how they will maintain critical systems where support has historically been provided by third parties but that support is no longer available or set to expire. Measures need to be established to prevent adverse impact to product quality and product data and how they will ensure business continuity during any system outage.

The U.S. Uniform Computer Information Transaction Act gives vendors the power to deactivate software without a court order so long as this is defined in a license agreement.[1] Users are to be given 15 days' notice of any turnoff. This raises several key compliance concerns:

- Notification of software license termination: What if warnings of software termination for whatever reason go astray, the vendor may not hold the company's current address, the company's name may have changed through merger or divestment, the employee who signed the agreement may have left the company, or the employee who signed the agreement may be absent from work for holiday, birth of a child, or sickness?

- Business Continuity: While the loss of a word processing package will be generally irritating, the loss of a server might be critical if it led to the outage of a network. The effects of disabling software may not be limited to the target company and may extend through supply chains. The ability to turnoff software will not be limited by national boundaries. Key suppliers (or equipment, drug ingredients, and services) may not be able to function and fulfill their commitments to pharmaceutical and healthcare companies. Distribution and wholesale of drug products, often outsourced, may themselves be halted because of disabled software which could affect the availability of vital products to patients. Joint ventures, partnerships, and intercompany initiatives may also be in jeopardy.
- Consequential Loss: Questions have been raised if the turnoff of software led to the corruption or loss of GMP data. Pharmaceutical and healthcare companies will be forced to assign significant resources on checking licensing agreements of COTS products.
- Unauthorized Disabling of Software: Another concern is that disabling codes for potential use by the vendor could also be used by hackers.

Design features to disable software are not new. In the early 1990s a chemical manufacturer suffered the loss of an MRP system when unwittingly it failed to renew a support contact over the New Year period. The software was automatically disabled mid-January, with severe business impact. The software vendor had not escalated the license issue when there was no reply to a renewal request sent a few months earlier.

The FDA has indicated that such features may compromise management of electronic records and electronic signatures and has indicated software products with such features should not be used in validated systems.[6] Unfortunately, suppliers may insist on the right to use such features or charge a higher price to compensate for its absence. Pharmaceutical and healthcare companies should:

- Know the terms of supply for the software being used
- Write procedures, if necessary, to ensure record integrity is maintained in case the software stops functioning
- Assess how automatic restraints impact compliance and validation
- Make sure the above issues are considered when purchasing software

Recent Inspection Findings

- There were no written Standard Operating Procedures for disaster recovery. [FDA 2001]
- Following flood damage in September 1999 to your facility and equipment, you or your employees failed to evaluate the raw data storage conditions … or implement any procedures or changes to existing procedures to alleviate future damages. [FDA Warning Letter, 2000]

SECURITY

Hardware, software, and data (local and remote) should be protected against loss, corruption, and unauthorized access.[8] Physical security is required to prevent unauthorized physical access by internal and external personnel to computer system hardware. Logical security is required to prevent unauthorized access to software applications and data. The network and application software should provide access control.

Management

Standard Operating Procedures for managing security access (including adding and removing authorized users, virus management, and physical security measures) must be specified, tested, and approved before the system is approved for use. Topics to be covered include the following:

- Issue unique User-ID codes to individual users.
- Passwords should be eight characters long.[17]
- Do not share personal passwords or record them.
- Do not store information in areas that can be accessed by unauthorized persons.
- Do not download from the Internet.
- Applications are protected from viruses: virus check all floppy disks, CDs, hard disk drives, and other media from internal and external sources.
- Do not disable virus checks.
- Do not forward unofficial messages containing virus warning (may be a hoax and unnecessarily increase traffic, or may further propagate a real virus).
- E-mail over the Internet is not secure without Public Key Infrastructure (PKI).
- Do not send messages from someone else's account without authorized delegation and management controls.
- Do not buy, download, or install software through unauthorized channels.
- Do not make unauthorized copies of software or data.
- Amendments to electronic records should be clearly identified and not obscure original record.
- Use of electronic signatures is controlled.
- Electronic links used to transfer data are secure.
- Take backups of software and data.

Passwords should be securely issued to their users, ensuring that the users concerned have been authorized to access the computer systems for which the passwords are being granted. Merely issuing a User-ID and sending an e-mail to the user with the password enclosed is insufficient. It is very difficult to guarantee that unauthorized staff might have access to the e-mail or the user's account. The identity of the user should be authenticated before a password is issued. Some pharmaceutical and healthcare companies do this by verbally communicating passwords in two halves, one half to the user's line manager and the other half to the user. Neither party can use a portion of the password to gain access to a system without knowledge of the other party's portion of the password. In the process proposed, the line manager authenticates the user as authorized for the computer system concerned before giving the user the other half of the password they need.

Once users have been granted access to a computer system, it is common practice to prompt them to renew their passwords every few months (e.g., expire every 90 days for networked users). There is no formal regulatory requirement to change passwords that are still secure. Many users struggle to remember passwords that change frequently, often reverting to writing the passwords down or using passwords that can be easily memorized such as family names and vehicle license plate numbers. Some pharmaceutical and healthcare companies are looking at random alphanumeric passwords with longer expiry periods to improve overall security.[7] Such passwords by their nature are virtually impossible to guess but also harder to remember. The issue of remembering passwords is compounded when users have access to a number of computer systems each nominally having individual passwords. It can be very tempting to manage all systems to share User-IDs and associated passwords, in which case the controlling mechanism needs careful validation.

User Access (Profiles)

The rules and responsibilities for assigning access rights should be specified in procedures approved by QA. Access rights need to be documented and reviewed regularly to ensure they are appropriate. All users need to receive appropriate training about their user access privileges. Default user access should be no access. Users with changing authority levels should have their access rights modified to accurately reflect their new roles. Access rights for those who no longer are authorized to use a system should be immediately removed. Screen locks should be used to prevent unauthorized access from unattended user terminals.

Computer Viruses

The vulnerability of computer services to computer viruses is not easily managed. Besides deploying antivirus software the only other defense is to stop unauthorized software and data being loaded on computer systems and to build firewalls around networked applications. This is a prospective approach that assumes existing computer services are free from computer viruses. However, this approach cannot entirely remove the threat of computer viruses from computer services. The source of authorized software and data may itself be unknowingly infected with a computer virus. Novel viruses can also break through network firewalls. It is therefore prudent to check software and data related to computer services that are used within an organization.

The management of computer viruses is primarily based on prevention:

- Strict control of access to computer services
- Policies forbidding the use of unauthorized software
- Vigilant use of recommended antivirus software to detect infections

Procedures should be established covering:

- Stand-alone computer systems including laptops
- Client workstations
- Network servers providing file services to PC workstations
- Floppy diskettes (both 3.5 in. and 5.25 in.)
- Compact disks (CDs)
- Other removable storage media

Virus checking should be performed on all computer systems and removable storage media if:

- They originate from an external organization (including but not limited to universities or other educational establishments, research establishments, training organization, external business partners).
- Their origin is unknown (including but not limited to unsolicited receipts).
- They have been used with other computer systems or removable storage media of unknown status (including but not limited to being sent off-site for repair, maintenance or upgrade).
- They are received for demonstration, training, or testing purposes.
- They belong to a representative or employee of an external organization and are to be used in conjunction with *in situ* computer equipment.
- They were last on an external system for business, educational, training, or private purposes (including but not limited to software acquired electronically from external networks or the Internet).

Regular virus checking arrangements (sweeping) should be defined with service providers. Local instructions will be needed for users to carry out the necessary checks. It is important to understand that virus checking software only checks for known viruses. Updates to the antivirus software must be applied when available. The application of multiple antivirus software utilities may be recommended to offer higher combined detection coverage of viruses. Only vetted and approved antivirus software utilities should be used.

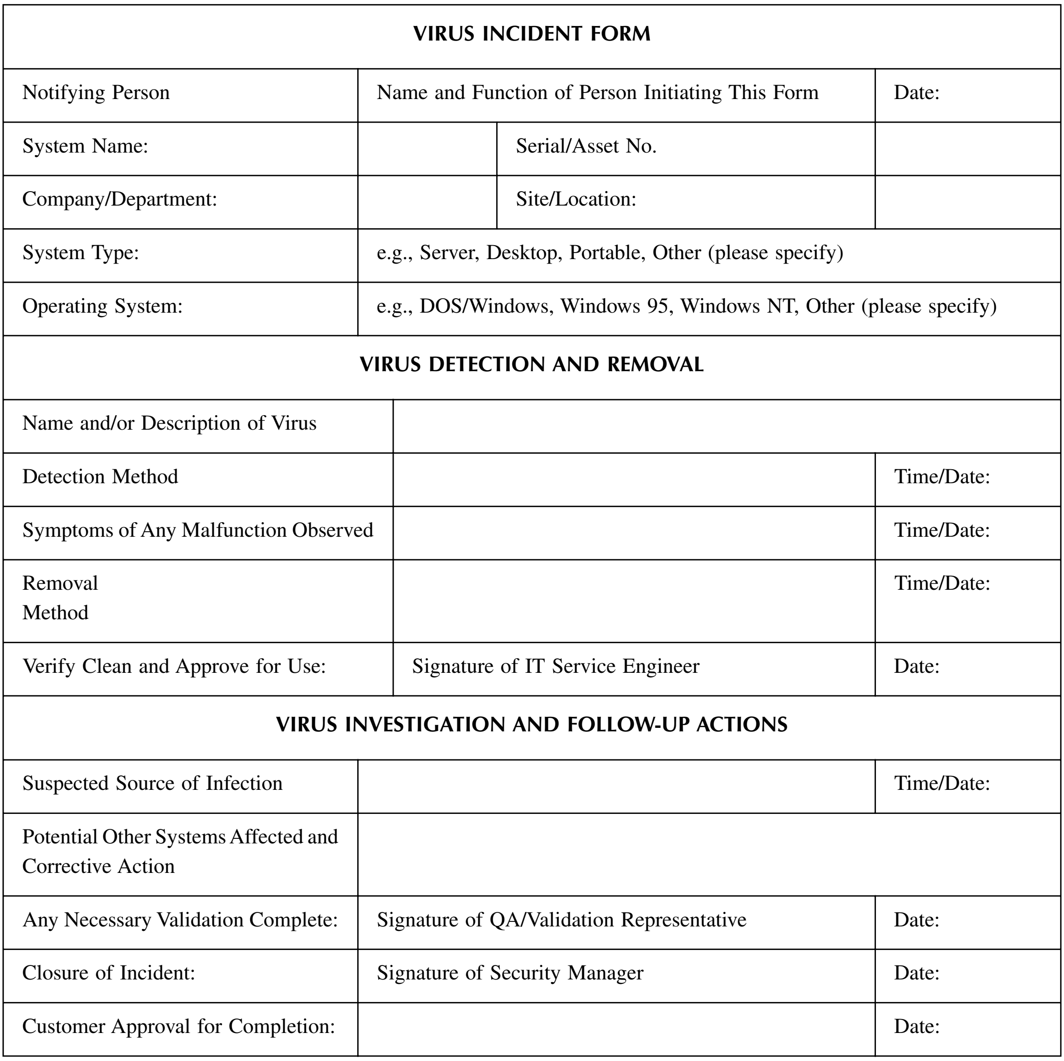
Detected computer virus should be reported so that the virus is removed and the integrity of the computer system restored. If a virus is found or suspected, then:

- No application must be run on the affected computer system. Any error or warning messages displayed must be recorded along with details of any unusual symptoms exhibited by the computer system.
- Local support staff must use their judgment as to whether or not it is safe to save data and exit any currently executing application in a controlled manner. Where it is determined that this is not safe to do, then the machine must be powered down immediately.
- Every effort must be made to find the source of the virus. The virus must be identified and instructions sought from the antivirus software documentation or elsewhere on how to remove it. Unresolved virus infections must also be noted.
- After investigation, infected removable storage media should be destroyed, but if important data is needed, the virus must be removed under the supervision of the IT support contact. Systems that may have come into contact with the diskette must be checked immediately.
- Computers must be rebooted using clean, write-protected system diagnosis disks. This will ensure that a true analysis of the computers is performed without any viruses being resident in memory. All local hard drives must be scanned. If the virus has been supplied from an external source, then that source should be noted. If no virus is detected, this should be recorded.
- Any servers that may have come into contact with the virus must also be checked immediately. Any computer system that has come into indirect contact with the infected computer system via removable storage media must also be checked.
- All deleted data files and software must be restored from backups or the original installation media. Local computer drives should be checked after restoration to verify that they are still clear of any computer viruses.
- Crisis management will be required where a computer virus has manifested itself causing a computer system malfunction. Senior management should be kept informed of the incident and corrective actions being undertaken, and the wider user community should be warned of incident to reenforce vigilance.

Deploying antivirus software without validation may be a necessity to control virus attacks or avoid anticipated attacks. Virus attacks may pose a more significant risk to GxP data than lack of validation of antivirus software. An example virus incident form is shown in Figure 12.3.

Recent Inspection Findings

- The system administrator and [users] had access privileges that enabled and disabled switches for the system configuration editor, editing permissions for fields/commands and files, and system menu functions. Functions included: read/write access, delete and purge data, modify and rename a data file, overwrite the raw data file, and copy and rename files. [FDA 483, 1999]



VIRUS INCIDENT FORM			
Notifying Person	Name and Function of Person Initiating This Form		Date:
System Name:		Serial/Asset No.	
Company/Department:		Site/Location:	
System Type:	e.g., Server, Desktop, Portable, Other (please specify)		
Operating System:	e.g., DOS/Windows, Windows 95, Windows NT, Other (please specify)		
VIRUS DETECTION AND REMOVAL			
Name and/or Description of Virus			
Detection Method			Time/Date:
Symptoms of Any Malfunction Observed			Time/Date:
Removal Method			Time/Date:
Verify Clean and Approve for Use:	Signature of IT Service Engineer		Date:
VIRUS INVESTIGATION AND FOLLOW-UP ACTIONS			
Suspected Source of Infection			Time/Date:
Potential Other Systems Affected and Corrective Action			
Any Necessary Validation Complete:	Signature of QA/Validation Representative		Date:
Closure of Incident:	Signature of Security Manager		Date:
Customer Approval for Completion:			Date:

FIGURE 12.3 Example Virus Incident Form.

- Passwords never expired and consist of four characters. [FDA 483, 1999]
- System configuration did not allow for the unintended operation of an instrument in a secure mode during processing and collection of data. [FDA 483, 1999]
- The firm has failed to establish procedures to maintain a current list of approved users with user levels of access for the XXXX system. [FDA 483, 1999]
- The computer system used to monitor and control manufacturing equipment lacked appropriate controls to ensure that only authorized personnel had access to the system. [FDA Warning Letter, 2001]
- There is no written procedure to describe the process that is used to assign, maintain passwords and access levels to the control system. [FDA 483, 2001]
- There is no written procedure to describe the security and control of XXXX floppy disks. [FDA 483, 2001]
- Failure to establish and implement computer security to assure data integrity in that during this inspection it was observed that an employee was found to have utilized

another person's computer access to enter data into the XXXX computerized record keeping system. [FDA Warning Letter, 2001]
- There is no written procedure to describe the process that is used to assign, maintain passwords and access levels to the control system. [FDA 483, 2001]
- There were no written Standard Operating Procedures for virus detection. [FDA 2001]
- There were no written security guidelines. [FDA 2001]
- There was no validation data to demonstrate that an authorized user of the corporate WAN did not have access to analytical data on the laboratory's LAN. [FDA 2001]
- The client/server password system failed to adequately ensure system and data integrity in that passwords never expired and could consist of four characters. [FDA 2001]
- Once an analyst initiated data acquisition, anyone could access the system. [FDA 2001]
- You failed to have adequate security controls for your XXXX systems because your system, once accessed by one employee, is left open and available for other personnel to gain access to the original employee's analytical test results. [FDA 483, 2002]
- There was no established written procedure that addressed the access code for the software development room and notification of team members of the changes. [FDA 483, 2002]
- Users could grant authority to themselves or any other person high-level access within the application. [FDA 483, 2001]
- The firm failed to produce an approved list of personnel currently authorized to use the *[computer system]*. [FDA 483, 2001]
- System security has not been defined. [FDA 483, 2001]
- An employee user name and computer password were publicly posted for other employees to use to access the XXXX system. [FDA Warning Letter]
- Three previous employees, who had terminated employment in 1997 and 1998, still had access to critical and limited functions on March 18, 1999. [FDA Warning Letter]
- The firm has not established any security procedures for the XXXX computer systems. [*System*] password function was disabled. [FDA 483, 2002]

CONTRACTS AND SERVICE LEVEL AGREEMENTS

Contracts should be established with all suppliers. For standard items of equipment and software this can take the form of a purchase order. For support services it is common practice for users of computer systems to establish a Service Level Agreement (SLA) with their suppliers.

SLAs should unambiguously define the system being supported, the services to be provided, and any performance measures on that service.[1] Examples of services that might be provided include:

- Developing and installing software upgrades, bug fixes, and patches
- System management and administration
- Support for underlying IT infrastructure
- Use of any particular software tools
- Routine testing and calibration

Other relevant information normally held as appendix or schedule to the SLA include user and supplier contact details, definition of fixed costs, charge-out rates, and penalty payments as appropriate. Contractual terms and conditions might also be included if not managed as a separate document. Escalation management processes should be documented and understood.

Service providers should have formal procedures in place to manage their work. They can, however, agree to use customer procedures if this is more appropriate.

Pharmaceutical and healthcare companies should reserve the right to audit use of whatever governing procedures are being used. Service providers should be audited just like other suppliers (see Chapter 7). This is especially important for system development, IT infrastructure, and maintenance activities. Audit reports should be retained and any audit points followed up as required.

Service levels should be periodically reviewed and summary reports prepared. Performance measures should be established with target minimum service levels. Responsibilities for collecting data to support performance measures should also be agreed upon along with any calculations to be used to derive performance levels. Trending topic areas may provide a useful indicator regarding emerging issues. Consideration should be given to the question of who will receive SLA reports and how often such reports are required. As a minimum, such reports should be reviewed when considering contract renewal.

Recent Inspection Findings

See Contracts of Supply in Chapter 7.

USER PROCEDURES

Experience suggests that human error accounts for up to one fifth of system malfunctions.[14] This emphasizes the importance of accurate and practical User Procedures accompanied by suitable training.

User Procedures for operating and maintaining the computer systems, control system, or laboratory system must be specified, approved, and where possible tested, before the systems are approved for use.[15] User procedures can make good use of Role Activity Diagrams (RAD) to help readers understand the specific responsibilities associated with different roles. An example RAD is shown in Figure 4.3 in Chapter 4.

Procedures should be put in place to pick up possible system errors as well as human error or misuse. It is important to track trends and demonstrate proactive management of issues. Statistical analysis should be applied to data gathered.

User procedures should be periodically reviewed and updated as necessary. Expiry dates should be clearly noted on SOPs, and should not normally exceed 3 years from date of approval of the SOP.

Recent Inspection Findings

- Despite assurances that no operator's manual was needed because the system was as easy to use as a microwave, inspectors found that the night supervisor did not know how to respond to alarms. [FDA Warning Letter, 1994]
- Failure to establish and maintain procedures for validating … design, and failure to assure … conform to defined user needs and intended uses, including testing under actual and simulated use conditions. [FDA Warning Letter, 1999]
- There were no written user standard operating procedures … [for] system validation, hardware and software change control, revalidation, user operations, security guidelines, software revision control, virus detection, disaster recovery, and backup and audit trail archival. [FDA 483, 1999]
- The computer software your firm uses … is deficient. Your procedures do not require the documentation of calculation and entry errors. [FDA Warning Letter, 2000]
- There is no established written procedure to describe the reuse of a floppy disk. [FDA 483, 2001]
- There are a number of nonapproved documents or instructions that are used by personnel, for example:
  - In the event of an alarm from the [*computer system*] the operators are to acknowledge the alarm, call or contact a designated individual.
  - There was a videotape labeled and dated in the XXXX control room.

  - "NOTICE!!! The Environmental Monitoring data files are to be accessed by Environmental Monitoring Personnel ONLY! Please ask for assistance if data is needed. THANK YOU."

    These documents do not list they have been reviewed and approved by Quality Control or *[are]* part of the officially established written procedures. [FDA 483, 2001]
- No SOP for Control Panel used to store product recipes and process parameters. [FDA 483, 2001]
- There were no written Standard Operating Procedures for user operations. [FDA 2001]
- There is no user manual for the XXXX computer system. [FDA 483, 2002]
- User manuals for applications were found referenced from currently approved procedures to provide specific details on how to perform various operations. Regarding the user manuals:
  - All user manuals are obsolete, having not been updated since 1992.
  - The outdated application user manual lacked indication of review and approval.
  - The outdated user manual lacked indication of what revision XXXX it applied to [FDA 483, 2001].
- The Quality Unit failed to put in place procedures defining the use of the *[application].* [FDA 483, 2001]

PERIODIC REVIEW

Computer systems, as critical items of equipment, should be periodically reviewed to confirm that their validated status has been sustained.[16] Validation Reports concluding the implementation of a computer system should identify when the first periodic review is expected. The selected interval for periodic review needs to be justified. Many companies conduct periodic reviews every 12 months for their most critical systems. Less critical systems do not generally warrant such regular review. It is recommended that intervals between periodic reviews do not exceed 3 years to reduce the risk of undetected deviations.

It may be possible to collectively review a number of less critical systems by the product they support (e.g., through annual product reviews) or by the physical area in which they reside (e.g., laboratory, manufacturing line). Sometimes periodic reviews combine process validation and computer validation. If either of these approaches is taken then the coverage (list of systems) must be defined for the review.

The following criteria can be used when evaluating suitable intervals between periodic reviews and the scope of review:

- Nature of use — potential impact on the quality of drug and healthcare products
- Character of system — size and complexity of the computer system, and how easily unauthorized changes can be made
- Extent of design changes — cumulative effect of changes to the computer system (including software upgrades) made since the last (re)validation exercise
- System performance including any system failures — any problems experienced with the system's operation (e.g., user help desk inquiries, system availability, access control, data accuracy)
- Changes to regulations — effect of changes made to regulatory and/or company requirements since last (re)validation exercise

Organizations often establish a review panel to conduct periodic reviews. Before the panel meets, the chairman should estimate the scope of the review, the time needed to undertake the review, and determine the size and composition of the review panel. The level of review should be based on a documented risk assessment. Members of the review panel should include operations

TABLE 12.4
Example Periodic Review Topics

Topic	Comments
Performance	Check critical process performance parameters and whether any problems are potentially due to supporting computer system.
Procedures and Training	Check training records are current. Examine the need for refresher and induction courses for new employees (permanent and temporary staff, consultants and contractors). SOPs should be reviewed on a biennial basis and hence do not require retraining within that time unless something has changed.
Change Control	Have the change control procedures been correctly adopted? Is the cumulative effect of change understood? Have company or regulatory computer validation standards changed? Does the URS adequately describe the current use of the computer system? Check what has changed with computer system instrumentation, computer hardware, and computer software. Do design documents reflect these changes? Check whether any unauthorized changes have been made. Conduct spot checks to compare running systems with documentation. Check requirements traceability to verify IQ/OQ/PQ testing covers the system as used. Review the criticality of any outstanding change requests and how long they have been outstanding.
Calibration and Maintenance	Check software copyrights and licenses. Some software applications cease to function upon expiry of a license. Check maintenance and calibration schedules. Exercise UPS batteries and check ongoing records monitoring the operating environment (e.g., humidity and temperature).
Security	Review physical access arrangements and any attempted breaches. Review accuracy of lists of active users. Review user access profiles for access rights that are no longer required. Review unauthorized access attempts.
Data Protection	Check lockdown of user access to alter data. Check audit trail of any data maintenance activities.
Backups	Verify backups and archive copies are being made and can be restored.
Business Continuity	Review any SLAs to check that details are correct, still appropriate, and that the supplier is aware of his/her obligations. Walk through contingency and disaster recovery plans to check they are still applicable.

staff and management, system support staff, and quality assurance. User-communities of the networked applications should also be represented.

The review panel meeting should only take a few hours if all the necessary information for the periodic review is collated before the meeting. Table 12.4 identifies some topics for consideration in the periodic review. The review meeting must be recorded either by minutes or a formal report. It will normally begin by reviewing progress on actions assigned at last meeting and close by assigning a new list of actions that should be assigned to individuals with target dates for completion.

A particularly important decision to make during a periodic review is whether or not revalidation is required. At a certain point in time, maintaining an old system becomes too ineffective for the expense incurred. There are no predefined metrics to base this decision on, but certain characteristics signal system/software degradation.

- Frequent system failures (partial or catastrophic)
- Significant growth in size of software modules/subroutines (possible emergence of complex system structure and spaghetti code)

- Excessive and increasing maintenance effort (possible difficulty in retaining maintenance personnel — key knowledge being lost)
- Documentation does not adequately reflect actual system (e.g., need to refer to supplementary change control records to understand system)
- Over 3 years since last (re)validation

The greater the number of such characteristics the greater the scale of potential reengineering required. In fact it may reach a stage where it is more cost-effective to entirely replace the system. Pharmaceutical and healthcare companies are encouraged to collect their own metrics to make this decision process more objective. Typically such decisions are very subjective, and care should be taken to make sure the decision is not unduly influenced by dominant personalities rather than real needs.

Occupational Health

Consideration must be given to the potential effects of the computer system and associated equipment on the personnel who may use or come into contact with the system. Typically these risks are associated with the interfacing to Visual Display Units (VDUs) and environmental conditions.

Recent Inspection Findings

- There are no provisions for periodic audits of validated computer systems.
- Require periodic review of findings by a responsible individual to assure the corrective action is effective. [FDA Warning Letter, 1998]
- Supporting documentation requirements must be defined for validation reviews. [FDA Warning Letter, 1999]
- While the individual changes have been reviewed during the change control process, a comprehensive review of all the collective changes has not be performed in order to assure the original IQ/OQ remains valid, and to assure the [computer system] does not require requalification or revalidation. [FDA 483, 2001]
- While the individual changes have been reviewed during the change control process, a comprehensive review of all the collective changes has not been performed in order to assure … the XXXX does not require requalification or revalidation. [FDA 483, 2001]
- No controls or corrective action after frequent HPLC software errors caused computer lock up. [FDA 483, 2001]
- On XXXX a laptop computer was swabbed and tested for detection of XXXX. There is no documentation of whether and when this item was decontaminated and whether and when it was used in the XXXX and subsequently in the XXXX facility. [FDA 483, 2002]
- Automated analytical equipment left in service even though system software reliability had been questioned due to frequent malfunctions that had impeded quality control procedures. [FDA 483, 2002]

REVALIDATION

Computer systems undergo change even to sustain their original design intent. Operating systems and software packages will require upgrading as vendors withdraw support for older products. New technology may prompt hardware changes to the computer system and supporting computer network infrastructure. Unless documentation is completely revised to embed changes, the document will have to be read in conjunction with change control records. As progressively more changes are made, it will become harder and harder to accurately understand current system as a whole. This will make the rigor of future change control harder because the impact of proposed

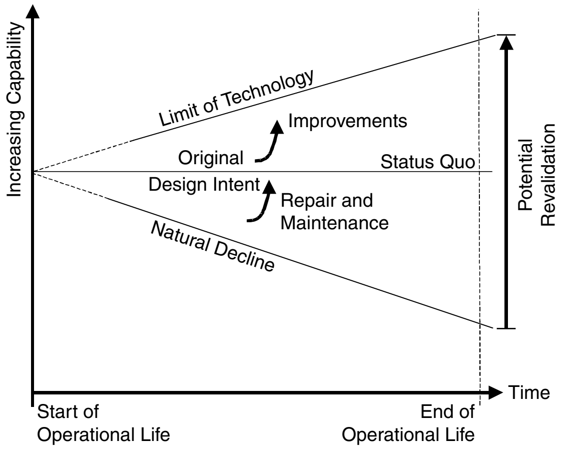


FIGURE 12.4 Degrading Validation.

changes on the existing system will be harder to evaluate. Hence the value of validation will tend to decline until the computer system validation and associated documentation is rebaselined by a revalidation exercise.

If a periodic review identifies the need to reestablish or test the confidence in the validated status, the computer system should be revalidated. Equally, if significant changes have been made or if regulatory requirements have altered, it may be deemed prudent to revalidate a computer system. In practice, the attention of operational staff to quality procedures and records often wanes unless they are carefully coached or monitored (see also Inspection Readiness in Chapter 15). As the period between successive revalidations increases, so too does the likely amount of revalidation work required (see Figure 12.4). Intervals of between 3 to 5 years between revalidations are typically appropriate.

Revalidation does not necessarily imply a full repeat of the validation life cycle; partial requalification is acceptable when justified. An analysis of changes implemented can be used to ho help determine how much revalidation is needed. Were there changes evenly spread throughout the system (sporadic) or were there focal points? Computer systems with modular architectures may allow revalidation to be segregated to particular functional elements.

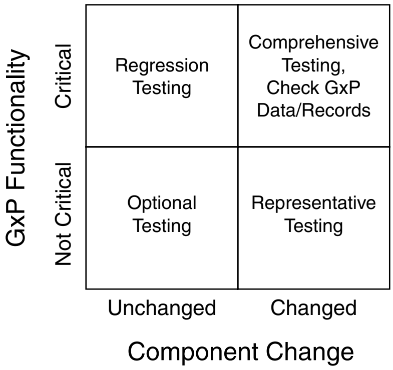
The testing strategy should ensure all critical functions are subject to comprehensive retesting regardless of whether they have changed or not (see Figure 12.5). GxP Assessments discussed in Chapter 7 can help identify what critical functionality is. Comprehensive testing should also be conducted on non-GxP-critical areas of the system functionality that have changed since original validation. All other used functionality needs only representative testing. Additional checks for GxP data over and above routine data maintenance should also be considered.

Revalidation may be synchronized to coincide with computer system upgrades in a bid to make most effective use of resources. Such strategies should be defined and approved in advance.

Revalidation can often be conducted without restricting release of the drug products whose manufacturer is supported by the computer system. Authorized Quality Assurance personnel must approve release of drug products during revalidation. In Europe this should be a Qualified Person.

Recent Inspection Findings

- There were no written Standard Operating Procedures for revalidation. [FDA, 2001]
- There was no revalidation of the XXXXXX system following revisions to the … software to demonstrate the [*function*] remains capable of the same [*operation and performance*] as demonstrated before the revision. [FDA Warning Letter, 1998]



GxP Functionality

	Unchanged	Changed
Critical	Regression Testing	Comprehensive Testing, Check GxP Data/Records
Not Critical	Optional Testing	Representative Testing

Component Change

FIGURE 12.5 Focus of Revalidation Testing.

- Changes to [*software*] processes were not always reviewed and evaluated or revalidated, where appropriate, and documented. [FDA Warning Letter, 1999]
- The software XXXX system is not periodically challenged and evaluated. [FDA 483]

REFERENCES

1. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (www.ispe.org).
2. EU Guide to Directive 91/356/EEC, *Annex 11 — Computerized Systems*, Guide to Good Manufacturing Practice for Medicinal Products.
3. ISPE (2002), “Calibration Management,” *GAMP Good Practice Guide*.
4. Pharmaceutical Inspection Co-operation Scheme (2003), *Good Practices for Computerized Systems in Regulated GxP Environments*, Pharmaceutical Inspection Convention, PI 011-1, Geneva, August.
5. OECD (1995), *GLP Consensus Document: The Application of GLP Principles to Computerized Systems*.
6. FDANews.com (2001), *Devices & Diagnostics Letter*, 28 (9), March.
7. PDA (2002), Good Practice and Compliance for Electronic Records and Signatures: Part 1 — Good Electronic Record Management (GERM), ISPE and PDA (www.pda.org).
8. ACDM/PSI (1998), *Computer Systems Validation in Clinical Research: A Practical Guide*, Version 1.1., December.
9. U.S. Code of Federal Regulations Title 21: Part 211, *Current Good Manufacturing Practice for Finished Pharmaceuticals*.
10. U.K. IQA (1994) *Pharmaceutical Supplier Code of Practice Covering the Manufacturing of Pharmaceutical Raw Material, Active Ingredients and Excipiants*, Document Reference No. P00020, Issue 2, Institute of Quality Assurance, Pharmaceutical Quality Group.
11. FDA (1995), A Memo on Current Good Manufacturing Practice Issue on Human Use Pharmaceuticals, *Human Drug Notes*, 3 (3).
12. Toigo, J.W. (2000), *Disaster Recovery Planning*, Prentice Hall, Upper Saddle River, NJ.
13. Donoghue, A. (2000*),* A Software Licensing Time Bomb That May Soon Start Ticking, *Computing*, May 4.
14. Wingate, G.A.S. (1997), *Validating Automated Manufacturing and Laboratory Applications*, Interpharm Press, Buffalo Grove, IL.
15. ICH (2000), *Good Manufacturing Practice Guide for Active Pharmaceutical Ingredients*, ICH Harmonised Tripartite Guideline, November 10.
16. U.S. Code of Federal Regulations Title 21: Part 210, *Current Good Manufacturing Practice in Manufacturing, Processing, Packaging, or Holding of Drugs*; Part 211, *Current Good Manufacturing Practice for Finished Pharmaceuticals.*

13 Phaseout and Withdrawal

CONTENTS

The end of the operational life of a computer system needs to be managed. This chapter discusses the implications of phasing out computer systems as a result of site closures, divestments, and acquisitions. Various system-management and record-management options are discussed. Key steps for all these situations include

- Retirement of the legacy system
- Archiving of electronic records and documentation
- Migration to a replacement system where appropriate
- Final decommissioning

SITE CLOSURES, DIVESTMENTS AND ACQUISITIONS

Disentangling computer systems as part of site closures, divestments, and acquisitions is becoming more complex as systems become more integrated. A decade ago systems could be switched off with little consequence. Nowadays record retention, data integrity, and security access requirements for GxP information mean the management of computer systems needs careful planning.

Site Closures

There are no additional or reduced regulatory requirements for closing sites. Computer systems should be maintained in a validated state of compliance up until the very last day of their operational life. GxP records must be archived and stored for the required retention periods. Archived records should be readily retrievable to support critical quality operations like recall, customer complaints, and batch investigation. Computer systems should then be decommissioned, as discussed later in this chapter. Some computer systems may be disassembled and sent for installation at other sites as part of a program of drug product transfers.

Site Divestments

Divested sites can typically expect a regulatory inspection within the first year after sale. Regulatory authorities will typically be interested in how operational aspects of the business were managed through the divestment process.

There are two pivotal transition dates during site divestments. First, there is the date of sale/purchase for the geographic site with computer systems *in situ* as a going concern, and second, there is the date at which the inventory of work in progress is handed over as part of the ledger of assets. Disentanglement of computer systems must take account of data responsibilities as well as operational dependencies between site systems and the other retained systems in the divesting organization.

Systems Management

Compliance issues affecting system management during divestment can be grouped under three topics:

- Validation of the computer systems
- Operation and maintenance controls
- Inspection support and dependencies

New owners of legacy computer systems are dependent on the validation conducted before they took over responsibility for the systems. Due diligence exercises are usually conducted by the new owner before taking possession, followed by a Supplier Audit on the divesting organization's support organization. Replacement systems introduced by the new owner should, of course, be validated to the new owner's standards. This will include any data migration of records from legacy systems to new systems. Table 13.1 presents various system management options.

Typically, organizations that are divesting sites will want to sever all dependencies with the divested site other than those links that may be required for an ongoing business relationship. This will reduce the regulatory dependency between the divesting organization and the new owner and the inspection vulnerability that it brings. For instance, a divested site may continue for some period to use the divesting organization's networks and MRP II system. An inspection of these computer systems at the divested site could result in regulatory corrective actions not only at the site but also across the divesting organization even though the inspection was not directly on the new owners's organization. Some divesting organizations set a threshold of 6 to 12 months support from the date for sale after which the new owner is expected to be self-sufficient. The new owner will be keen to preserve operational continuity through this period including the transition to any new system. Limited resources in the divesting organizations may mean that they cannot afford to divert operation staff to support the ongoing business of the sold site for any longer period.

TABLE 13.1
System Management Options

Option 1	Option 2	Option 3
Retain computer systems and operate applications on behalf of Divested Site	Transfer computer systems "as is" for Divested Site to operate applications	Sever computer systems and require Divested Site to migrate to new system
Advantages (largely to new owner)	**Advantages**	**Advantages (largely to former owner)**
• Continuity in business operations, no process changes • Best option in terms of lowest immediate cost	• Continuity in use of computer system • New owner empowered to make own changes • Less disruption and potential cost compared to Option 3	• Intellectual property protected • Divesting organization does not become external service provider • No inspection liability for divesting organization
Disadvantages (largely to former owner)	**Disadvantages**	**Disadvantages (largely to new owner)**
• New owner locked into divesting organization for ongoing operation and maintenance support; new owner–requested changes managed in context of divesting organization environment and priorities • Potential confidentiality issues concerning shared processes/data • Divesting organization could be included during regulatory inspection of new owner because of dependency on original integrity of production data	• New owner may still require divesting organization's network and shared servers ("open system") and hence extra controls may be required • Divesting organization could still be inspected as a result of new owner regulatory inspection if computer systems have cross-reference dependencies on divesting organization documentation; significantly less risk of inspection than Option 1	• Discontinuity in use of legacy systems (may also be advantage to new owner) • Divestment of site may be delayed in order to bring new system into operation (may also be disadvantage to divesting organization) • Probably most disruptive and expensive option
Implementation Activities	**Implementation Activities**	**Implementation Activities**
• New owner conduct Supplier Audit on divesting organization as external service provider • Formal contract of supply required • Service Level Agreement established for maintenance and inspection support	• New owner conducts due diligence on divesting organization's validation; controlled copy of all relevant documentation made available to site, marked "copy of original" • Local procedures should be made autonomous by new owner	• Agree on replacement system • Conduct data migration from legacy computer systems • New owner validates new systems in accordance with new owner standards

The operation and maintenance of regulated computer systems has already been discussed in Chapter 12. The new owner should ensure that whoever is supporting their computer systems (divesting organization, third party, or internal support group) is effectively managing these requirements:

- Performance monitoring
- Repair and preventative maintenance
- Upgrades, bug fixes, and patches
- Data maintenance
- Backup and restoration
- Archive and retrieval
- Business continuity planning

- Security
- Contracts and Service Level Agreements (SLAs)
- User procedures
- Periodic review and revalidation

The new owner should ensure operation and maintenance procedures are clearly marked as approved and operated by them when they take over responsibility for supporting the legacy systems.

Both the new owner and divesting organization may have particular sensitivities around inspection readiness. Regulatory observations on the new owner could imply corrective action for the divesting organization. Equally, the new owner will, at least for a period, be dependent on inspection support from the divesting organization for existing systems until he or she becomes sufficiently familiar with them. A transitional support agreement is typically built into the sale/purchase contract possibly as a Service Level Agreement (SLA). Both the divesting organization and new owner are usually keen for the new owner to become independent of the divesting company as soon as reasonably possible. Transitional arrangements — both technical and inspection support — typically last for less than a year.

Records Management

Compliance issues affecting records management during divestment can be grouped under four topics:

- Records retention
- Records retrieval
- Access controls
- Data integrity

Records retention affects both the divesting organization and the new owner. Figure 13.1 illustrates the various record management scenarios that might exist. The divesting organization is accountable for historical product data within required retention periods. Examples of GxP records include batch records and supporting information such as analytical results. Contracts should specify any transition period after the date of the site sale during which work in progress is completed and subsequently owned by the divesting organization. A complete copy of relevant product inventory information should be taken by the divesting organization. Operational data meanwhile typically becomes the responsibility of the new owner from the date of sale/purchase. Examples of GxP records would include calibration records, change control records, etc. Contracts should specify that the new owner will maintain historical records for a defined period and, where necessary provide copies in support of a batch investigation by the divesting organization.

Just as for closing sites GxP records should be maintained on systems that facilitate timely access in support of critical quality operations like recall, customer complaints, and batch investigation. The divesting organization will need to establish suitable record retention and retrieval systems. Alternatively the divesting organization could ask the new owner to retain GxP record and provide a retrieval service where it has been agreed the new owner will maintain legacy data. In this scenario the new owner becomes a service provider and formal contracts with Service Level Agreements should be agreed and audited by the divesting organization. The regulations require ready access to records and documentation; there are no requirements prohibiting this being the new owner on behalf of the divesting organization.

Access controls are needed to restrict change to authorized users and protect information from authorized modification (inadvertent or malicious). Computer applications managing records may be under the control of the divesting organization ("open systems") or the new owner ("closed systems"). Open systems require additional controls, as discussed in Chapter 15. Security in general is discussed in Chapter 12.

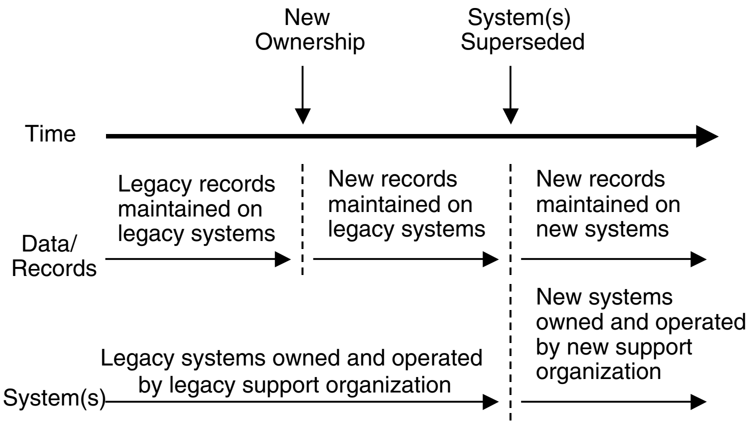

New Ownership
System(s) Superseded
Time
Data/ Records
Legacy records maintained on legacy systems
New records maintained on legacy systems
New records maintained on new systems
System(s)
Legacy systems owned and operated by legacy support organization
New systems owned and operated by new support organization

FIGURE 13.1 Transition Timeline.

Change control and audit trails are key aspects requiring management to assure data integrity and detect corruption (see Chapter 12 for a discussion of data maintenance). In addition there may be electronic record management requirements. More information regarding regulatory expectations in this regard can be found in Chapter 15.

Site Acquisitions

It should be recognized that what is a phaseout for one organization may be a phase-in for another organization. Divestments are also acquisitions depending on which side of the fence an organization is on. The new owner needs to consider both system-management and records-management requirements as already indicated. Once new owner computer systems are installed, data migration will be needed from the legacy systems that can then be decommissioned. Migration and decommissioning are discussed further later in this chapter.

RETIREMENT

Computer system retirement describes the process of taking a system out of operation. That is, the system is not in active or routine use. This decision may be taken for a number of reasons including that the application is redundant, the computer technology is obsolete, it cannot comply with new regulatory requirements, or perhaps a replacement system with added functionality is planned. It is important to understand that retirement does not necessarily indicate the end of a computer system's life. A computer system may be brought out of retirement if required, unless it has been fully decommissioned (scrapped).

When a GxP computer system is retired, the request is often made and implemented through a Change Control process. A Retirement Plan should be formulated to address the steps needed to retire the system, identify what (if any) new system will replace the current system, timelines for the retirement process, and the individuals responsible for the retirement process. The rationale for retiring the system must be documented.

The process of transferring records from current to the new system should be an element of the project plan and must be qualified. Measures should also be in place to ensure that archived records on retired system can still be accessed and read.

Once retirement is complete a Retirement Report should be prepared in response to the Retirement Plan. After this is done, a decision can be made whether or not to switch off and decommission the system.

Electronic Records Management

An electronic records management framework should be formulated and deployed. Steps within the framework might include:[1]

- Determine and document which records need to be retained.
- Maintain a system for tracking the locations where electronic records are stored (hard drives on mainframes and personal computers, magnetic tapes, disks, CDs and other media). This system is required to enable timely retrieval of electronic data.
- Ensure the storage media can be read, maintaining mechanical tools such as microfiche readers and logical tools such as record indexes as required.
- Provide for off-site storage of the records needed for disaster recovery.
- Ensure that contracts with consultants, services providers, and other third parties require compliance with the company's record policies and permit periodic audits.
- Document policies and procedures for creating, storing, destroying, and indexing different types of information. Disposition should cover evidence that a record was destroyed, when it was destroyed, who destroyed it, and how it was destroyed.
- Ensure that similar records are treated similarly, whether paper or electronic.
- Require authorized procedures to be followed in purging electronic records.
- Develop a procedure to suspend the disposition of records if a lawsuit is filed or is imminent.
- Document that policies and procedures have been followed in retaining and disposing of electronic records.
- Educate employees and other personnel authorized to use the company's advanced technologies about the company's records retention policy.
- Conduct periodic audits to ensure compliance with the company's records retention policy.
- Identify persons responsible for compliance with records programs.
- Provide review of the framework to adapt to changing technology, evolving company directions, and emerging judicial and regulatory trends.

A regular review of data stored in the archive is essential not only as indicated earlier to detect any degradation of the storage media, but also to determine if the archive technology or record is becoming redundant. Periodic assessments will be needed to decide whether or not to maintain the archive electronic records. It may be decided only to maintain critical records such as those involved with batch records, batch sentencing and recall, over longer periods of time. Once the retention period is over, a follow-on decision will need to be taken as to whether to retain the electronic records for a further period or to destroy them. The minimum retention times for some example electronic records is indicated in Chapter 12.

No electronically stored data should be destroyed without management authorization and relevant documentation. Other data held in support of computerized systems, such as source code and development, validation, operation, maintenance, and monitoring records, should be held for at least as long as the records associated with these systems (e.g., Section 9 of the *GLP Consensus Document*[2]).

Long-Term Preservation of Archive Records

The FDA has clearly stated in an industry guide and conferences that 21 CFR Part 11 compliance extends beyond the retirement of a computer system. For example:[3]

> *Recognizing that computer products may be discontinued or supplanted by newer (possibly incompatible) systems, it is nonetheless vital that sponsors retain the ability to retrieve and review the data*

> *recorded by the older systems. This may be achieved by maintaining support for the older systems or transcribing data to the newer systems.*

Long-term storage presents its own special challenges. The FDA expectations are summarized below:[3]

- All versions of application software and software development tools involved in processing of data or records should be available as long as data or records associated with these versions are required to be retained.
- Any data retrieval software, script, or query logic used for the purpose of manipulating, querying or extracting data for report generating purposes should be documented and maintained for the life of the report.
- [Pharmaceutical and healthcare companies] may retain these themselves or may contract vendors to retain the ability to run (but not necessarily support) the software.
- Although the FDA expects [pharmaceutical and healthcare companies] or vendors to retain the ability to run older versions of software, the agency acknowledges that, in some cases, it will be difficult for [pharmaceutical and healthcare companies] and vendors to run older computerized systems.

The content of an electronic record must therefore be maintained in a form that is readable after the system used to create it is obsolete. For instance, a document originally stored today in Microsoft Word 7 format might need to be retained for regulatory reasons for 30 years when Microsoft Word 7 is no longer available. This issue is compounded as Microsoft Word, for instance, has links to other applications that may be used to generate and maintain inserted content (e.g., PowerPoint diagrams) in the electronic record. It is insufficient just to store the text, as the record should appear to retrievers in its original format. Furthermore, the file formats may be dependent on systems software (operating systems, databases, compilers, etc.) and hardware. Potentially software and hardware will need to be archived, but the practicality of this must be questioned.

A strategy must be put in place to migrate electronic records to new types of media as and when they are introduced. Media reliability is a potential problem, but is fairly well understood. For instance, DAT and CD-ROMs have a notional operational life of 5 and 10 years, respectively, if they are not copied and kept in good storage conditions. It is more likely that the media technology will become obsolete within the electronic record's operational lifetime. Media technology is currently being superseded every 5 years. The content of old media archive will need to be copied to new media archive to prevent any loss. It is wise not to rely on a single archive copy just in case the operational life of an archive copy degrades earlier than expected.

Whereever possible employ standard data formats for archive copies to assist in any recovery process when original equipment to read specialist data formats may not be available. Industry standards are not widely used at present, with products often specifically implementing new functions and standards as a means of retaining existing customers and attracting new ones. Portability seems a long way off.

Pharmaceutical and healthcare companies need to keep appropriate computer systems that are capable of reading electronic records for as long as those records must be retained. Maintaining a legacy computer system just to read old records can be expensive especially when this strategy might still require transfer to a new system or format at a later date when maintenance becomes impractical.

Where system obsolescence forces a need to transfer electronic documentation from one system format to another, the process must be recorded step by step and its integrity verified.[1] An exact copy must be verified prior to any destruction of the original media. The obsolete system could alternatively be maintained as a legacy system, an approach that can be expensive and one that might still require transfer to a new system or format at a later date when maintenance becomes impractical.

If the existing system is not validated, the integrity of the data within the system cannot be relied upon. Data cannot simply be transferred to a new electronic repository without data verification.

Retrieval Considerations

Archive records need to be accessible for a number of years, perhaps to people who were not involved in any way with their generation. For this reason, other related information needs to be stored alongside the original information, and this is usually referred to as metadata, to provide a context that makes the information easier to retrieve.

Preservation Considerations

Retention requirements for electronic records are discussed in Chapter 12. It is important to remember that electronic data capture can undermine data integrity. Image capture techniques may reproduce an original record very accurately, but if the original has insufficient dots per inch for clear reading, then the reproduction may not be usable. Electronic records are often not nearly as rugged and durable as their paper counterparts. The following factors may affect their life expectancy:[1]

- Quality of storage medium
- The number of times the medium is viewed
- Care in handling
- Storage temperature and humidity level
- Cleanliness of storage environment
- Quality of the recorder used to write to the media

Business Continuity Plans should prompt the development of a media storage strategy for critical records (e.g., paper or fiche) to enable the retention of access to these records in the event of a system failure or access to critical records once the system has been switched off.

Archiving Options

The long-term archive of electronic records would seem to be fraught with difficulties. Options for a way forward that would allow the original system and software to be decommissioned include the following:

- Maintain records on legacy system (time capsule).
- Emulate old software on new hardware/software.
- Migrate electronic records to new system.
- Store data in an industry standard format.
- Take a printed paper copy, microfilm, or microfiche.

An assessment must be performed and documented to determine the most appropriate method for preserving archives. Selection of the appropriate method must be considered within the context of the size, complexity, scope, and business impact of the system to be decommissioned. The method chosen must be documented using the appropriate Change Control form.

Maintain Legacy Computerized System

Retaining the legacy computer system as a "time capsule" is one method of maintaining original software and configuration functionality.[4] However, it is unlikely that the hardware and software will be supported by the supplier for the extended period that some record retention periods require.

Any inability to maintain legacy systems will increase the likelihood that the retrieval may be unsuccessful. Therefore it is recommended that this method is not to be relied upon for periods of a few years beyond the supplier support for that system.

Key steps:

- Back up the entire system for contingency protection in case of failure.
- Reduce user access to "read only" operation in relation to required electronic record, amend SOPs accordingly, and validate.
- Maintain the ability to restore the application, data, and operating environment on a vendor-supported hardware environment.
- Operate system only when needed.
- Ensure integrity of electronically signed records is demonstrable.
- Validate record retrieval relevant to GxP processes.

Emulation of Old Software on New Hardware/Software

Suppliers sometimes provide this facility as part of an upgrade or replacement product. This option, if available, is a useful alternative to migrating records to entirely new computerized archive system. The integrity of the emulation facility must be verified. Hopefully the emulator can be considered as standard software; otherwise the software will have to be treated as bespoke code and validated as such.

Key steps:

- Back up the entire system for contingency protection in case of failure.
- Ensure search and sort query reporting facilities are available or developed.
- Ensure integrity of electronically signed records is demonstrable.
- Validate emulation created including record retrieval relevant to GxP processes.

Migrate Electronic Records to a New System

Electronic records are copied, possibly reprocessed, to make them accessible by a new computerized archive system. This can be a large and complex task but has the advantage that the new system is specifically designed for the purpose. This method, however, should not be used where the integrity of the original records being migrated can be disputed, unless data accuracy checks are implemented. Data load requirements are discussed in Chapter 11.

Key steps:

- Back up the entire system for contingency protection in case of failure.
- "Mirror" legacy data architectures within new system/database(s).
- Validate data migration including any support programs used.
- Ensure search and sort query reporting facilities are available or developed.
- Ensure integrity of electronically signed records is demonstrable.
- Validate new system created, including record retrieval relevant to GxP processes.

Store Data in an Industry Standard Format

This approach works well with simple data configurations (e.g., small self-contained data tables). Because industry standard formats are used, the risk of technical obsolescence is reduced and consequently the likelihood of archive migration minimized. Examples might include RTF rather than Microsoft Word 7 formats. Electronic records can also be stored as images (e.g., PDF format) although this increases storage volume requirements significantly. This method should not be used where there is a loss of data processing capability (e.g., search and sort cannot be run, and spreadsheet formulas are lost when the records are converted).

Key steps:

- Capture any necessary metadata in converted electronic records
- Validate data migration including any programs used to generate output to archive media
- Ensure search and sort query reporting facilities are available or developed
- Ensure integrity of electronically signed records is demonstrable
- Validate record retrieval relevant to GxP processes

Take a Printed Paper Copy, Microfilm, or Microfiche

This sounds simple but may not be practical because the volume printing can be enormous. Printing may also be complicated where electronic records are made up of distributed data that requires electronic queries to retrieve it. These data structures are usually by far the most efficient storage mechanism for the electronic records. Printing can multiply the scale of archive task by a factor of 100. When large volumes of information are archived in this way, it is often pertinent to build a companion index to aid search and retrieval. A simple computer system can typically be developed to do this. Any programs or tools used to generate records suitable for archiving on paper, microfilm, or microfiche must be validated.

Key steps:

- Capture any necessary metadata in converted electronic records
- Validate data migration including any programs used to generate output to archive media
- Ensure search and sort query reporting facilities are available or developed
- Ensure integrity of electronically signed records is demonstrable
- Validate record retrieval relevant to GxP processes

Regulatory authorities do accept printed copies of original electronic records provided prints are exact copies of original records. For instance, GMP and GLP predicate rules that it relies on to identify affected records also state (Clause 180(d) of U.S. Code of Federal Regulations[5,6] and Clause 195(g) of the Code[7]):

> *Records required by this part may be retained either as original records or as true copies such as photocopies, microfilm, microfiche, or other accurate reproductions of the original records.*

It is not necessary to reprocess archived information to prove the integrity of historical records; rather, it is expected that archived information can be used as constructive evidence to support the accuracy of historical records.

REPLACEMENT SYSTEMS

Companies should set and review a migration strategy that addresses both near-term and long-term corporate needs for individual computer systems. When migrating from manual to computerized systems or upgrading computer technology, the following implications should be considered:

- Configuration flexibility and capacity for expansion
- Financial cost
- Installation impact on operations
- Integration capability
- Performance improvement

- Personnel requirements
- Technology risk
- Validation requirements
- Supplier capability

Computer systems employed should meet or exceed the validation requirements for the manual functions they replace.[7] The new computer system must be at least as reliable as the computer system it replaces. Pharmaceutical and healthcare regulations do not mandate parallel operation of manual systems being replaced by computerized systems. If a period of parallel operation has been decided upon, it should be run with the purpose of demonstrating that the computerized system is better than the old manual system, and the manual system can be decommissioned. It is unacceptable, however, to rely on parallel operating as the sole basis of validation.[8] The replacement system must be validated in its own right.[9,10]

Practitioners should not necessarily run the system in parallel until there are no "bugs"; the real question is whether the bugs can be managed. Parallel operation, of course, may not always be possible or desired. The personnel requirements to run two systems together may be considered too high or perhaps would require two production facilities.

Migration Strategy

Once a computer system has been implemented, the pharmaceutical and healthcare company must appreciate that computer technology is continually advancing. The next generation of microprocessor technology and software (half the price or double the functionality) has been arriving on average every 2 or 3 years, and there seems to be no reason to suspect that this trend will not continue. The next generation may consist of an upgrade to the computer system or its replacement. The various migration options are shown in Figure 13.2.[11] Not every option to upgrade may be accepted, but care must be taken not to slip unknowingly into obsolescence when older versions are no longer supported by their suppliers. Alternatively, there may be reasons for ceasing all updates and establishing a legacy system.

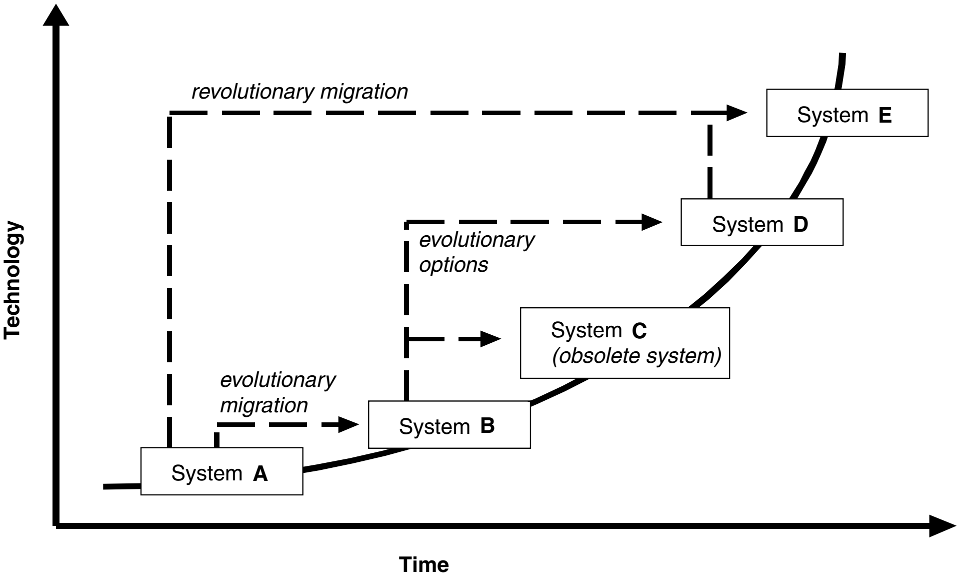


FIGURE 13.2 Migration Routes.

Regular upgrades following an evolutionary migration are associated with low technology risks, but the combined validation effort for every upgrade can be considerable. One aspect of computer systems that can be overlooked is the upgrading of hardware components (such as printers, monitors, instruments) and system software (such as operating systems and standard packages). In particular, system software is continually being upgraded, and while upward compatibility may be claimed on the initial release of an upgrade, confidence without supporting evidence should be limited. In this situation, it is recommended that installation of the upgrade be delayed until the new release is market tested.

Major step changes in technology across several generation upgrades (a revolutionary approach) will reduce the overall validation effort, but the technology risk can be high. Examples of step changes include the cut-over of large systems, such as MRP IIs, where parallel operation may not be practical because of the large volume of data and user interaction. In order to reduce the risk, larger systems are usually implemented in stages with phased cut-overs for main functional elements. Within the MRP II system, cut-overs might include financials, customer services, and manufacturing.

Legacy Systems

Computer systems that do not implement software and hardware upgrades will become obsolete. Updated software and hardware are usually installed only if they include bug fixes or if support for the old version is being removed. New versions of products, however, do not always bring operational benefits. Early adopters may find bugs yet undiscovered by the supplier (e.g., Pentium® processor). Equally, new product versions may actually degrade overall system performance (e.g., the original system memory is insufficient for the new data processing requirements of Windows 95®). In these circumstances, it is advisable to retain and operate the original system, wait a period (perhaps 6 months) for a favorable track record to be established by other industry practitioners with the updated products, and only then install the revisions.

Other obsolete systems exist because suppliers are no longer supporting their software or hardware products. A decline in the number of users of a product may lead a supplier to question the financial viability of their continued support of the product. Pharmaceutical and healthcare companies must discuss this topic with their suppliers so that a suitable validation strategy can be planned.

Legacy systems are quite acceptable, provided the original system has been validated to current GMP requirements and its validation status is being maintained. Validation activities will include the following:

- Establishing version and change control
- Collating documentary evidence that the software and hardware provided by a supplier have been developed and maintained under a quality assurance regime supporting validation
- Reviewing documentation and preparing any supplementary information required to make the documentation complete
- Investigating the supply chain of any second-hand software and hardware used to maintain the system to establish whether it came from the original supplier and has not suffered any damage
- Testing critical features with additional tests to supplement, where necessary, supplier testing

If validation is not practical, pharmaceutical and healthcare companies should consider selecting and replacing legacy software and hardware with equivalent products or replacing the entire computer system. This may involve using alternative suppliers. New software or hardware in a legacy system will require validation to confirm that its functionality operates as required and that it does not affect what remains of the original system.

DECOMISSIONING

Computerized systems are generally decommissioned either when they have become technologically obsolete, they have become too unreliable, or the process they are controlling has become obsolete. Decommissioning may also take place after an adverse regulatory inspection demands their replacement. The computerized system may, nevertheless, still be needed at a later date to support a new or rejuvenated process. The validation requirements of decommissioning must be carefully considered. There are validation issues if documentation is needed in relation to a future recall of a drug product or if the system is used again in the future. Documentation may also be required if for any reason there is a regulatory investigation affecting the system.

Decommissioning will normally be based on an established shutdown procedure. There may, however, be special decommissioning operations that have not been used before on the live system. Operations management must ensure that decommissioning hazards are identified and that procedures are defined to avoid any accidents. Critical instrumentation should be checked to verify that it is still operating within calibrated ranges.

When decommissioning is complete, a short report on the validation of the computerized system should be composed to pass on any learning points. Only when this report has been issued and any archiving complete can operations managers relinquish their responsibility for the system.

Validation cost for the new system could be halved if it is similar to the original application. If there is any possibility of the system being used again, it should be dismantled and tagged, carefully packaged and labeled, and stored in a secure location. Documentary evidence supporting its validation must be archived and retained. System specifications, Development Testing, IQ, OQ, PQ, user manuals, and maintenance procedures could prove very useful if the system is reused.

REFERENCES

1. Kahn, R.A. and Vaiden, K.L. (1999), If the Slate is Wiped Clean — Spoliation: What It Can Mean for Your Case, *Business Law Today*, American Bar Association Publication, May/June.
2. OECD (1995), *GLP Consensus Document: The Application of the Principles of GLP to Computerised Systems*, Environment Monograph No. 116, Environment Directorate, Paris, 1995.
3. FDA (1999), *Computerized Systems Used in Clinical Trials*, Guidance to Industry, April.
4. PDA (2002), *Good Practice and Compliance for Electronic Records and Signatures: Part 1 — Good Electronic Record Management (GERM)*, published by ISPE and PDA (www.ispe.org).
5. U.S. Code of Federal Regulations Title 21: Part 211, *Current Good Manufacturing Practice for Finished Pharmaceuticals*.
6. U.S. Code of Federal Regulations Title 21: Part 58, *Good Laboratory Practice for Non-Clinical Studies*.
7. U.K. Department of Health (1989*)*, *Good Laboratory Practice: The Application of GLP to Computer Systems*, United Kingdom Compliance Programme, Department of Health, London.
8. Tetzlaff, R.F. (1992), GMP Documentation Requirements for Automated Systems: Parts 1, 2 and 3, *Pharmaceutical Technology*, 16 (3): 112–124, 16 (4): 60–72, 16 (5): 70–82.
9. *Australian Code of Good Manufacturing for Therapeutical Goods* (1990), Medicinal Products — Part 1, Section 9, Therapeutic Goods Administration, Woden, Australia.
10. European Union Guide to Directive 91/356/EEC (1993), Computerised Systems, Annex 11 of *European Commission Directive Laying Down the Principles of Good Manufacturing Practice for Medicinal Products for Human Use*.
11. Salazar, J.M., Gopal, C., and Mlodozeniec, A. (1991), Computer Migration and Validation: A Vendor's Perspective, *Pharmaceutical Technology*, June.

APPENDIX 13A
EXAMPLE RETIREMENT CHECKLIST

This checklist provides the activities, concerns, and issues that may need to be addressed when a system is retired.

- Determine and document rationale for retiring the system.
- Determine the impact of system retirement on other systems or users.
- The records retention requirements for the specified records will determine whether or not the records must be archived in a format that will allow for subsequent inspection of the records.
- If the system is being replaced by another system, retrieve archived records for loading into the replacement system. Verification of the successful migration of the records will be demonstrated as part of the validation process of the new system.
- Develop the retirement schedule for the system.
- Communicate the retirement schedule to the client community.
- Document client community approval for retirement.
- Determine what system-related documentation should be archived (e.g., source code, life-cycle documentation, user and technical manuals, security, system change control logs, etc.).
- Document final disposition of system hardware and software.
- Retire any system-specific SOPS.
- Determine appropriate storage medium for archived materials (e.g., ASCII format files, printed records stored to microfiche, etc.).
- Remove access to the system.
- Clean up any system logical/symbols/menu references.
- Delete the software and associated files from the system.
- Notify all affected personnel to discontinue regular system support activities (such as regular backups, preventive maintenance, etc.).

14 Validation Strategies

CONTENTS

This chapter examines various validation strategies that can be adopted around organizational roles, outsourcing, standardizing computer applications and software reuse, segregating GxP aspects of integrated systems, retrospective validation of legacy systems, and use of statistical techniques to support validation.

ORGANIZATIONAL STRUCTURES

Questions often arise regarding the relationship of internal vs. external suppliers, especially within large pharmaceutical and healthcare organizations, and the corresponding role of Quality and Compliance. Expectations for these organizational structures are discussed below.

Quality and Compliance Roles

Regulatory authorities require pharmaceutical and healthcare companies to have a Quality organization (sometimes referred to as a Quality Unit). The role of the Quality organization covers both Operational Quality (individual project/system support) and Compliance Oversight (corporate governance of management practices). Table 14.1 compares R&D (GCP/GLP), manufacturing and distribution (GDP/GMP), and medical device regulatory clauses relating to quality organization responsibilities for computer compliance.

TABLE 14.1
Quality and Compliance Organizational Roles

		Organizational Roles	
		Operational Quality	**Compliance Oversight**
Regulatory Responsibilities	**GCP/GLP** [FDA refer to Quality Assurance Unit]	• Ensure all data are reliable and processed correctly[6]	• Set policy • Responsible for procedures applicable to the QA Unit[1,6,8] for in-house and purchased systems[7] • Compliance auditing[2,6] • Compliance monitoring (review and inspection)[1,6–8]
	GDP/GMP [FDA refer to Quality Control Unit]	• Ensure validations are carried out[2] • Oversee whole qualification and validation process[3] • Review and approve validation protocols[4,5] and validation reports[4] • Review changes[5] that potentially affect product quality[4] • Determine if and when revalidation is warranted[5]	• Set policy • Oversight of validation procedures[5] • Compliance auditing[2,5] • Make sure internal audits (self-inspections) are conducted[4] • Review effectiveness of QA systems[2] • Conduct GDP/GMP training[2]
	Medical Devices	• Establish quality plans[9]	• Set policy[9] • Establish Quality System procedures[9] • Conduct quality audits[9] • Review performance of Quality System[9]

Operational Quality and Compliance Oversight groups can exist as separate groups or as a single group, depending upon the size and structure of the organization. Some controls on who does what do need to be specifically managed. For instance, a quality professional providing direct project/system support on one system should not be allowed to audit that same system because this would compromise the auditor's independence. One pharmaceutical company describes this way of working as "QC at home, QA away."

GDP/GMP Quality Unit

The Quality Unit must be independent of those parts of the organization responsible for testing[1] and production[2] and has a critical role in overseeing the whole qualification and validation process.[3] It is expected to:

- Be involved in all quality matters[4]
- Review and approve all appropriate quality-related records and documentation[4]
- Ensure timely notification of compliance issues to management[1,4]

The main responsibilities of the Quality Unit should not be delegated.[4] The FDA believes such accountability will result in more consistent and reliable compliance.[5]

GCP/GLP Quality Unit

The British Association for Research Quality Assurance (BARQA) has interpreted international GCP/GLP regulations and expects the GCP/GLP Quality Unit to:[10]

- Conduct GCP/GLP awareness training, validation training, and change control training
- Review and approve validation and change control procedures
- Review quality plans and key validation documents (i.e., Validation Plan, Requirements, Test Plan, Test Results, Acceptance, Record Retention (Archiving and Change Control)
- Advise projects on software development
- Review changes (individually or as part of periodic review process)
- Conduct system audits (including system development, software, operation, and use)

Concept of Internal Supplier

IT organizations within pharmaceutical and healthcare companies sometimes refer to themselves as internal suppliers. Often inherent in the use of this description is the belief that they can abdicate responsibility for validation — validation becomes entirely the responsibility of the end user. This is a serious misjudgement. End-user validation is typically highly dependent on compliant work by the central IT organization. Regulatory authorities are likely to inspect central IT organizations when they realize this dependency. It is important to recognize that regulatory expectations and validation standards are the same for internal suppliers as for end-user developments. The basic role of the Quality Unit remains unchanged and indeed is likely to have line management outside the IT organization to demonstrate its independence.

Any change in the so-called internal supplier organization or associated ways of working must be carefully managed. Care must be taken not to inadvertently create a discontinuity in support or system documentation. For example, tracing the validation documentation between two different Quality Management Systems years later can be quite difficult to explain in a credible manner. Transitions between organizational structures and Quality Management Systems are fertile ground for noncompliance.

Central Development and Support Groups

In an effort to exploit standardization many organizations have established central groups to develop and support common systems. The objective is to establish consistent, effective, and efficient business processes and to minimize development, support, and validation costs. As such, site adaptation of applications is strongly discouraged if not forbidden. Examples of situations where central development and support groups make sense include:

- Multiple locations served by a single shared implementation of an application (e.g., MRP II)
- Multiple locations sharing the design for their own implementation of a common application (e.g., LIMS, Distribution Systems, and common DCS)

Central development organization for a particular system may be separate or combined with its central support organization. However, if a central development organization exists without a reciprocal central support organization or an acting custodian (e.g., lead site), common systems tend to diverge and overall management control is lost. Both central development and central support groups should have Quality Unit support.

Pharmaceutical and healthcare companies tend to have an ebb and flow in regard to centralized and decentralized organizations. This is often reflected in the harmonization or disparity in validation practices adopted between sites or geographic regions of a company. The cyclic nature of the organizational changes must be managed to minimize the impact on consistent validation standards and practices. Centralized Validation Departments must not lose touch with the hands-on experience of the operating site. Decentralized Validation Departments must ensure that a suitable support network is established with focal points for maintaining a common vision and approach. A hybrid of centralized and decentralized organizational structures is recommended to release the best of both worlds and avoid the pitfalls of relying solely on either.

External Supplier Responsibilities

Goods and services, including software-based systems,[11] must correspond to their description and be of a merchantable quality (fit for purpose).[12–14] Both GxP regulatory requirements and commercial contract law share the objective of computer systems being “fit for purpose,” and this should be achieved through good professional practice.

Although GxP requirements hold the pharmaceutical and healthcare companies directly accountable for all aspects of computer validation, in contract law if the supplier knows the customer’s application intent (regardless of the product’s common usage), the goods or services must be fit for that intended purpose. This does not mean pharmaceutical or healthcare companies can defer regulatory observations of noncompliance and the liability for corrective actions directly on to their suppliers. Rather it opens up the possibility for pharmaceutical and healthcare companies, after receiving a noncompliance observation from a regulatory authority, to take the supplier separately to court if under commercial contract law it is felt the supplier’s actions were responsible for the regulatory deficiencies.

Duty and Standard of Care

Duty of care is based on avoiding reasonably foreseeable adverse consequences. The failure of “duty of care” implies negligence. It has been successfully applied to deficiencies in:

- Design
- Construction

- Inspection
- User instructions
- Data security

and hence covers some basic attributes of GxP. In addition, there is the general expectation of safe operation.[15–17] Data security, which includes access security, is mandated in many counties by laws protecting individuals and organizations from the misuse of information.[18]

Within the U.K., a "standard of care" is imposed on the equipment producer who is liable to compensate the pharmaceutical or healthcare company for personal loss but not for corporate damage.[19] The concept of "standard of care" is very similar to that of "duty of care." Prosecution for negligence of care must usually be brought within some limited period from the date of supply. In the U.K., this period is 3 years from loss or awareness of loss and cannot be brought after 10 years from the date of original supply. Other legislation may strengthen the regulation affecting some aspects of supply, such as supply chains.[20]

Breach of Contract

A successful suit for damages (breach of contract) must satisfy a "reasonableness" test demonstrating negligence. Exclusion clauses are usually implemented as a defence against damages. However, within the U.K., their use is limited, and negligence can never be the subject of such a clause.[21] Indemnities can be used to pass damage responsibility to a third party who supplied the source of system noncompliance. There may, of course be a joint responsibility between the primary subject of the breach of contract and the third party, in which case responsibility may be shared. In this way (depending on supply roles), a combination of system integrator, equipment hardware supplier, and software supplier may be held accountable for breach of contract between the end customer and the primary supplier. Examples of accountability include software installed on an inappropriate hardware platform, system inappropriately implemented as customer solution, or operational instructions not followed during maintenance servicing. Indemnity is strictly controlled through common law; there must be no doubt of accountability. Secondary contracts limiting the liability of the initial contract are legally permissible but are unlikely to pass the "reasonableness" test. The EU Directive on Unfair Terms in Consumer Contracts (93113/EEC) interprets all ambiguous contract clauses in favor of the end customer, which in the case of validation is likely to be the pharmaceutical or healthcare company.[22]

Associated with the breach of contract is "misrepresentation." This is a misleading understanding given outside the contract but that is integral to the contract agreement. Such an understanding might involve the qualification and experience of personnel implementing the contract. Pharmaceutical and healthcare companies or their suppliers may be allowed to rescind the contract but only to recover costs where misrepresentation is fraudulent or negligent.[23]

Legal Defensive Positions

The overwhelming majority of contracts are brought to a successful close. If fulfillment of the contract is disputed, the prosecuted party has four basic defenses:

1. Presentation of an ISO 9000–accredited quality system, adherence to established company and industry practices, and use of competent personnel.
2. Demonstrating the likelihood that the adverse consequence was introduced by the user subsequent to delivery by the supplier. Evidence of predelivery inspection and testing is required.
3. Presentation of a "development risk" whereby the bespoke nature of an application is presented as a source of acceptable risk. This argument, however, is usually self-defeating

because in such circumstances it is "reasonable" to apply more rigorous development practices.

4. The goods or services supplied conform to the customer's formal requirements and it was these requirements that were deficient. In practice, user requirements are rarely precise enough to begin debating this defense.

These defenses highlight the importance of mutual respect and partnership within a working supplier–customer relationship. It is in both parties' interests to ensure that contracts are fair and rigorous.

Liability of Personnel

Employees can, in theory, be sued for breach of contract if they are shown not to have taken reasonable care in their duties. In practice, this rarely happens due to the limited recoverable resources from the individual. Instead, the employee is subject to disciplinary action and the possibility of dismissal.

Negligent work by an employee under employer management or established employer practice is the responsibility of the employer. It is the employers' responsibility to demonstrate that their management and practices were not negligent to defend against this position. Company directors representing the employer may be accountable for the employee's negligence if they have a duty covering the negligence, and there is "gross negligence." However, proving gross negligence in the absence of unambiguous evidence is extremely difficult.

Contractors under a "contract for services," like employees, can be sued for breach of contract where they are shown not to have taken reasonable care in their duties. In practice, however, because of their limited recoverable resources, it is far more likely that they will be dismissed. The position of contractors as "independent" for the purpose of prosecution for negligence is complex. Independence implies that the contractor worked outside employer management and employer practice. This is rarely the case and contractors are treated by the law as employees.

Regulatory Authority Responsibilities

GxP regulatory authorities also have a "duty of care" to the pharmaceutical and healthcare companies inspected, but what constitutes their duties is not precisely defined. Few cases have been successfully brought against GxP regulators.

OUTSOURCING

Outsourcing can be a very attractive means to reduce the cost of ownership associated with computer systems. With added pressures on pharmaceutical and healthcare companies to reduce headcount, the transfer of personnel to the outsourcing company as a part of the "deal" can be an added benefit. Outsourcing, however, should not be gone into lightly. The pharmaceutical or healthcare company will become entirely dependent on the outsourcing company for the computer systems included. Poor levels of service often have a direct impact on the operation of the pharmaceutical or healthcare company. Breaking away from one outsourcing company back to the pharmaceutical or healthcare company or to another outsourcing company can be a very painful experience.

Regulatory Requirements

Pharmaceutical and healthcare companies are accountable to the GxP regulatory authorities for the actions undertaken by the outsourcing company. FDA regulations, for example, simply require that personnel have the appropriate combination of education, training, and experience to perform their assigned tasks.[24] It is further expected that training in current good manufacturing practice shall be conducted by qualified individuals on a continuing basis and with sufficient frequency to assure that employees remain familiar with the GxP requirements applicable to them. European Union regulations meanwhile discuss extensively the roles of the contract giver and contract acceptor. Due diligence on behalf of the pharmaceutical or healthcare company is expected not only on the technical ability of the outsourcing company to perform the desired job but also that any outsourcing company meets the regulatory compliance requirements.[25] A Supplier Audit as presented in Chapter 7 should therefore be conducted. This principle is consistent with the expectations regarding system suppliers discussed earlier in this book. If the outsourcing company operates in a way that results in regulatory noncompliance, then the contracting pharmaceutical or healthcare company will have a regulatory compliance issue as well. It is the responsibility of the pharmaceutical or healthcare company to find suitable business partners.

Planning and Supervision

Good contract management is vital for successful outsourcing. The following checklist is based on material from David Begg Associates:[26]

- Prepare a written statement of requirements for the outsourced company to tender against (make sure there are no misunderstandings before work starts).
- Identify what needs to be done to minimize cost and ensure that the necessary information and expertise remain in-house.
- Provide ongoing compliance oversight of activities being outsourced.
- Develop exit strategy just in case outsourcing relationship irrevocably breaks down.

QA should be involved at the outset in helping to define compliance requirements. Clear responsibility needs to be given to particular QA departments to ensure ongoing provision of resource for review and audit activities. There also needs to be a clear escalation process for the QA function to progress on any compliance issues identified.

Organizational Capability

The outsourcing company should have a designated Quality Manager and Quality Management System. The effective use of the QMS should be demonstrable, as to the capabilities of the Quality Manager. Outsourcing companies may need to consider recruiting suitable qualified personnel. Additional training may be required to fulfill regulatory expectations (see Chapter 4). Some pharmaceutical and healthcare companies transfer members of their organization to the outsourcing company either as a secondment or to be directly employed by the outsourcing company. It is very important that the outsourcing company's organization, structure, and culture support GxP principles.

McDowall identified documentation practices and change management as particular topics that indicate that an outsourcing IT organization may not fully appreciate pharmaceutical and healthcare regulatory expectations.[27] It is not just an issue of having SOPs or working instructions but also following them and having documentary evidence that the procedures are being followed. Software engineers are frequently not trained on GxP documentation practices. The use of pencils instead of pen; the use of typewriter correction fluid instead of marking a single strike-out and writing alongside the right information (initialed and dated) for corrections; and the use of Post-it notes and regulatory information written on scraps of paper are commonplace in many IT departments. The documentation of changes is also often poor. Historical practice within the outsourcing organization may not be sufficient. Documentation is often incomplete and not detailed enough, missing review and approvals, and lacking rigor of change specification and testing. Changes must be fully tested and approved before being implemented. Training, documentation, and change management, together with configuration management, self-inspection, and managing deviations (as discussed in Chapter 4) are vital supporting validation practices. It is important to demonstrate unequivocally that they work well to ensure any potential regulatory inspection. Poor practices will totally undermine a regulatory authority's confidence that the computer systems they are inspecting are being effectively and compliantly managed.

The pharmaceutical or healthcare company together with the outsourcing company should anticipate possible regulatory inspection. Consideration should be given as to whether the outsourcing company is inspection ready and would know how to handle an inspection or inspection request. Regulatory inspections and knowledge management are discussed in Chapter 16.

Disentanglement

A process of disentanglement usually has to be undertaken in order to transfer systems to the outsourcing company. Compliance issues can be divided into the following categories:

Systems Management

The operation and maintenance of regulated computer systems has already been discussed in Chapter 12. The outsourcing company should effectively manage these requirements:

- Performance monitoring
- Repair and preventative maintenance
- Upgrades, bug fixes, and patches
- Data maintenance
- Backup and restoration
- Archive and retrieval
- Business continuity planning
- Security
- Contracts and Service Level Agreements (SLAs)
- User procedures
- Periodic review and revalidation

Records Management

Compliance issues affecting the management records held on the outsourced computer systems can be summarized as follows:

- Records retention
- Records retrieval
- Access controls
- Data integrity

Contracts should specify that the outsourcing company will maintain historical records for a retention period defined by the pharmaceutical and healthcare company. Means to ensure timely record retrieval also need to be established. Among other activities, record retrieval will be required to support:

- Audits from the pharmaceutical and healthcare company
- Inspections by regulatory authorities
- Critical quality operations like recall, customer complaints, and batch investigation

The administration of access controls is usually passed to the outsourcing company. Access must be restricted to authorized users. Users may be from both the pharmaceutical or healthcare company and the outsourcing company. Access controls must protect information from unauthorized modification (inadvertent or malicious). Extra controls may be required so that previously "closed systems" do not unwittingly become "open systems." Security in general is discussed in Chapter 12. Open and closed systems are discussed in Chapter 15.

Data maintenance practices to assure data integrity and detect corruption should be instituted if they are not already established. Change control and audit trails are key aspects requiring management. Reference should be made to Chapter 12 where data maintenance is discussed in more detail. In addition, there may be electronic record management requirements; more information regarding regulatory expectations in this regard can be found in Chapter 15.

Ongoing Oversight

It is important to agree at the outset on management and controls concerning security, confidentiality, intellectual property, documentation ownership, and compliance oversight. These topics should be included in legal contracts defining the outsourcing service to be provided.

The pharmaceutical and healthcare company's QA staff should retain ongoing involvement in the following key compliance activities:

- Approve the outsourcing company's Quality Plans so that compliance requirements are visible and understood from the outset.
- Review work at regular agreed intervals.
- Audit the work against agreed plan and standards.
- Manage modifications through change control (ensure appropriate level of participation from pharmaceutical, healthcare, and outsourcing companies).
- Ensure outsourcing company completes and properly organizes all validation documentation.
- Conduct periodic compliance reviews as part of any contract renewal process.
- Keep the outsourcing company up to date with regulatory developments and compliance expectations (possibly conduct tailored training programs).
- Monitor knowledge retention in the outsourcing company and in the pharmaceutical/healthcare company's organization concerning the use and validation of relevant computer systems.
- Define and use problem escalation and resolution processes as appropriate, and not let compliance issues remain unresolved.

Pharmaceutical and healthcare companies should not assume that the outsourcing company will conduct particular activities unless it is defined in service agreements. At least one major pharmaceutical company has fallen foul of this principle, resulting in its "world class" outsourcing company not doing some "good practice" configuration management and documentation for system modification because these activities were not specified as required in its contract. In the end, the pharmaceutical company had to replace the computer system concerned because retrospective validation was deemed too expensive.

STANDARDIZING COMPUTER APPLICATIONS

Standardized computer applications are defined here as those using common software across a number of installations (e.g., use of COTS products and shared use of custom applications across multiple sites). Corporate computer system strategies of many pharmaceutical and healthcare companies are now based on the use of standard software because of the advantages it offers:

- ***Standard Release Documentation:*** The specification and testing documentation is shared among many installations so its unit cost per application should be less than that for bespoke software.
- ***Wide User Base:*** A large user community implies that if there are any problems they will be discovered quickly and rectified (i.e., market tested).
- ***Less Effort to Validate:*** Leverage on central development so that less supplementary work is required by end users.

Approach to Standardized Software Validation

The approach to standardized software should follow a variant of the V-Model called the X-Model (see Figure 14.1). Assuming that the standardized software has been developed under a suitable quality management, the end user validation can be abridged from the full bespoke software validation life cycle.

Getting the right balance between end user validation, system development, and development testing is vital. User validation should concentrate on the end application and therefore include the following:[28]

- System specification (refer to but do not repeat standard software documentation)
- Configuration details including any macros used to build the application
- Definition and testing of any customization including bespoke developments
- Verification of critical algorithms, alarms, and parameters

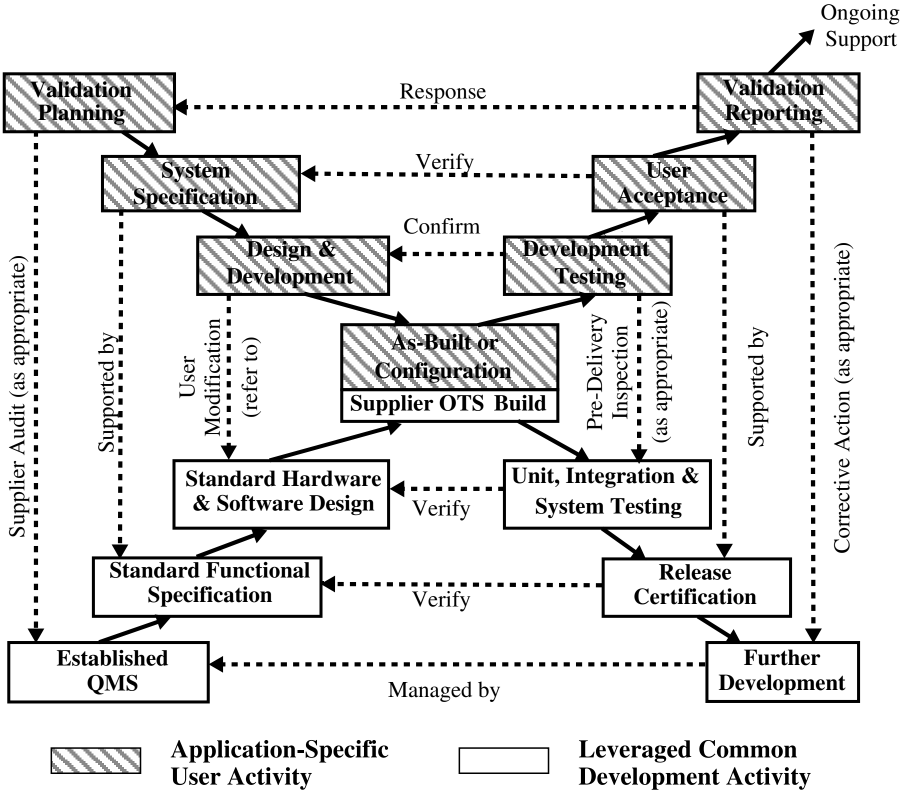

Ongoing Support
Validation Planning
Response
Validation Reporting
System Specification
Verify
User Acceptance
Confirm
Design & Development
Development Testing
As-Built or Configuration
Supplier OTS Build
Supplier Audit (as appropriate)
Supported by
User Modification (refer to)
Pre-Delivery Inspection (as appropriate)
Supported by
Corrective Action (as appropriate)
Standard Hardware & Software Design
Verify
Unit, Integration & System Testing
Standard Functional Specification
Verify
Release Certification
Established QMS
Managed by
Further Development
▨ Application-Specific User Activity
□ Leveraged Common Development Activity

FIGURE 14.1 X-Model Life Cycle for Standardized Software.

- Integrity, accuracy, and reliability of static and dynamic data
- Operating procedures being complete and practical
- System access and security

The relationship between user validation and development of the standard application must be clearly understood and described in an application's Validation Plan. Users should review and accept standardized application release documentation. Supplied documentation must match the version of the standard software being implemented. Table 14.2 suggests the general split in documentation between a user validation and standardized application document.

Access agreements should be established that support regulatory inspection of any software and documents not released to the user. Figure 14.2 indicates what documentation should be held by whom when dealing with COTS software.

TABLE 14.2
Documentation for Standardized Software

User Validation Documents	Standardized Application Release Documents
Validation Plan	Quality Plan
User Requirements Specification	Product Specification
Functional Specification	Product Design
Configuration Details	Program Specifications
Design Review	Source Code Review
Installation Qualification	Development Testing
Operational Qualification	Product Release Certification
Performance Qualification	Change Control
Validation Report	Product Development Plans
Change Control	Service Level Agreements/Warranties

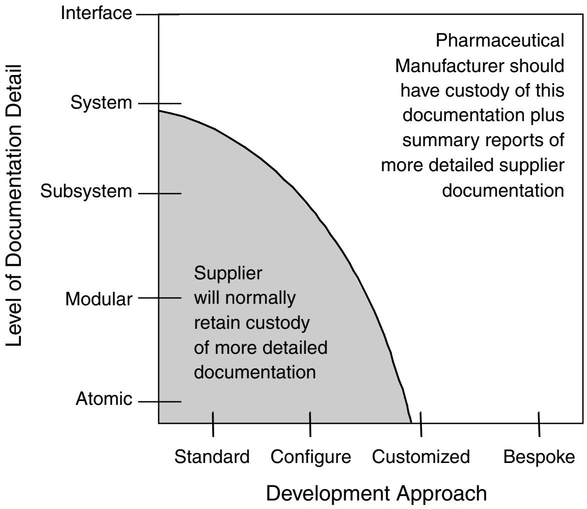

Interface
System
Subsystem
Modular
Atomic
Level of Documentation Detail
Pharmaceutical Manufacturer should have custody of this documentation plus summary reports of more detailed supplier documentation
Supplier will normally retain custody of more detailed documentation
Standard
Configure
Customized
Bespoke
Development Approach

FIGURE 14.2 Custody of Documentation.

Managing User Modifications

It is important to understand that users are often tempted to modify standardized applications and thereby undermine the standard status. There are basically four types of modification that need to be managed:

- ***Configuration:*** Setting process parameters and process paths. This modification does not impinge on the standard software status.
- ***Customization:*** Rewriting portions of standardized application code to meet specific user requirements. This modification makes the standardized application nonstandard. Detailed specifications and structural (white box) testing will be required for the modifications and other aspects of remaining system functionality altered by the change.
- ***Bespoke Element Developments:*** Writing extra software to complement the standardized application. These modifications may impinge on standard software status, but can be compensated by overall functional (black box) testing. Bespoke code must itself be fully validated, including structural (white box) testing.
- ***Upgrade Versions:*** Caution is needed when implementing new versions or bug fixes to standardized applications. Release documentation should confirm continued quality of software. If serious doubts exist over software quality, commonsense should prevail and the software should be treated as customized or entirely bespoke, and hence require full validation.

If the standard status of software has been compromised, the following steps should be taken to recover the situation:

- ***Review and Document Concerns:*** Do not hide or ignore issues. Quality and validation after all are really about good business sense; if there is a problem, fix it in the most appropriate way.
- ***Determine and Document Action Plan:*** Identify supplementary work that can be undertaken to compensate for any concerns. This may be achieved through a Risk Assessment process.
- ***Raise Concerns with Supplier:*** A Supplier Audit should be considered for external suppliers, possibly positioned as free consultancy on pharmaceutical and healthcare requirements. Be realistic about corrective action planning. Prioritize where effort needs to be placed.
- ***Work with Supplier:*** Possibly offer ongoing free consultancy. For critical applications it may be worth considering the placement of one of the customer's quality engineers in the supplier organization to help the supplier understand and address issues.
- ***More User Acceptance Testing (Qualification):*** Increment rigor of user testing commensurate with application to improve confidence in software.
- ***Replace Application:*** Finding an alternative source of supply may be necessary as the only practical solution to longer-term compliance. Pharmaceutical and healthcare companies should not disregard this option out of hand.

Software Reuse

Pharmaceutical and healthcare companies and suppliers are faced with the task of balancing increased programming efficiency offered by reuse and the potential hazards reuse may incur. It has been suggested that the reuse of small amounts of software can actually introduce more problems than writing the whole application from scratch because the new software must fit around the reused software. To reap the dividend of reuse, it has been recommended that at least 70% of a program must consist of reused software components of proven functionality.[29] Furthermore, it must be understood that, while reused software may be configured, any customization will negate its proven component functionality and the software must be considered as bespoke for the purpose of validation. Caution is also required when considering reuse of software of unknown pedigree, or open source software. Without an audit trail to its original development, such software cannot be treated as standard software and should be subject to the more rigorous validation requirements of bespoke software.

Recent examination of some tableting PLC software revealed the original code was written in Spanish, with subsequent functional revisions in German and English before a final modification for a French application. It is important to realize with software such as this that older portions of the software may not have been developed to current validation requirements, and features from earlier versions that are no longer needed may still remain. This situation occurs quite regularly with suppliers who are asked by pharmaceutical and healthcare companies to provide standard software with a few additional features. Pharmaceutical and healthcare companies should be aware that such developments increase the validation requirements because the software can no longer be considered "standard."

A special case of reuse involves the portability of software across a range of operating platforms. Standard programming languages, communication protocols, and application environments should be significantly reducing the modifications required to adapt software for different computers and operating systems. Practitioners sometimes use the term *open systems* to describe standard software capable of running on a variety of system architectures. As noted above, however, it is important to distinguish between customized and configured software when considering the validation implications of reuse. Practitioners should not underestimate the problems they may experience with portability.

SEGREGATING INTEGRATED SYSTEMS

Use of integrated applications increases the complexity of the overall "system" that in turn impacts the complexity of the validation required. In some cases, it is difficult to conclusively demonstrate that functions not requiring validation do not affect functions that do need validation. This situation often leads to increases in the scope of validation to include functions, which taken separately on their own merits, would not be considered as requiring validation.

Isolating GxP Functionality for Validation

A strategy for segregating integrated systems into those requiring validation and those that do not is considered here. This strategy can be extended to segregating distinct modules in large computer systems such as MRP II systems. A clear definition of system/module boundaries is required. This often prompts additional validation efforts for automated and manual interfaces.

Individual computer systems should be validated when they are either:

- Creating, modifying, or deleting GxP master data
- Used for GxP processes and functions
- Providing GxP data to other systems for use in GxP processes and functions

Interfaces should be validated when GxP data is being output from or input into those computer systems identified using the above criteria.

The identification of GxP processes and functions has already been discussed in Chapter 7 as part of GxP Assessments. Validation Determinations Statements should be prepared for each system to document the rationale for situations where validation is and is not deemed necessary. Validation would then be conducted for those systems and modules that require it as described in Chapter 6 through Chapter 13.

It may be appropriate in some circumstances to implement and validate independent monitoring systems for critical GxP processes rather than validate the primary system. Chapter 7 provides guidance on identifying critical components and devices where this approach is appropriate.

Validation is not required for individual systems that have no GxP functionality. However, the following controls are expected across the integrated systems to protect the integrity of the validated systems:

- Contemporaneous management of GxP data is replicated in multiple systems.
- The integrated architecture of systems is robust against individual system failures.

Change control during operation and maintenance must assess and verify that the rationale for validation is not affected by modifications to individual systems. The use of individual systems often changes over time, and at some point it is possible that a non-GxP system may be used in a GxP context. It is important not to inadvertently undermine the validation rationale for the overall integrated system.

Separating Computer Network Infrastructure

Validating applications and the computer network infrastructure separately should reduce potential duplication of testing of common infrastructure shared by multiple applications. GxP applications should be validated as outlined in Chapter 6 through Chapter 11. Testing multisite applications can be based on a comprehensive test at a single site of shared functionality across multiple sites. In addition, separate tests may be needed to test site-specific functionality. OQ testing should include at least one test to verify operability from each user site.

Computer network infrastructure should be qualified in support of validated applications. Bristol Meyer Squibb have adopted a three-level model to assist the qualification of their computer network infrastructure.[30] This approach is summarized in Table 14.3. Layer 1 comprises computers that provide shared resources such as servers, hosts, mainframes, and mini computers. Layer 2 is the network infrastructure (e.g., hubs, routers, and switches). Layer 3 comprises the user desktop environment (i.e., workstations, personal computers, and laptops).

Functional specifications should be developed for the host machine, its operating system, and utilities. The scope will include the use of any servers. Design documentation should cover the actual configuration and setup of the computing hardware and associated equipment.

IQ needs to cover both hardware and software aspects. Hardware installation of the host computer should be documented with the installation method. Components added to standard hardware should also be recorded (e.g., memory, NIC card, and hard drives). Operating system

TABLE 14.3
Infrastructure Qualification Documentation

Validation Documents	Layer 1	Layer 2	Layer 3
Functional Specification	Y	Y	Y
Design Documentation	Y	Y	N
Installation Qualification	Y	Y	Y
Operational Qualification	Y	N	Y
Performance Qualification	N	Y	N
Summary Report	Y	Y	N

details together with any patches and upgrades must be documented. For larger systems, particular use of modules, utilities, or library functions should also be recorded so that the software environment is defined.

OQ should include backup and recovery, data archive and retrieval, security, system administration procedure verification, startup and shutdown, UPS continuity, communications loss and recovery, and systems redundancy challenges such as mirrored drives, secondary systems, and fail-safe systems.

PQ of the network should cover loading tests as appropriate to verify network performance. Such testing is not always appropriate as PQ and may be included instead as part of ongoing performance monitoring.

A final summary report should be prepared for the computer network infrastructure to summarize the results of the qualification exercise. A case study on computer network architecture is provided in the second part of this book.

RETROSPECTIVE VALIDATION

Computer systems should be validated prospectively. It is not generally acceptable to implement a computer system and attempt to validate it after it has been installed for use. This said, where a system has had a change in use to bring it within scope of an existing validation related regulation, or new validation related regulations have been introduced such as U.S. CFR Part 11 to include the computer system within their scope, then retrospective validation is acceptable.

Validating existing systems, however, can be more than five times more expensive than if that same system had been validated when it was new. Practitioners should, therefore, consider whether it is cheaper to implement a replacement system rather than conduct retrospective validation.

SETTING PRIORITIES

It is often necessary to prioritize validation projects when validating the backlog of existing systems. Priorities for validating different computer systems should be set according to a defined strategy. Some projects may be given a higher priority because a regulatory inspection that is likely to include the system is imminent, or there are outstanding noncompliance issues from a previous regulatory inspection, or the computer system is supporting a process subject to a new drug regulatory submission. Equally, a lower priority may be given to computer systems that are soon to be replaced. Some pharmaceutical manufacturers, for instance, when prioritizing the validation of their existing computer systems, decided not to validate those systems due for replacement within a year. If such a stance is taken, it is important that the system is replaced within the stated time frame. It is all too easy to delay the replacement of a system so that it is permanently to be replaced within the year — such situations are not acceptable to the GxP regulatory authorities.

The first step in determining an order of work is to define levels of risk and system characteristics that affect risk. Individual computer systems can then be classified against the set criteria and a weighted risk factor calculated. The state of existing validation is then calculated and subtracted from the weighted risk factor to give a compliance gap. The compliance gaps can then be compared between systems to order work.

Three levels of risk are suggested here (low, medium, and high) although some pharmaceutical and healthcare companies may like to consider five levels of risk to match the system integrity levels defined by IEC/ISO 61508 for safety critical systems. Each system should be rated against a number of weighted risk factors to determine an overall level of risk. Seven example risk factors are considered in Table 14.4:

- System Development
- Security Practice

- Performance History
- Support Service
- Visibility of Use
- Regulatory Exposure
- Remaining Life

Multiplying the score for each row in Table 3.2 with its corresponding weighting and taking the sum across all the rows yields a total that can be used to determine the level of risk. Total scores of between 21 and 35 are considered a LOW risk, scores of between 36 and 49 are considered a MEDIUM risk, and scores of between 50 and 63 are considered a HIGH risk. A worksheet should be developed to log the risk assessment. It must be stressed that Table 3.2 is given only as an example. Pharmaceutical and healthcare companies should give careful consideration as to which risk factors and weights are best suited to their business.

The state of validation for each computer system can be determined from examining its associated documentation. The examination is not intended to be a detailed review. Rather it should be a rough-cut evaluation delivering a quick result. Locating and retrieving what documentation exists is likely to be a much more time-consuming task than the examination of the documentation itself. Documentation should be marked according to a scale such as 1 — Does not exist, 2 — Exists but needs work to fulfill current regulatory requirements, 3 — Exists and is adequate to fulfill current regulatory requirements. Document names will vary between systems; generic document types for guidance are suggested in Chapter 4. Again, worksheets should be developed to log the document examination. The sum of marks given for the generic document types provides the state of validation.

The compliance gap is calculated by subtracting the "state of validation" score from the maximum possible "risk assessment" score for that system's level of risk. The maximum possible "risk assessment" scores for LOW, MEDIUM, and HIGH risk systems are 35, 49, and 63, respectively. To avoid negative scores the state of validation assessment should be designed so that its maximum score is equal to or less than the maximum possible "risk assessment" score for a LOW risk system. The compliance-gap score can be included in the system inventory. The priority attached to validation should be based on tackling the systems with the highest compliance-gap scores first.

Completion of retrospective validation across a number of computer systems, whether by remediation or replacement of individual systems, should be achieved within 2 to 3 years from the outset of the overall program of work. Status reports should be periodically prepared to demonstrate progress. It may be useful to extend the inventory of systems discussed in Chapter 3 to include a status flag indicating whether retrospective validation is outstanding or in progress.

Hazard Control

When prioritizing validation, it is important to consider critical dependencies on particular computer systems. Hazards must be controlled. A stepwise approach to Hazard Control is given below:

- Assess each computer system to determine whether or not it can influence the strength, identity, security, purity, or quality of a drug product. The assessment should be conducted in accordance with a defined process and the outcome of each assessment recorded.
- Precisely how a computer system impacts drug product attributes should be documented. Those computer systems that impact drug product attributes require validation. The decision to validate or not to validate should be approved by an authorized person as part of the validation determination.
- Validation should place a priority on critical processes and their associated computer applications. All computer systems should be considered critical unless reliance can be placed on an independent downstream system. A downstream system may be a manual

TABLE 14.4
Example Risk Factors and Weightings

Risk Factors	Low Risk (Score 1)	Medium Risk (Score 2)	High Risk (Score 3)
System Development (Weighting ×1)			
Standard Software	Commercial Off-The-Shelf (COTS) application	Used in complex or critical application	Not applicable
Configuration	Not applicable	Only parameters set, no bespoke code	Bespoke macros or customization
Customization	Not applicable	Not applicable	Customize software
Bespoke Application	Not applicable	Not applicable	Bespoke software
Security Practice (Weighting ×1)			
Physical Access	Restricted by physical barrier (e.g., locked room)	Restricted by location only (e.g., panel key, removed keyboard)	No restrictions
Logical Access	Different levels of password access for users and system administrator	System protected by single level of password access	No password protection in use
Virus Management	Automatic	User dependent	No management
Performance History (Weighting ×1)			
Downtime	< 1 h (or one occurrence) per year	< 1–8 h (or 1–5 occurrences) per year	> 8 h (or > 5 occurrences) per year
User Changes and System Upgrades	None within last year None planned	< 3 user changes and < 1 system upgrade in last year	> 3 user changes and/or > 1 system upgrade in last year Some planned
Support Service (Weighting ×2)			
Supplier Capability	QMS and SLA	QMS or SLA	No QMS or SLA
Staff Turnover	< 3%	4–8%	> 8%
Dependency on Contractors	< 30% of staff	30–50% of staff	> 50% of staff
Spare Parts	Spares and/or alternate system available on-site	Only available off-site < 24 h (unless cannibalize?)	Only available off-site > 24 h; no alternative supply
Data/Software Backups	Regular backups	Infrequent backups	No routine backups
Visibility of Use (Weighting ×3)			
Criticality	GxP functionality	Analytical results	Batch release and recall
Size	1–3 users Process control systems <100 I/O	4–10 users Process control systems 100–500 I/O	More than 10 users Process control systems >500 I/O
Replication	One-of-a-kind application on site	Multiple systems on-site used in same or similar manner	Application running multiple sites in same division of company
Regulatory Exposure (Weighting ×4)			
Inspection History	Not covered by or no comments from last inspection	Observations from last inspection	Critical observations from last inspection
Submission	Not applicable	No new submissions, general inspections still expected	Preapproval inspection (PAI)/expected < 1 year
GxP Application	Not applicable	Indirect application	Direct application
Remaining Life (Weighting ×2)			
Expected Remaining Operational Life	Planned withdrawal within 2 years	Anticipated life approx. 3–5 years	No planned replacement

system, a further computer system, or a nonsoftware-based item of equipment. Whether individual computer systems are critical or not must be stated on their validation determination.

- Where there is reliance on an independent downstream system, this system must be considered critical. Downstream systems based on computer systems must be validated. Downstream systems based on manual ways of working and nonsoftware-based items of equipment should be periodically challenged at suitable intervals during its operational life.
- If the downstream system is a checking device and is not a separate computer system (i.e., it forms part of the functionality of the computer system under review), then the whole system including the checking device must be considered critical. A regime of sampling the output of the computer system will not be accepted as a downstream quality check.
- A remedial action plan is required where a compliance gap is determined against a computer system's validation requirement.
- Where a significant compliance gap is identified for a critical computer system, the remedial action plan will need to consider whether replacement of the computer system is more cost-effective than revalidation.
- Once critical computer systems are validated, the remaining computer systems should be validated.

Hazard Control can help focus effort and thereby rapidly establish significant GxP improvements. This is likely to be especially important where skilled resource and/or available time to address validation are limited.

Interim Measures

Interim measures are additional controls applied in relation to computer functionality that support critical quality-related activities. They are implemented where compliance gaps are considered to exist, to provide added assurance of control, and to justify the continued use of a computer system. Interim measures are used to supplement or replace defined computer functionality. Examples of interim measures include:

- Independent manual procedures used in parallel to support computer system functionality
- Comparison of data sampled from specific functions with independently derived data
- Independent computer systems to monitor critical quality-related activities
- Independent downstream computer systems to detect quality failures
- Combination of the above

The type of interim measure implemented should be appropriate to the computer functionality being addressed. Computer functionality being addressed should be mapped so that appropriate interim measures can be identified. The mapping should include both a workflow analysis and a dataflow analysis. Controls that are already in place may provide the basis for the interim measures. Critical activities that should be given particular consideration for interim measures include:

- Stages in the operational process where status change occurs such as approval of a raw material or intermediate product
- Critical processing activities that are reliant on computer systems such as dispensing
- Label information and printing
- Product quality-related specifications held by or used by computer systems

- Approval of product to release to the market
- Access points where GxP data can be modified or deleted

Interim measures do not eliminate the need for full corrective actions; they do not resolve actual computer system compliance issues. Full corrective solutions must still be planned and implemented to bring computer systems into compliance. If interim measures are implemented, this activity must be properly planned and must form part of an overall plan to install permanent corrective solutions. Interim measures should be kept as simple as possible.

Validation

The following checklist is based on work by the German APV for practitioners validating existing computer systems that were not, or were only partially, developed in accordance with validation requirements.[31] Some practitioners prefer to use the term *retrospective evaluation* to highlight that the exercise is founded on the principle of a compliance gap analysis and consequential remedial actions. It is important to realize that any retrospective validation takes more effort than prospective validation and rarely achieves the same standard.

- Freeze the computer system to stop any changes during revalidation.
- Conduct a compliance gap analysis on the GxP-relevant components and functions of the system with reference to the past operational experience. Assess the completeness of documentation, outstanding internal audit observations, and outstanding regulatory commitments.
- Stop or justify the continued use of the computer system.
- Prepare a Validation Plan.
- Create/revise the documentation describing the computer system.
- Conduct a Design Review.
- Inspect critical application software, conduct an IQ, conduct an OQ with emphasis on GxP component and functions of the system, and conduct a PQ.
- Prepare a Validation Report.
- Release the computer system for use, if necessary implementing system modifications and additional organizational measures under change control.

The general approach to retrospective validation is the same as for prospective validation (see Chapter 6 to Chapter 11). However, it may not be possible to conduct some prospective activities such as Supplier Audits if the supplier is no longer trading, Source Code Reviews if there is no access to source code and relevant design documentation, and Development Testing if detailed design information is not available. Historical records demonstrating reliable operation may be available to aid validation.

The content and structure of Validation Plans should fulfill the recommendations outlined in Chapter 6. Validation Plans usually have an additional section giving a brief history of the system from its original procurement, through any developments, to the current system configuration. The Validation Plan should indicate the new and existing documentation that will be used to support validation of the computer system. If original design and development documentation is missing or the change history is missing or incomplete but there is evidence to demonstrate ongoing reliable operation, then the computer system can be treated like a software of unknown pedigree (see relevant comments in Chapter 8 and Chapter 10).

Some pharmaceutical and healthcare companies combine the intent of URS and Functional Specification when conducting retrospective validation into a document called a System Specification. The System Specification will include a statement to the effect that the document represents not only a description of the system in use but also that this description fulfills user requirements for the system. Although the original design intent of the computer system may have changed, it may not be necessary to totally rewrite existing specification documents. Instead, it may be possible to write a short frontispiece to existing documents, defining the changes and their impact on the original design.

Supplier Audits should be conducted where practical for bespoke and critical applications. Emphasis will be placed on the level of support available from the supplier. Remember that the supplier may be a function within the pharmaceutical or healthcare company's organization. In such instances, the Supplier Audit becomes an internal audit and document search.

Software and hardware design documentation may have to be reverse engineered, both at module and system level. The GAMP Special Interest Group on Legacy Systems recommends reverse engineering only for custom (bespoke) software elements; COTS software at this level only needs configuration to be defined.[31] Software logic flows should be described and flowcharts developed as appropriate. All algorithms need to be defined. Hardware configuration items should be listed.

A Design Review should be conducted before testing begins. This will normally involve developing a Requirements Traceability Matrix (RTM). If no detailed design information is available then cross-references should be made between the newly prepared System Specification, available operator manuals, and user procedures. Source Code Reviews will be expected for custom (bespoke) software under the control of the pharmaceutical or healthcare company, and redundant code identified should be removed.

Development Testing by definition for an existing system should have already been conducted, although original test records may be incomplete, insufficient, or missing. Test protocols should be reviewed to ensure that they reflect the current operating environment. Some pharmaceutical and healthcare companies take the opportunity to supplement their User Qualification with additional tests to unit, system, and integration tests that might otherwise be conducted as a separate activity.

User Qualification should comprise of IQ, OQ, and PQ. The IQ effectively baselines the system for OQ and can be conducted while the system is making pharmaceutical and healthcare grade products. The OQ should cover all functional aspects now defined in the System Specification. Some OQ testing such as safety-related test and disaster-recovery tests may have to be delayed until a planned facility shutdown takes place. Some facilities may not have a planned shutdown for more than a year, in which case consideration should be given especially to planning one for the validation project. The final phase of qualification, the PQ, can use, but must not rely solely upon, historical evidence of dependable operation. Retrospective product PQ should be conducted over larger samples rather than prospective product PQ. For instance, it has been suggested that the product PQ should review at least 30 batches of manufactured drug products.

Procedures and user manuals may be outdated, with users relying on typed or handwritten instructions to supplement or replace old manuals. Procedures for operating the computer system should be reviewed and updated as necessary to reflect the current use of the system. Training records should be current and reflect training in these updated procedures. Access rights should be checked as appropriate and authorized. Role specifications may need to be updated. Business Continuity Plans should also be reviewed and amendments made as required.

Finally, a Validation Report should be written in reply to the Validation Plan. Internal and third-party Service Level Agreements may need to be established to ensure that validation is maintained. Arrangements for effective change control and configuration management must be put in place.

Recent Inspection Findings

- Retrospective validation may be conducted for a well-established process used without significant changes to [drug product] quality due to changes in raw materials, equipment, systems, facilities, or the production process. This validation approach may be used where
  1. Critical quality attributes and critical process parameters have been identified

2. Appropriate in-process acceptance criteria and controls have been established
3. There have not been significant process/product failures attributable to causes other than operator error or equipment failures unrelated to equipment suitability
4. Impurity profiles have been established for the existing [drug product]

- Once an existing process has been validated retrospectively, and the process needs to be revalidated due to changes that may affect the quality of a [drug product], the validation should be done prospectively, or in certain limited cases, concurrently. Most important, these changes should be controlled by a formal change control system that evaluates the potential impact of proposed changes on the quality of the [drug product]. Scientific judgment should determine what additional testing and validation studies should be conducted to justify a change in a validated process. [FDA Warning Letter, 2000]
- It could be difficult to retrospectively validate a computer system if there were changes and revisions that were not documented and the cumulative effects of many revisions had not been assessed. Lack of sufficient system documentation would make it impossible to perform meaningful retrospective validation. FDA concludes that the XXX and YYY systems lack adequate validation and therefore are unacceptable for use in the production of drug products. Please indicate whether you can perform a retrospective validation of XXX and YYY systems or rely in the interim on manual operations, which use source documentation until the new validated computer systems are functional. [FDA Warning Letter, 2001]
- Manual verification of calculations and inventory checking with the existing computer software that has been found to be problematic is not an adequate reason for lack of validation. Existing computer software should be validated or replaced. [FDA Warning Letter, 2001]
- Validation is incomplete, e.g., mentions “historic evidence” without explanation or supportive documentation. [FDA Warning Letter, 1999]
- We continue to find proposed timeline to complete validation of the XXXX system to be unacceptable. The XXXX system should not be in use unless they have been completely validated to current standards. [FDA Warning Letter, 2002]
- Software “bug” that could result in erroneous release not scheduled for correction … Headquarters has allowed a workaround for a software problem to be in place for 8 years. [FDA 483, 2002]

STATISTICAL TECHNIQUES

When statistical sampling is used it is recommended that professional statistical support is used rather than relying on *ad hoc* advice. It is vital that statistical techniques are used appropriately.

APPROACH TO PROJECTS

Statistical sampling can be considered part of a testing strategy for projects implementing/deploying multiple systems that are the same or very similar (i.e., within an acceptable delta). A similar approach, sometimes referred to as matrix validation, is used in the context of validating manufacturing equipment and processes.

The determination of the sample size must be documented. An important aspect to consider in applying statistical sampling is the need to predefine the acceptability of “similar systems.” If the systems and their operational environment are exactly identical then a sample size of one may be sufficient. If the systems are not identical, then consideration needs to be given to what is an acceptable delta for the differences between those “similar systems.” Some of the deltas that one can consider may include the differences in software (operating systems, third-party tools, application program) version, patches, and fixes, as well as the deltas in hardware and equipment, instrument, or other peripheral that are the components of the system. Great care must be taken in justifying an acceptable delta. Computer systems should be considered separate applications and validated accordingly when there is significant variation.

Approach to Data

Data checking can be a resource-intensive process. Statistical sampling can provide a viable method to reduce the effort, resources, and time required to check data while retaining a high degree of assurance that the required level of data accuracy is being maintained.

Data can be classified into different types, each type with a different level of acceptable accuracy. Three basic classifications are described here by way of example:

- Critical Data (includes GMP data) are required to be 100% accurate. This can only be established by a 100% check, preferably independently by two persons, to minimize the likelihood of mistakes, for example, due to fatigue and other random errors.
- Significant Data (if this is to be distinguished from critical data) are required to have a predetermined acceptable accuracy (e.g., has a maximum of 5% error rate). This can be established by a randomly drawn sample so long as a small risk is accepted that, even though the sample strongly indicates that the error rate is below the predetermined acceptable level, in fact the "true" error rate is above the predetermined acceptable level. This is an inevitable consequence of using a sample. The only alternative is a 100% check, as above.
- Other Data (which can be divided up into further subcategories) are required to have a predetermined acceptable accuracy (e.g., have a maximum of 25% error rate). This can be established as above for significant data by a randomly drawn sample so long as a small risk is accepted.

The objective of statistical sampling is to establish likely values for the "true" error rate in the population of data being considered. If the "true" error rate was known, the probabilities of given numbers of errors in samples could be obtained mathematically using standard statistical distributions. Statistical inference allows the reverse process — from an observed error rate in a sample likely and possible "true" error rates can be inferred. Likely data population error rates are defined by the 99% single upper confidence limit, and possible data population error rates by the 99.9% single upper confidence limit on the sample error rate.

Large populations of data (in excess of 5000 items) can be regarded as infinite and thus a binomial approximation to the hypergeometric distribution can be applied. It is assumed that errors occur randomly throughout the data population. If data within the population has been obtained from different sources in different ways, there may be an expectation that error rates for these subpopulations may differ. If this is the case, the data population should be split into "strata" and analyzed separately. Note that for populations less than 5000 items it is recommended that all items be checked rather than a sample taken.

The *likely error rate*, as stated earlier, is defined as all values less than the 99% single upper confidence limit on the population error rate. That is,

$$100 * \{p + 2.3263 * \sqrt{[p(1-p)/N]}\}$$

If extra assurance is required, the *possible error rates* are defined by the 99.9% single upper confidence limit on the population error rate. That is,

$$100 * \{p + 3.0902 * \sqrt{[p(1-p)/N]}\}$$

where p is the observed proportion of errors in the sample and N is the sample size.

Tables in Appendix 14A are provided to support the statistical analysis. Extra tables can be easily developed to support other error rates and smaller data populations, if need be. To determine the required sample size from the tables follow the steps below:

1. Select the target error rate (5% or 25% for the tables provided).
2. Select the observed error rate that is believed likely to become true and use that (rounding up as necessary) to choose a column in the table. Rounding up will give a sample size larger than is strictly required but makes it easier to use the table.
3. Identify the smallest sample size so that the chosen column gives a likely error rate that is less than the target error rate (e.g., an observed error rate of 3.5% is applicable to the table for error rates not exceeding 5%, and yields a sample size of 1050).
4. Obtain a random sample of this size and measure the error rate. Note that the sample must be (effectively) random in order to avoid potential bias from unknown or ignored influences on the data population.
5. If the observed error rate in the sample is equal to or less than the predefined acceptable level, no further action is required. Nevertheless, it is recommended that the opportunity be taken to correct any errors found and to investigate any commonalties between the errors, to identify any root cause that might affect the rest of the data population.
6. If the observed error rate is greater than the predefined acceptable level, repeat step 3 using the observed error rate. Note that part of the required sample has already been taken. In the example given in step 3 if the observed error rate is 4%, a further sample of 1050 is required.

The tables with *likely error rates* will normally be used unless a very cautious approach is being taken, in which case the *possible error rates* should be used.

Recent Inspection Findings

- Failure to establish and maintain procedures to ensure that sampling methods are adequate for their intended use and are based on a valid statistical rationale. [FDA Warning Letter, 2000]
- No documentation to support statistical techniques used. [FDA 483, 2002]

REFERENCES

1. U.S. Code of Federal Regulations Title 21: Part 58, *Good Laboratory Practice for Nonclinical Laboratory Studies*.
2. European Union Guide to Directive 91/356/EEC (1991), *European Commission Directive Laying Down the Principles of Good Manufacturing Practice for Medicinal Products for Human Use*.
3. PIC/S Recommendations for Validation Master Plan and Installation/Operational Qualification, 2001.
4. ICH (2000), *Good Manufacturing Practice Guide for Active Pharmaceutical Ingredients*, International Conference on Harmonisation, Harmonised Tripartite Guideline, November.
5. U.S. Code of Federal Regulations Title 21: Part 211, *Current Good Manufacturing Practice for Finished Pharmaceuticals,* plus Federal Register (1996) — Current Good Manufacturing Practice: Amendment of Certain Requirements for Finished Pharmaceuticals; Proposed Rule, 61 (87).
6. ICH (1996), *Guideline for Good Clinical Practice*, ICH Harmonised Tripartite Guideline, International Conference on Harmonisation of Technical Requirements for Registration of Pharmaceuticals for Human Use.
7. OECD (1995), Principles of Good Laboratory Practice to Computerised Systems, Organisation for Economic Co-operation and Development, Paris.
8. United Kingdom Department of Health (1995), The Application of GLP to Computer Systems, The Principles of Good Laboratory Practice, United Kingdom Compliance Programme, London.

9. U.S. Code of Federal Regulations Title 21: Part 820, Good Manufacturing Practice for Medical Devices.
10. BARQA (1997), *Regulatory Compliance and Computer Systems*, Conference Proceedings.
11. Lloyd, I.J. and Simpson, M.J. (1997), Computer Risks and Some Legal Consequences, in *Safety and Reliability of Software Based System,* Springer-Verlag, New York.
12. United Kingdom Sales of Goods Act (1974).
13. United Kingdom Supply of Goods and Services Act (1982).
14. United States Food, Drugs, and Cosmetics Act.
15. United Kingdom Supply of Machinery Regulations (1992).
16. United Kingdom Health and Safety at Work Act (1974).
17. United Kingdom Environmental Protection Act (1990).
18. United Kingdom Data Protection Act (1984).
19. United Kingdom Consumer Protection Act (1987).
20. United Kingdom Product Safety Regulations (1994).
21. United Kingdom Unfair Contract of Terms Act (1977).
22. Unfair Terms in Consumer Contracts, EU Directive 93113/EEC (1993).
23. United Kingdom Misrepresentation Act (1967).
24. FDA, Current Good Manufacturing Practices for Finished Pharmaceutical Products 21 CFR 211.25(a).
25. European Union Food Manufacturing Practice for Pharmaceuticals, Medicines Controls Agency, 1997.
26. David Begg Associates (2002), *Computers and Automated Systems Quality and Compliance*, June, 24–27, York, U.K.
27. McDowall, R.D. (2002), Regulatory Compliance Considerations When Outsourcing (Part 1 and Part 2), *European Pharmaceutical Review.*
28. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), International Society for Pharmaceutical Engineering (www.ispe.org).
29. Hatton, L. (1997), Unexpected (and Sometimes Unpleasant) Lessons from Data in Real Software Systems, in *Safety and Reliability of Software Based Systems*, Springer-Verlag, New York.
30. Williams, Y. and Torres, J. (2002), *Documentation of Infrastructure Qualification and System Validation*, IVT Conference on Network Infrastructure Qualification & Systems Validation, Philadelphia, October 8 and 9.
31. GAMP Forum (2003), *GAMP Good Practice Guide – The Validation of Legacy Systems*, International Society for Pharmaceutical Engineering.

APPENDIX 14A
ERROR RATE TABLES

TABLE 14A.1
Likely "True" Error Rates (%) for Observed Error Rates (%) in Samples of Given Sizes with Target Errors Rate of at Most 5%

	Observed Error Rate in Sample					
Sample Size	**1%**	**2%**	**3%**	**3.5%**	**4%**	**4.5%**
350	2.24	3.74	5.12	5.79	6.44	7.08
700	1.87	3.23	4.50	5.12	5.72	6.32
1,050	1.71	3.01	4.22	4.82	5.41	5.99
1,400	1.62	2.87	4.06	4.64	5.22	5.79
2,100	1.51	2.71	3.87	4.43	4.99	5.55
2,800	1.44	2.62	3.75	4.31	4.86	5.41
3,500	1.39	2.55	3.67	4.22	4.77	5.32
7,000	1.28	2.39	3.47	4.01	4.54	5.08
10,500	1.23	2.32	3.39	3.92	4.44	4.97
14,000	1.20	2.28	3.34	3.86	4.39	4.91
17,500	1.17	2.25	3.30	3.82	4.34	4.86

Note: %. True error rate is defined at 99% single upper confidence limit.

TABLE 14A.2
Likely "True" Error Rates (%) for Observed Error Rates (%) in Samples of Given Sizes with Target Errors Rate of at Most 25%

	Observed Error Rate in Sample					
Sample Size	**4%**	**8%**	**12%**	**16%**	**20%**	**24%**
10	18.42	27.96	35.91	42.97	49.43	55.42
25	13.12	20.62	27.12	33.06	38.61	43.87
50	10.45	16.93	22.69	28.06	33.16	38.05
100	8.56	14.31	19.56	24.53	29.31	33.94
200	7.22	12.46	17.35	22.03	26.58	31.03
300	6.63	11.64	16.36	20.92	25.37	29.74
350	6.44	11.37	16.04	20.56	24.97	29.31
700	5.72	10.39	14.86	19.22	23.52	27.76
1,050	5.41	9.95	14.33	18.63	22.87	27.07
2,100	4.99	9.38	13.65	17.86	22.03	26.17
7,000	4.54	8.75	12.90	17.02	21.11	25.19
14,000	4.39	8.53	12.64	16.72	20.79	24.84

Note: %. True error rate is defined at 99% single upper confidence limit.

TABLE 14A.3
Possible "True" Error Rates (%) for Observed Error Rates (%) in Samples of Given Sizes with Target Errors Rate of at Most 5%

	Observed Error Rate in Sample					
Sample Size	**1%**	**2%**	**3%**	**3.5%**	**4%**	**4.5%**
350	**2.64**	**4.31**	**5.82**	**6.54**	**7.24**	**7.92**
700	2.16	3.64	4.99	5.65	6.29	6.92
1,050	1.95	3.34	4.63	5.25	5.87	6.48
1,400	1.82	3.16	4.41	5.02	5.62	6.21
2,100	1.67	2.94	4.15	4.74	5.32	5.90
2,800	1.58	2.82	4.00	4.57	5.14	5.71
3,500	1.52	2.73	3.89	4.46	5.02	5.58
7,000	1.37	2.52	3.63	4.18	4.72	5.27
10,500	1.30	2.42	3.51	4.05	4.59	5.13
14,000	1.26	2.37	3.45	3.98	4.51	5.04
17,500	1.23	2.33	3.40	3.93	4.46	4.98

Note: %. Possible error rate is defined at 99.9% single upper confidence limit.

TABLE 14A.4
Possible "True" Error Rates (%) for Observed Error Rates (%) in Samples of Given Sizes with Target Errors Rate of at Most 25%

	Observed Error Rate in Sample					
Sample Size	**4%**	**8%**	**12%**	**16%**	**20%**	**24%**
10	23.15	34.51	43.76	51.83	59.09	65.73
25	16.11	24.77	32.08	38.66	44.72	50.40
50	12.56	19.86	26.20	32.02	37.48	42.66
100	10.06	16.38	22.04	27.33	32.36	37.20
200	8.28	13.93	19.10	24.01	28.74	33.33
300	7.50	12.84	17.80	22.54	27.14	31.62
350	7.24	12.48	17.37	22.06	26.61	31.05
700	6.29	11.17	15.80	20.28	24.67	28.99
1,050	5.87	10.59	15.10	19.50	23.81	28.07
2,100	5.32	9.83	14.19	18.47	22.70	26.88
7,000	4.72	9.00	13.20	17.35	21.48	25.58
14,000	4.51	8.71	12.85	16.96	21.04	25.12

Note: %. Possible error rate is defined at 99.9% single upper confidence limit.

15 Electronic Records and Electronic Signatures

CONTENTS

Many countries have now introduced regulations governing the use of electronic records and the legal equivalence of electronic signatures to handwritten signatures. The basic requirements are based on established GxP expectations. Interpretation of the electronic record and signature regulations, and appropriate methods for achieving compliance, have been subject to much debate and discussion in the industry. This chapter discusses the practicalities of compliance with U.S. 21 CFR Part 11 on electronic records/signatures and other principal international regulatory requirements and expectations. Topics covered include:

- Practical definition of what constitutes an electronic record
- Audit trails for creation, modification, and deletion of electronic records
- Operational checks to verify authorized users
- Logical and physical security measure for access control
- Training for use of electronic records and electronic signatures
- Legal admissibility of electronic signatures
- Integrity of biometric controls where they are applied
- Validation of procedural and technical controls

ELECTRONIC RECORDS

Electronic records are defined here as those records used for GxP decision/review processes or regulatory submissions. Appendix 15A helps identify examples. Financial, Data Protection, and other non-GxP records held electronically may also have regulatory requirements, but these are not specifically covered here.

The FDA is currently developing guidance to assist understanding of what exactly constitutes an electronic record.[1] The FDA looks to predicate regulations (Predicate Rules) to identify records that when stored electronically will require electronic record controls.[2] The predicate regulations, however, were developed on the whole without this use in mind and there remains significant ambiguity in what exactly on a practical level the FDA considers as falling within the scope of definition of an electronic record (e.g., are status flags, configuration parameters, and software programs considered electronic records?). In response the FDA has suggested that risk assessments be conducted to identify those records that may impact pharmaceutical or healthcare product quality and safety and hence require special management to preserve data integrity.[3]

Other regulatory authorities expect pharmaceutical and healthcare companies to make their own determination based on published GxP regulations and guides on what critical records in their computer systems are and to apply electronic record controls accordingly.[4]

Regardless of terminology the process of identifying most important records is basically the same. Risk assessment and criticality are inextricably linked. The ISPE has distinguished high-risk and lower-risk records with a view to the risk posed to patient and consumer health.[5] Examples of high-risk records include product quality decisions, batch records, laboratory test results, and clinical trial results. Examples of low-risk records include training, computer setup, and configuration parameters. The premise is to identify primary records protecting patient/consumer health.

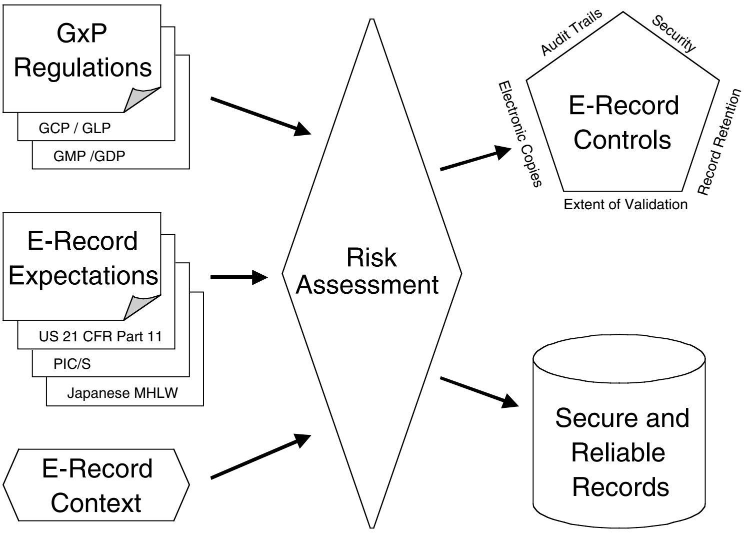


FIGURE 15.1 Electronic Record Risk Management.

The GAMP Forum has published guidance to help distinguish critical records and appropriate controls.[6] Figure 15.1 outlines the basic concept being promoted. The process can be used to identify all records requiring specific management and control. The level of control should be commensurate with the importance of the record. Computer system validation is all that is necessary for low-risk records. Particular technical and procedural controls will be needed to address high-risk records.

The risk assessment process can be conducted by examining record types to see if they are GxP or non-GxP, and then applying severity checks, likelihood, and probability of detection criteria, as illustrated in Figure 15.2. The most severe scenarios should be linked to direct patient/consumer impact. GxP noncompliance and broken license conditions are severe in their own right but not as critical as patient/consumer health in this analysis.[7] Its likelihood will be influenced by the degree of human error in how the record is input and used. The probability of detection needs to take into account the probability of the impacted record being used. Once failure modes are understood, then the appropriate design controls can be introduced. These should be documented and validated as part of the computer system life cycle discussed earlier in this book.

The FDA excuses electronic records from 21 CFR Part 11 where they are printed and it is the printed copy that is used rather than the electronic version.[3] The electronic record in these circumstances is considered incidental. The FDA will, however, challenge how such printed copies are used to determine whether in practice there is still a dependency on the electronic version. It is recommended that pharmaceutical and healthcare companies document their use of electronic and printed copies within SOPs. Printed copies must not be taken in an effort to side-step regulatory requirements.

Record Life Cycle

A data flow analysis should be conducted to identify the creation and maintenance of electronic records. The life cycle of a record is shown in Figure 15.3 (based on GERM[8]).

Electronic records are created when their component raw data is processed and stored to a durable media. From this point on, electronic records require audit trails and metadata to be maintained as discussed later. Examples of electronic raw data used to compile electronic records include calculations used to determine a sample potency range, individual temperature readings from an autoclave used to plot a temperature profile, individual points used to plot a peak in a

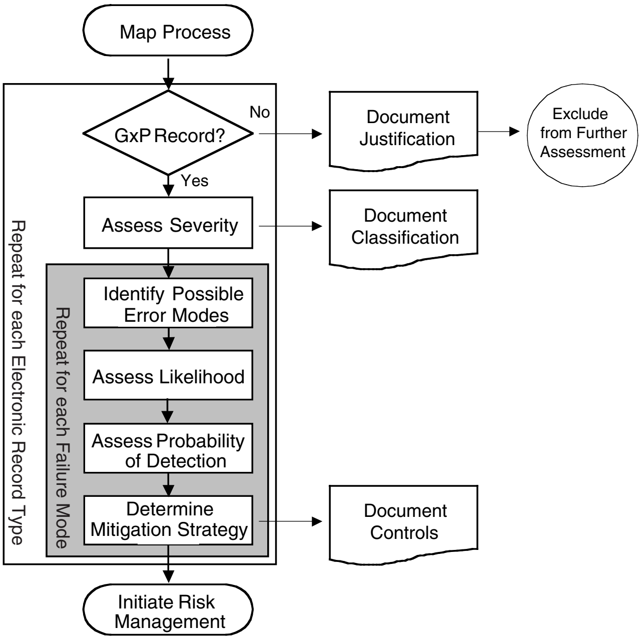

Map Process
GxP Record?
No
Yes
Document Justification
Exclude from Further Assessment
Assess Severity
Document Classification
Identify Possible Error Modes
Assess Likelihood
Assess Probability of Detection
Determine Mitigation Strategy
Document Controls
Repeat for each Electronic Record Type
Repeat for each Failure Mode
Initiate Risk Management

FIGURE 15.2 Electronic Record Risk Assessment Process.

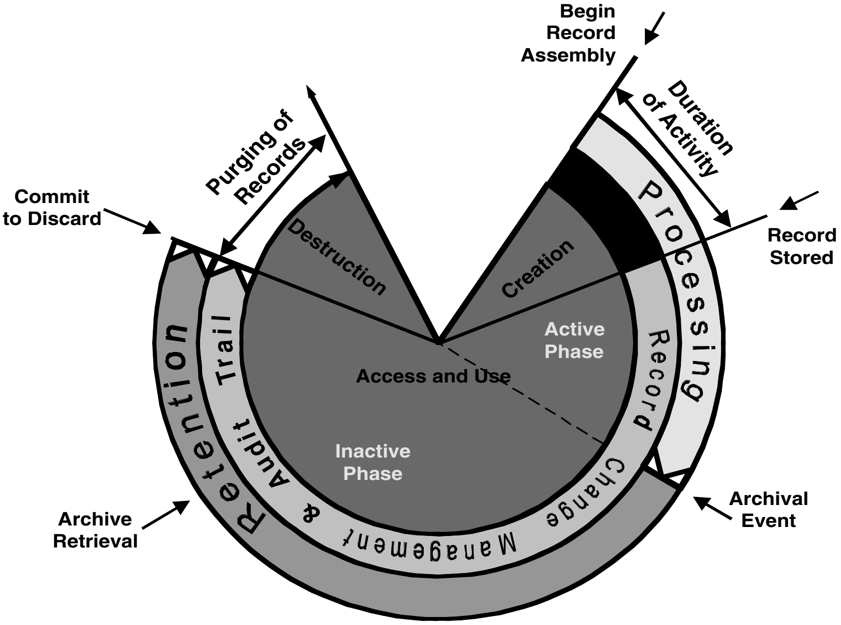


FIGURE 15.3 Electronic Record Life Cycle.

chromatogram, and configuration/control parameters used for equipment setup. Electronic raw data must be protected from alteration, periodically backed up and retained in a secure environment, and not deleted without necessary archiving. Data maintenance requirements are discussed in Chapter 12.

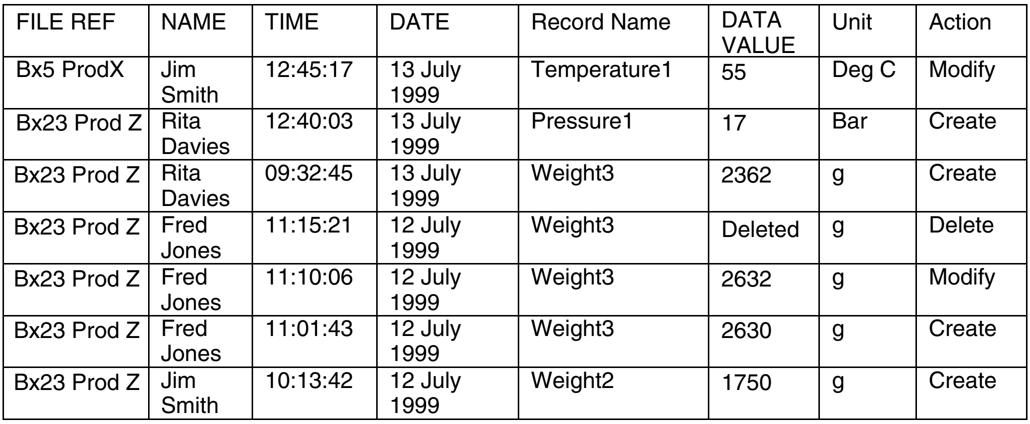
It is important to appreciate that some data may be transient and will never be stored to durable media while other transient data may be processed to derive data before being stored. Systems that only handle transient data are excluded from 21 CFR Part 11. These are systems that acquire and temporarily store data in files that have no user access but, as part of normal workflow, pass that data on to a printer or another system before the process task is complete and the data are purged. Electronic buffers (including temporary files) cannot be considered transient data if user modifications to committed data are permitted. Battery backups for retention of temporary storage invalidates the definition of transient data as do situations where multiple cycles of so-called transient data are stored before being purged.

Audit Trails

Audit trails log who created, modified, or deleted the record, and when ("timestamp"). They should explicitly identify either who or what made the change or allow that information to be unambiguously determined. The FDA has suggested that predicate regulations may be used to determine whether or not audit trails on specific records are warranted.[3] The FDA stresses that it is particularly important to track users who created, modified, or deleted records.

Electronic audit trails are recommended for the most critical electronic records. An example audit trail is shown in Figure 15.4. This example does not imply any preferred format but rather is included here to help demonstrate the principle of construction.

Hybrid audit trails electronically logging "last changed by" with date and link to related paper-based change records are acceptable for critical records so long as previous versions of the record are maintained. It may be possible in some cases to fulfill the audit trail requirements with a transaction database log. Some database designs require the user to execute a "commit record" step, while others commit the data as soon as the next field is tabbed to. In cases where a conscious decision to commit the record is required, data entered should not be defined as an electronic record until this action is taken, thus potentially simplifying the audit trail. In cases where there is no "commit" step, the audit trail should start as soon as each data item is entered.



FILE REF	NAME	TIME	DATE	Record Name	DATA VALUE	Unit	Action
Bx5 ProdX	Jim Smith	12:45:17	13 July 1999	Temperature1	55	Deg C	Modify
Bx23 Prod Z	Rita Davies	12:40:03	13 July 1999	Pressure1	17	Bar	Create
Bx23 Prod Z	Rita Davies	09:32:45	13 July 1999	Weight3	2362	g	Create
Bx23 Prod Z	Fred Jones	11:15:21	12 July 1999	Weight3	Deleted	g	Delete
Bx23 Prod Z	Fred Jones	11:10:06	12 July 1999	Weight3	2632	g	Modify
Bx23 Prod Z	Fred Jones	11:01:43	12 July 1999	Weight3	2630	g	Create
Bx23 Prod Z	Jim Smith	10:13:42	12 July 1999	Weight2	1750	g	Create

FIGURE 15.4 Example Audit Trail. (From ISPE/GAMP (2001), *Good Practice and Compliance for Electronic Records and Signatures: Part 2 — Complying with 21 CFR Part 11 Electronic Records; Electronic Signatures*, published by ISPE and PDA, available from www.ispe.org.)

Entirely paper-based change records alone should be sufficient for noncritical electronic records. Basic data maintenance controls described in Chapter 12 apply.

Audit trails must be available for the duration of a record's retention period and protected from any form of alteration. It should be possible to establish the current value and all previous values of a record by using the audit trail. Normal working practices (procedural and built-in computer controls) should prevent audit trail content being altered without definitive authorization by a second documented supporting party. Audit trails need to be available with their electronic records in human readable form for purpose of inspection.

Timestamps

Timestamps have three basic components: date, clock time, and time zone. The use of dates must be defined to avoid any misinterpretation (e.g., is 02/03/04 understood as February 3, 2004 or March 2, 2004?). System clocks should be set to required levels of accuracy (e.g., hours and minutes). Time zones should be specified except where they can be unambiguously determined.

The application of timestamps should be periodically reviewed. Checks should be made to verify that authorized clock changes such as the change between summertime and wintertime, have been correctly implemented. Checks should also be made for unauthorized modification of system clocks and drift. Networked computer systems can be used to synchronize clocks. Procedural controls should be established to prevent unauthorized system clock changes in the absence of technical means.

Metadata

The FDA has in the past promoted the ability to reprocess electronic records, that is, to retrospectively process necessary raw data again using the same or equivalent conditions to "prove" the integrity of original records. Such processing requires metadata: data about data. Audit trail information is insufficient to reprocess electronic records. Details of the software originally used to create and maintain the records are also required to reprocess records together with hardware platform dependencies.

The FDA has now reconsidered and at present only requires the meaning and content of electronic records to be preserved.[3] This is achieved typically through appropriate validation of supporting computer systems and by applying audit trails where necessary to individual electronic records. Metadata will normally be managed through computer validation rather than as part of the electronic record as required previously by the FDA. This is consistent with other regulatory authorities who only expect constructive evidence to support the accuracy of electronic records.

Copies of Records

During the course of an inspection, it must be possible to provide the inspector with a full and correct copy of the electronic record, both in electronic form and in paper form (human readable form). If it is not possible to evaluate the requested electronic record without the corresponding application, then the inspector or agency should be consulted to determine the action to be taken in each individual situation. Another option for human readable form is saving the data in ASCII format.

Analogous to today's paper-based environment, companies must be able to make requested data available within a reasonable period (typically a few hours for on-line data, and between 24 to 48 h for archived data). This is achieved by displaying the data on screen or by printing it out. As a rule, databases are usually more able to meet the individual requirements of inspectors than is currently the case with paper-based filing systems. However, because the systems used can only be operated in accordance with their specifications, it cannot be assumed that they will be able to answer every conceivable query. For each individual case, it must, therefore, always be clarified with the inspector or agency as to how best the data can best be collected for the purpose of the inspection on the basis of what is technically feasible. This also applies to formats and media used for transmitting data in electronic form.

If it is not possible to evaluate the requested electronic record without the corresponding application, then the inspector or agency should be consulted. It may be necessary to give regulatory authorities access to a pharmaceutical or healthcare company's computer systems to read electronic records. In such circumstances direct access to computer terminals should only be given to trained personnel in accordance with established SOPs — the inspector can witness the company computer systems access.

Record Maintenance

The World Health Organisation GMPs suggest that electronic records should be stored and protected by backup transfer on magnetic tape, microfilm, paper printouts, or by other means.[10] There is no obligation to maintain electronic master copies of electronic records where accurate printed copies exist. The FDA has recently announced a similar position with the proviso that GxP processes do not refer back to the electronic version of the record.[3] If GxP processes refer back to electronic records, then the FDA considers any disposition to paper or other nonelectrical media as incidental and consequently expect the electronic records to be maintained in electronic form. When printing an electronic record that will be retained for GxP purposes, remember to authenticate it either through validation or with a dated handwritten signature applied directly to the print.

Retention periods for electronic records should be the same as equivalent paper records. During the retention period, stored records must be readily available. This applies to records stored on electronic and nonelectronic media. Issues that need to be managed for long-term archiving for electronic records are discussed further in Chapter 13.

Electronic records, like their paper record counterparts, should be purged at the end of their retention period. Procedures for disposal should be defined and should require management authorization for final destruction of records. Some firms keep a log of purged records for a further retention period so that they can demonstrate management and control of the purging process.

E-mail messages, including attachments, should not be used as electronic records unless the e-mail system is validated as fit for this purpose. Validation requirements for e-mail include verifying integrity, authenticity, and confidentiality through appropriate use of protocols, encryption, and public key infrastructure. Individual e-mail messages can be managed as electronic raw data, prints taken with dated signatures annotated, and an electronic master copy maintained.

Software Programs and Configuration

Compiled software including firmware is not considered an electronic record under the scope of regulations like 21 CFR Part 11. Instead, software source code and configuration are considered analogous with Standard Operating Procedures.[11] GERM recommends that a source code listing be retained and the software managed under change control.[8] Where software listings are not available for COTS products, the version number should be recorded and any user-specified operational parameters (setup) documented.

Recent Inspection Findings

- The XXXX computer system … lacked audit trail function of the database, to ensure against possible deletion and loss of records. [FDA Warning Letter, 2001]
- Changes to data that are not recorded and stored on electronic media require an audit trail in accordance with 21 CFR 11.10e. For changes made … the documentation should

indicate who made the change, when it was made, and a description of why the changes were necessary. [FDA Warning Letter, 1999]

- This inspection disclosed deficient controls in the laboratory electronic record keeping system which is used for maintaining chromatographs and audit trails. [FDA Warning Letter, 2000]
- The firm's assessment of the computerized systems such as XXXXX (inventory control system) and XXXXX (LIMS System) found them to be noncompliant with 21 CFR Part 11 requirements. For example, the firm indicated that XXXXX exhibited deficiencies in the area of audit trail. [FDA 483, 2001]
- The electronic record system lacks computer generated time stamped audit trails. [FDA Warning Letter, 2000]
- There is no assurance that the XXXXXX could create an audit trail that was computer generated and time stamped to independently record the date and time of operator entries and actions as required by 21 CFR 11.10(e). [FDA Warning Letter, 1999]
- Review of your XXXX files reveals they have not been properly validated … there is no ability to generate accurate and complete copies of the records in human readable and electronic form, there is no protection of records to enable their accurate and ready retrieval … as well as other significant deficiencies. [FDA Warning Letter, 2001]

ELECTRONIC SIGNATURES

The purpose of an electronic signature in a computer application is to enable an individual to authorize an electronic record (e.g., author, review, approve, comment, etc.). Appendix 15B helps identify examples.

Electronic signatures can be based on nonbiometrics, biometrics, or digital technology. An example of a nonbiometrics signature is the use of the traditional user-ID and password combination. Examples of biometrics signatures are fingerprints, hand geometry, and retinal scans. Digital signatures can be based on cryptographic user keys.

The application of electronic signatures is indicated in predicate regulations where a call is made for a signature, an initial, or an approval/reject (see Appendix 15B). For example, master production and control records are required to have the full handwritten signature of the person preparing the record, an independent checker, and signatures of persons performing and checking laboratory tests. It is important to appreciate, however, that most predicate rules were not written in anticipation of electronic signature requirements and not too surprisingly, they do not comprehensively identify all expected signings. For example, U.S. CFR 211 (cGMP for finished pharmaceutical products) does not specifically identify recall, investigation, or out-of-specification records as requiring signature. Care must be taken not to rely too heavily on predicate rules.

It is recommended that a work flow analysis be conducted to identify checkpoints appropriate for electronic signature. Not all existing handwritten signing or initialing need to be transposed as electronic signatures. In many instances signatures and initials have been implemented to facilitate identification of an individual rather than as any legal signing.[12] Consequently, the availability of audit trail information identifying individuals can remove historical instances of handwritten signatures and initials. A good example of this is the use of initials for nonsignificant activities recorded on batch records. Only significant or critical activities formally require signature. Nonsignificant entries on batch records only require the identification of an individual where relevant. Electronic signatures on electronic batch records are therefore not needed for all signatures and initials found on their equivalent paper records. Caution is in order as the FDA has indicated that all signatures performed electronically, whether or not they are required by predicate rules, must comply with Part 11. Therefore it is advisable to limit electronic signings to those required.

Admissibility

Regulatory authorities such as the FDA, MHRA, MHLW, and TGA expect electronic signatures to be legally binding electronic equivalents to handwritten signatures.[1,4,13] The FDA goes further and requires firms to notify it, in writing, of the use of electronic signatures as an equivalent to handwritten signatures. A standard format letter is provided for this purpose in a docket on the FDA Web site www.fda.gov.

Individuals who apply electronic signatures to electronic records are accountable and responsible for actions initiated under their electronic signatures. Electronic signatures should be declared within the pharmaceutical and healthcare company's organization to be the legally binding equivalent of the person's handwritten signature or initials. Users should be trained to appreciate this equivalence. The consequences of falsifying data or signatures must be made clear.

- Employees should be disciplined for failure to follow company procedures regarding the use and administration of electronic record and electronic signatures.
- Employees should be considered for dismissal if they have deliberately falsified electronic records or electronic signatures.

User acknowledgement that they understand the significance of electronic signings should be documented. This can be done as part of the user request for system access.

Signature Attributes

Electronic signatures must be uniquely assigned to one person and must not be reassigned to another person. Before authorizing the assignment of an electronic signature, the company must identify the individual in question. If a person leaves the company, the signature is not transferable.

The signature application process must, by appropriate technical (computer-controlled) and procedural means, ensure as a minimum that signature creation:

- Can only be applied by the rightful owner
- Cannot, with reasonable assurance, be derived and that the signature is protected against forgery using currently available technology
- Can be reliably protected by the legitimate signatory against the use of others
- Can be linked to the data to which it relates in such a manner that any subsequent change of the data is detectable

In addition, signature creation must not alter the record being signed or prevent such records from being presented to the signatory prior to the signature process. Electronic signatures should be verified at the point of signing to ensure with reasonable certainty that the signature is authentic. Detected discrepancies must be alerted. The signature verification process itself must allow the contents of signed records to be reliably established and any security relevant changes to be detected.

Electronically signed records must contain the following information and this information must be visible each time the record is viewed or printed out:

- Name of the signatory
- Date and time of the signature
- Reason for signature (review or release, for example)

E-mail messages should not be used to authorize GxP activities or approve GxP documentation unless the e-mail system is validated and individual e-mails comply with electronic record requirements.

Linking a Signature to an Electronic Record

Electronic signatures need to be unequivocally linked to their respective electronic records, and in such a way that they cannot be removed as the preamble to 21 CFR Part 11, say by "ordinary means" (e.g., cut and paste). With electronically signed records, the link can be ensured by, for example, a unique relationship within a database or by an additional check using hash algorithms (the hash value of the record is signed).* This unequivocal linking may present something of a technical challenge but has been eloquently achieved in some applications designed to capture and embed handwritten signatures to documents, e.g., PenOp and Entrust.

Identification Codes and Passwords

Administration of electronic signatures based on the combination of user-ID and password must be designed in such a way that the misuse of an electronic signature requires the cooperation of at least two people (e.g., divulging of one's password to a colleague). Only the owner of the signature must know the combination, which typically means only the owner knows their secret password.

User-ID

The unique identifier could be a personal identifier. It does not need to be secret. Old tried and trusted technologies such as a log on entered from the keyboard or more effectively from a card reader or bar code are satisfactory, but these are being superseded by newer ones that are on the way.

Passwords

The secrecy of the password is paramount for the integrity of the nonbiometric signature to be guaranteed. Thus a policy must be in place making this clear and rigidly enforced. It is usual for the deliberate sharing of password to be a dismissable offence. Should such action be necessary, should it be publicized within the organization as a mechanism for ensuring the importance of the policy?

Secret passwords need to be sensibly constructed and maintained. They should be memorized and changed at regular intervals. These requirements are often seen as mutually exclusive! Frequent changes mitigate against remembering the password, whereas never changing or "flip-flopping," i.e., changing between two at the prescribed intervals, risks their accidental exposure.

Guidelines need to be developed to manage this situation and should include:

- A minimum password length of six characters
- Mixed alphanumeric characters
- Avoiding obvious combinations like one's car registration number or dog's name
- Not incrementally changing a character so that it is possible to work out the current password from the key (starting combination) and the date

It was not uncommon in the past for passwords to be legally shared between teams of staff working together. This is acceptable practice as long as users are restricted to read-only access. Shared codes and passwords must not be used where unique identification of an individual is required, such as electronic signatures.

Operating procedures that specify the action to be taken if passwords, ID cards, or the like are lost or compromised in any way must be defined. Staff occasionally forget their passwords or make an attempt at intrusion. The software governing access should react to multiple attempts to gain

* A hash algorithm is a basic technique in asymmetric cryptography; it is an irreversible mathematical function that yields a certain value when used with a data file. For example, used with a document it always yields the same value but it is impossible to calculate the document from the hash value.

access using an invalid password (say three) by locking out the individual and sending an alarm to a responsible person to investigate, take appropriate action, and record the outcome. Some organizations require passwords to be changed every 3 months, but there is no regulatory expectation to force password changes at particular intervals. Indeed it could be argued that changing passwords too frequently will encourage staff to write them down because they will be unable to remember them.

It must be ensured that the unauthorized use of a user-ID/password combination for an electronic signature is detected by the system and that the company's relevant authorities are notified immediately. It must always be ensured that the system design does not permit such misuse — this must be verified as part of the validation.

A suitable escalation procedure should be in place that enables, for example, a typing error when entering a password to be handled differently from an attempt to deliberately falsify a signature. If an authorized user incorrectly types in a password, after three attempts the system blocks the user from using this function and logs the incident. If a user attempts to sign in to an area for which they have no authorization, the system also logs this in a file. The appropriate specified authorities, such as the administrator or system owner, are notified immediately (e.g., by automatic e-mail).

Old identifiers should be removed when staff leave and must not be reissued at least for a number of years (not less than 10), or there will be potential for repeating identifier password combinations and confusing audit trails. These passwords or ID cards must be immediately deactivated.

Hybrid Solutions

Hybrid solutions are systems that use handwritten signatures on printouts of electronic records as the means of approving those electronic records. The handwritten signature must be linked to the associated electronic record, not just to the printed copy. Including the unique file name and the date/time it was printed on the printout can facilitate this. If needed, the paper and electronic copies of the record can be compared later to verify that they have the same content. The meaning of the signature should also be clearly indicated. Labeling of printouts with wording such as "Approved by" may be accomplished as part of the printing process by manual application of a stamp or by writing directly on the paper.

Digitized copies of handwritten signatures (e.g., bitmap images) are not in themselves electronic signatures; they are simply handwritten signatures recorded electronically. Use of uncontrolled bitmaps or other facsimiles of a signature would not comply with the electronic signatures requirement, and may mislead viewers of the document into thinking that a valid signature had been given, when this may not be the case.

The FDA has until recently only considered hybrid solutions as an interim measure until new computer systems can be implemented which fully comply with all 21 CFR Part 11 requirements. This position has now changed[3] and the FDA, in line with other regulatory authorities, will allow the use of a hybrid solution as part of a final system. In either case, robust procedures must be implemented for hybrid solutions to ensure electronic records are contemporaneous with printed copies.

Recent Inspection Findings

- Your written responses dated XXXX and YYYY stated that you would formalize the policy regarding electronic data and signatures and notify the FDA. You have not provided this documentation. This response is inadequate. [FDA Warning Letter, 2000]
- You failed to certify to the FDA that the electronic signatures are legally binding. [FDA Warning Letter, 2001]
- With regards to your responses concerning the use of electronic records and signatures, we find your reply inadequate. 21 CFR 11.100 requires that prior to the time of use,

firms must certify to the Agency that the electronic signatures in their system, used on or after August 20, 1997, are intended to be the legally binding equivalent of traditional handwritten signatures. [FDA Warning Letter, 2001]

- No written procedures that would hold individuals accountable for actions taken under their electronic signatures. It is vital that employees accord their electronic signatures the same legal weight and solemnity as their traditional handwritten signatures. Absent such written and unambiguous policies, employees may be apt to make mistakes, under the erroneous assumption that they will be held to a lower level of accountability than they might otherwise expect when they execute traditional handwritten signatures. [FDA Warning Letter, 2002]
- The firm's assessment of the computerized systems such as XXXXX (inventory control system) and XXXXX (LIMS System) found them to be noncompliant with 21 CFR Part 11 requirements. For example, the firm indicated that XXXXX exhibited deficiencies in the area of "Signature/Record Linking." [FDA 483, 2001]
- The electronic record requires electronic signatures, for which there is no timestamp on the record. [FDA Warning Letter, 2001]
- Electronic documents are not electronically signed and there is no signed hard copy record. [FDA Warning Letter, 2000]

OPERATING CONTROLS

Device Checks

Appropriate measures must be taken to ensure the validity of the sources for data and commands. Validation of the automatic interfaces, or a check of the input medium in the case of manual inputs, is performed as part of system validation. For example, if several sets of scales are connected to a network, only calibrated scales with the correct weighing range may be accessed. Similarly, it should only be possible to use radio scanners assigned to a particular dispensary for weighing raw materials. In addition, personal identification devices (e.g., company identity badges or ID cards that are used in conjunction with a password) should expire after a period and only be issued to authorized users. On expiry such devices should need formal renewal.

The use of devices should be failsafe. However do not assume failsafe operation without thorough checking. A large pharmaceutical manufacturing site in the U.S. once found, for instance, that Visa credit cards could be used to gain access through their site-specific card-swipe system.[14]

The security devices such as strip or bar code readers need to be tested prior to their first use and at regular intervals thereafter. Device checks can be incorporated into routine internal audit procedures. Many of these checks and procedures may already be in place as part of "Good IT Practice" to protect the commercial confidentiality of information. A thorough review of IT security procedures and practices is nevertheless recommended to ensure compliance with electronic record/signature regulatory requirements.

Sequence Checks

The observance of critical sequences must be assured. System function checks should be implemented to verify steps that need to be performed in a particular order. For example, in the process "Input Data→Check Data→Release Data," the system must not permit step 2 to be performed before step 1, and step 3 must not be performed before step 2. Similarly, when a document is first created, the system should automatically check whether another document with the same file name exists. If the file name is already in use on the system, the system needs to force the user to change it. After confirming the acceptability of a file name the document can be stored.

Continuous Sessions System Access

Execution of the first instance of a signature requires full input of the signature (user-ID and password) unless the same user-ID and the same password were entered at login, in which case the password is required. An exception here is start passwords, which must be changed when first used. All subsequent signatures only require input of the password provided that the person who initially logged in continues to use the system without interruption.

There is a potential for unauthorized access when a user terminal is temporarily vacated with the application open, but the risk should not be exaggerated.[14] The default situation must be an automatic lockout of the access device after a defined period of time. Care should be taken to define practical intervals because too short intervals will pose excessive inconvenience. A typical timeout might be say after 10 min of inactive use. The security situation needs to be seen in the context of the total security system from the perimeter fence to the seat in front of a terminal in a manufacturing suite or a dedicated office. Access around sites is often controlled and restricted frequently for all but the most sensitive of tasks; others trained and authorized to carry out the same tasks will be around in the same area. These factors all mitigate against the need to have a very short lockout time. If a terminal were left open inadvertently and another person (authorized or not) entered the secret part of her/his password combination, the application should reject it as being incompatible with the identifier entered earlier. The application must then demand that both parts of the identifier/password combination are reentered and checked.

Open and Closed Systems

Computing environments can be classified as open and closed. A computer system whose access is controlled by authorized individuals is referred to as a closed system. This also applies to systems with modem access if a secured form of dial-in is used. Authorized individuals may be staff from any department within the organization who are responsible for GMP-relevant data, including internal or external personnel who are responsible for system maintenance.

Open systems refer to computer setups in an environment where a specific person who is responsible for the stored data does not control system access. A good example of an open system is the Internet. Specialist controls are required such as encryption and digital signature standards like Public Key Infrastructure (PKI) to provide necessary assurance in electronic records and electronic signatures.

Recent Inspection Findings

- No safeguards to prevent unauthorized use of electronic signatures that are based on identification codes/passwords when an employee who has logged onto a terminal leaves the terminal without logging off. This is serious because another employee or individual could impersonate the individual who has already been logged on and thereby easily falsify a record. The resulting batch production record, for instance, would not be an accurate and reliable indication of the lot's history. Moreover, in such an environment it would be fairly easy for the genuine logged on employee to disavow a signature as false, and thereby seek to avoid responsibility for actions under his/her signature (on the basis that it is fairly easy for someone else to apply his/her electronic signature). [FDA Warning Letter, 1999]
- Failure to establish and implement adequate computer security to assure data integrity in that during this inspection it was observed that an employee was found to have utilized another person's computer access to enter data into the XXXX computerized record system. [21 CFR 211.68(b)] Review 21 CFR Part 11 for regulations pertaining to the

utilization of electronic records and signatures, and security controls pertaining to both. [FDA Warning Letter, 2001]
- No protection of electronic records in Excel application software. [FDA Warning Letter, 1999]

EXPECTED GOOD PRACTICE

Regulatory authorities such as the FDA and MHRA have basic good practice expectations associated with the management and control of electronic records and electronic signatures. For instance, Annex 11 on Computerized Systems of the Guide to the EU GMP Directive 91/356/EEC outlines the following expectations:

- Validation of systems to ensure accuracy, reliability, consistent intended performance, and the ability to detect invalid or altered electronic records.
- Backup of electronic records, their audit trails, and related documentation must be retained for a period at least as long as that required for the subject electronic records and must be available for review and copying by regulatory agencies.
- Determination that personnel (including external suppliers) who develop, maintain, or use electronic record/electronic signature systems have documented education, training, and experience to perform their assigned tasks.
- Security measures employed should be documented and approved.
- The release of batches of finished pharmaceuticals using a computer system for sale or supply regulated by European Union should allow only for a Qualified Person to release the batches, and should clearly identify and record the person releasing the batches.
- Adequate alternative arrangements need to be available in the event of a computer system breakdown to maintain access to electronic records for business continuity purposes. The time to bring the alternative arrangements into use should be related to the possible urgency to use them (e.g., access to electronic records to effect a recall must be available at short notice).

These expectations logically extend to GCP, GDP, and GLP applications. MHRA is currently awaiting confirmation of legal status with use of electronic signatures to GCP and GLP applications.

VALIDATION

GxP regulations require pharmaceutical and healthcare companies to maintain a system of documentation, and this includes any computer systems supporting the management and control of electronic records. Take, for example, EU GMP Article 9.[15] Article 9.1 requires that documents be clear, legible, up-to-date, and retained for the appropriate period, and Article 9.2 goes on to anticipate electronic records, the main requirement here being that supporting computer systems are validated. The FDA expects recordkeeping systems to be validated where required by predicate rule or if they have direct impact on product quality, product safety, or record integrity.[3]

Validation must demonstrate that the computer system is able to store the electronic record for the required time, that the data is made readily available in legible form, and that the electronic record is protected against loss or damage. Both technical and procedural controls should be validated, including audit trail functionality and the successful application of electronic signatures to records.

BACKUPS AND ARCHIVES

EU Directive 91/356/EEC sets out the legal requirements for electronic records within the context of GMP documentation. There is no requirement to maintain electronic copies of records in preference to other media such as microfiche or paper.

Electronic records (including associated electronic signatures and audit trails) must be accessible in a readable form for the duration of the retention period. The retention period depends on the time periods prescribed. An appropriate backup procedure must also be used for operational data.

Examples of archiving include on-line systems and storage on external systems. Appropriate measures must be taken to ensure data availability and integrity. In particular, it must be checked whether a different medium data format is necessary for the archiving period. This may require associated hardware and software to be kept, along with the necessary operating documentation.

Training

Training records should be maintained that demonstrate that individuals, as appropriate, have sufficient education, training, and experience to develop, use, and maintain computer systems that support electronic records and electronic signatures (see also Chapter 4).

Security

See also Chapter 12.

Suitable mechanisms must be put in place to control system access. *ISO 17799 Information Security Management* is a good practice standard and is often quoted by European regulators making observations concerning information security management. It has general commercial applicability and is used outside the pharmaceutical and healthcare industry. It recognizes the existence of regulatory requirements in certain industry sectors. As standard users, organizations can be certified and audited by an independent assessor. While such independent certification is not accepted by regulators in lieu of their own inspections, it does provide clear evidence that an organization is committed to and has achieved basic good practice.

ISO 17799 includes implementation guidance including that for risk management. Relevant topic areas in ISO 17799 include:

- Security policy/organization
- Personnel security
- Physical/environmental security
- Communications and operations
- Access control
- System development and maintenance

The standard attempts to encourage a security culture and shares many of the expectations of 21 CFR Part 11. For instance, to improve personnel security, ISO 17799 recommends definition of security in job responsibilities, personnel screening, training and awareness, and incident reporting. Access controls recommended by ISO 17799 also match Part 11, e.g., user registration, user-ID and password management, definition of user responsibilities, user authentication, and monitoring system access for unauthorized access attempts.

In summary, electronic records must be protected against loss, damage, and unauthorized alteration.

Business Continuity Planning

Plans should be established to protect electronic records throughout their retention period. Such plans should also aim to preserve timely retrieval of electronic records for business and regulatory scrutiny purposes. ISO 17799 prompts:

- Are there procedures in place to ensure correct authorization of information or software when removed from site?

- Are there procedures/processes in place in order to prevent the exposure of information by exposure to Covert Channels or Trojan Code?
- Where software development is outsourced, are there procedures in place to ensure that defined contractual agreements and quality of work are met?
- Are projections of future capacity requirements made to ensure that adequate processing power and storage are available?
- Are agreements (including escrow agreements) established for exchange of information and software between organizations?
- Is there a managed process in place for developing and maintaining business continuity throughout the organization?
- Has a risk assessment been carried out, in order to identify possible interruptions to business processes, i.e., equipment failure, fire, and flood?
- Have plans been developed to maintain or restore business operations in the required timescales following interruption to, or failure of, critical business processes?
- Has a single framework of business continuity plans been maintained to ensure that all plans are consistent, and to identify priorities for testing and maintenance?
- Are business continuity plans tested regularly to ensure that they are up to date and effective?

Recent Inspection Findings

- Master production records are generated from a computer as electronic records without any apparent controls to assure authenticity and integrity. [FDA Warning Letter, 2001]
- In the event that there is an equipment alarm or process utility alarm, the computer system does not retain the alarm information as a permanent electronic record. [FDA 483, 2002]
- There is no documentation to establish that the system by which these [electronic] records were produced has been properly validated. [FDA Warning Letter, 2001]
- The firm did not validate software for electronic records and electronic signatures. [FDA Warning Letter, 2000]
- Your firm failed to validate the electronic documentation system [*and associated electronic records and signatures*] prior to implementation. [FDA Warning Letter, 2000]
- With regard to your responses concerning the use of electronic records and signatures, we find your reply inadequate. 21 CFR 11.10 requires these systems to be validated and to employ procedures and controls designed to ensure authenticity, integrity, and where appropriate, the confidentiality of electronic records. This part also required that adequate controls exist to ensure the distribution of, access to, and use of documentation for system operation and maintenance. Your system must also guarantee that only authorized individuals can access the system. Please be aware of these requirements if you decide in the future to institute the use of electronic signatures/records. [FDA Warning Letter, 2001]
- The firm has not fully implemented procedures for control of all documents for their electronic records and electronic signatures. [FDA Warning Letter, 2000]
- There is no documentation covering XXXX software, or any procedures instituted covering the protection of electronic records or an established backup system. [FDA Warning Letter, 1999]
- Several laboratory instruments (including HPLCs and GCs) were considered noncompliant due to limited security of saved analytical methods. [FDA 483, 2001]
- The firm's assessment of the computerized systems such as XXXXX (inventory control system) and XXXXX (LIMS System) found them to be noncompliant with 21 CFR Part 11 requirements. For example, the firm indicated that XXXXX exhibited deficiencies in the area of security. [FDA 483, 2001]

- Review of your XXXX files reveals they have not been properly validated … access to your system has not been limited … as well as other significant deficiencies. [FDA Warning Letter, 2001]
- Our investigator noted that the laboratory is using an electronic record system for processing and storage of data from the XXXX and HPLC instruments that is not set up to control the security and data integrity in that the system is not password controlled, there is no systematic backup provision, and there is no audit trail of the system capabilities. The system does not appear to be designed and controlled in compliance with the requirements of 21 CFR Part 11, Electronic Records. [FDA Warning Letter, 2002]

IMPLICATIONS FOR NEW SYSTEMS

Electronic record and electronic signature requirements must be specified and taken into account during any selection process for all new computerized systems. Relevant third-party suppliers of bespoke systems should have requirements contractually defined.

Pharmaceutical and healthcare companies should consider working with key individual suppliers and industry groups to help suppliers develop electronic record/signature-compliant COTS products. Current versions of COTS products need not be specifically customized for users to provide full electronic record/signature functionality; the development risk with bespoke development must balance with the complexity and criticality of the change. It should be possible to compensate for the lack of key software functionality by adding user procedural controls.

Open Source software must be fully evaluated by the user organization to assess relevant electronic record/signature functionality since there is no supplier accountable for functionality definition, product development, or maintenance.[12] Caution should be exercised since it is very difficult to truly demonstrate the trustworthiness of such software in the absence of life-cycle development and support documentation.

HAZARD STUDY

The PDA has recommended what is essentially a hazard study process to reveal where record integrity may be compromised.[12] The following checklist has been developed for use with both new and existing systems.

1. Lay out the basic workflow of computer applications and conduct a data analysis to identify electronic record creation and maintenance (include identification of supporting raw data)
2. Answer such questions as
   - Where do the records go?
   - Who uses them, internal and external to the company?
   - How are they used?
3. Identify critical steps along the workflow where the integrity of records may be compromised through use or transmission
   - Incomplete records
   - Duplicate records
   - Communications corruption
   - Transmission gaps/chain of custody issues
   - Opportunities for record corruption
4. Identify levels of control that exist or will be needed for these records
   - Identify how records are secured, backed up, and archived

- Identify how records are restored to active systems from backup
- Examine disaster recovery and security requirements

5. Determine the extent of validation of the computing environment

Application of this checklist can be incorporated into the hazard study process discussed in Chapter 8.

Common Practical Issues

The GAMP Forum identified the following common issues affecting practical compliance in 1999 and they are still very relevant today:[14]

Password Expiry — How to manage when systems do not facilitate automatic periodic change. Also, issue of making sure passwords are not repeated or take forms that are easily guessed (e.g., care registration number, street names, family names).

Retention of Data — Which data is required for retention and can any data be discarded. Maybe practical issues on volumes of data that need to be retained and how this can be managed. It must be practical to search data to find items of interest; otherwise why retain.

Audit Trails — Many systems do not facilitate electronic audit trails. What is an acceptable solution?

User Profiles — In complex systems it is not always practical to have individual user profiles as the management of many thousands of variants is too difficult. The role of all powerful super users needs to be defined and controlled.

Timeouts — Some systems do not facilitate timeouts when a user screen is not actively used. What practical solution is acceptable to regulators? What is an appropriate timeout period?

Virus Management — Virus is a major threat to modern systems; problems with full compliance to Part 11 should not prevent an organization from deploying virus management tools.

Electronic Signatures — When should these be used (for instance, at the point where a record is authorized/approved or is captured in a regulatory document such as a batch record)?

Timestamps in Multiple Time Zone Systems — This seems to have been resolved in that a universal time does not have to be established as long as actions and the order of actions can be established through the process of audit trail as it progresses through different time zones.

E-mail — When can e-mail be used to support validation processes, and should it be avoided (e.g., authorizations and approvals)?

Hybrid Solutions — What constitutes a practical hybrid solution? How do we ensure paper and electronic records are contemporaneous?

IMPLICATIONS FOR EXISTING SYSTEMS

While compliance with electronic record/signature regulatory requirements is not without its challenges for new systems, they are small in comparison with those involved in bringing legacy systems into compliance.

Regulatory Expectations

Regulatory authorities expect electronic record/signature requirements to be addressed although some leniency may be given to older legacy systems. Shared regulatory expectations include:

- Drawing up a timetable indicating how and when compliance with electronic record/signature requirements will be achieved in a company
- Creating an inventory of GMP-relevant computer systems

- Evaluating individual computer systems regarding their compliance, and creating a plan of what is to happen to these systems (e.g., will they be replaced by compliant systems or upgrades?)

However, the FDA and other regulatory authorities expect more than planning to take place. Meaningful progress is expected. Prioritization is accepted as it is widely recognized that it will take some time for all computer systems to come into full compliance. In the transition period, procedural controls are expected to be put in place to compensate for any technical deficiencies.

Management Approach

The GAMP Forum suggests the following key management steps:[9]

1. Agree upon the objective with senior management, gaining their support and approval. This is not a trivial task and may require the approval of significant resources.
2. Compile a list of systems, assign system owners, and identify those that need to be brought into compliance. Communicate the objective, including the support of management, to everyone involved.
3. Meanwhile, an *agreed* interpretation of electronic record/signature requirements for your organization must be developed. This is (politically) the most difficult step and is best done with a small team of informed individuals led by a senior technical manager. Adequate time for debate is necessary to allow all team members to justify the decisions to others when challenged later.
4. Form a team to assess the level of compliance for every legacy GxP system against the agreed interpretation. This is most easily done with a checklist and should be done together with the system owner.
5. Evaluate the strategic options for each system and agree on the actions. There are five basic strategies:
   - Stop the activity *(this is unlikely to apply in many cases)*
   - Retire the system and return to paper *(there are still a few activities which were computerized by an enthusiastic amateur and which add complexity for little or no benefit)*
   - Develop an interim solution *(putting manual procedures in place as an extra layer of control to prop up the computerized system)*
   - Upgrade the computerized system
   - Replace the computerized system *(here migration, record retention, and retrieval become serious issues)*

   This is the most difficult technical step since aquiring sufficient knowledge of the application software to make realistic estimates of the effort involved in updating as against replacement may take some time. The last three options are the most realistic and the latter the most expensive, involving as it does specialist programming in often superseded languages for an application with a limited life.
6. Develop a master plan. It is sensible to include a prioritization step in assessing which systems should be replaced/upgraded first, a decision that should again involve the system owner. Factors affecting prioritization include
   - The GxP criticality of the system
   - The extent of noncompliance (large, medium, small)
   - The age of the system or software and when its operational life is expected to end

Master Plans

The scope Master Plans need not be limited to particular regulatory authorities or regulatory requirements such as 21 CFR Part 11. Many pharmaceutical and healthcare companies have

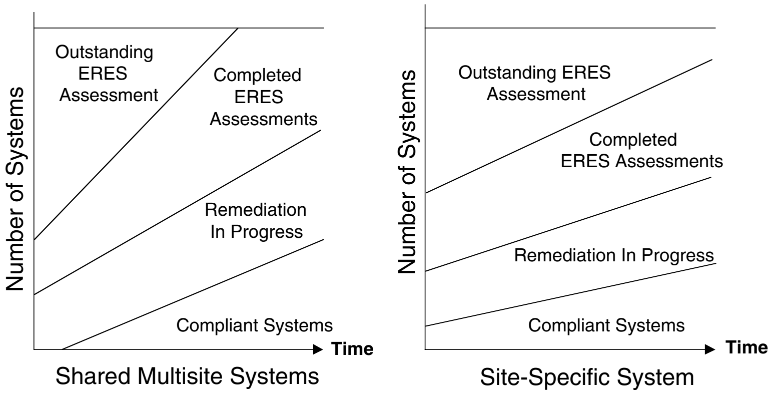


FIGURE 15.5 Example Progress Charts.

developed a more generic organizational plan to collectively address the various electronic record/signature requirements of those regulatory authorities that inspect their operations.

Master Plans should be reviewed and maintained on a regular basis, as business conditions may dictate changes to the actions originally agreed upon. Showing progress against this agreed plan is a vital part of being able to demonstrate progress toward compliance for legacy systems. Example progress charts are presented in Figure 15.5.

Arguing with regulatory authorities that computerized systems cannot be rescued in terms of electronic record/signature compliance or that there was no point in archiving data from a nonvalidated system is not a defensible position. As a bare minimum, interim measures will be expected to have been taken until a "final solution" is implemented. Appendix 15C outlines the use of procedural and technical controls applicable to both pharmaceutical and healthcare companies and their suppliers. The application of interim measures is discussed further in Chapter 14.

Recent Inspection Findings

- We strongly encourage you to perform a thorough and complete evaluation of all your electronic records in accordance with 21 CFR Part 11 as well as guidance generated by the FDA to assure conformance to our requirements. Do not limit your evaluation solely to the examples cited above. [FDA Warning Letter, 2001]
- In addition, we request details regarding steps your firm is taking to bring your electronic cGMP records into conformance with the requirements of 21 CFR Part 11; Electronic Records; Electronic Signatures. … please outline your firm's global corrective action plan, including timeframes for correction, to address this Part 11 issue. [FDA Warning Letter, 2000]
- There was no indication during the inspection that the XXXX system [*and associated electronic records and signatures*] was being validated. In fact there was no evidence that a concurrent manual system was in place. [FDA Warning Letter, 2001]

INSPECTION ANALYSIS

Pharmaceutical and healthcare companies should review their computer systems with regard to common regulatory observations so that mitigating action can be taken or the reasons for sharing such potential observations is understood, and could if necessary be explained during an inspection. An analysis of FDA Warning Letters referring to electronic records and electronic signatures is given in Figure 15.6. This analysis is based on a review of 16 Warning Letters issued by the FDA

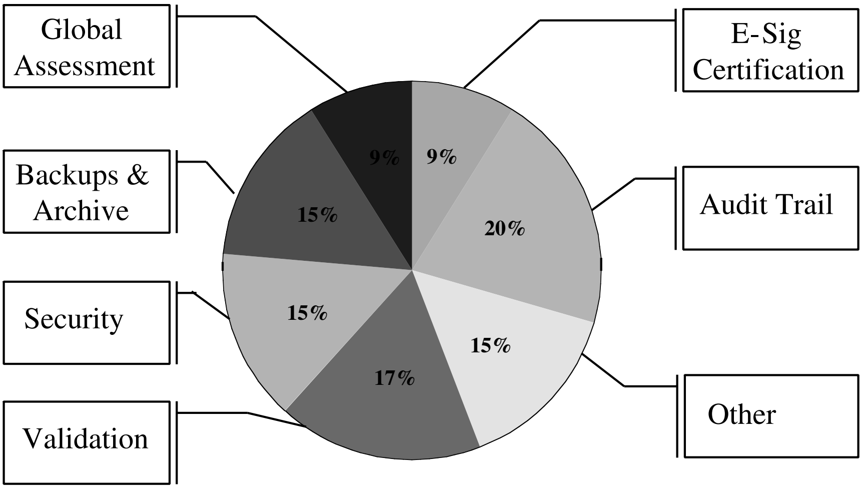


FIGURE 15.6 Part 11 Warning Letters Observation Analysis.

since 21 CFR Part 11 became effective in August 1997. A full list of computer-related Warning Letters reviewed in this book can be found in Chapter 16.

The most common observation made by the FDA concerns the lack of (or incomplete) audit trails. This is often associated with the incorrect identification of electronic records. Specifically, the Warning Letters referred to Chromatography Data Systems (CDS), Electronic Document Management Systems (EDMS), Databases, Batch Records, Change Records, and Device History Records.

The lack of validation or incomplete validation was the next most common observation. The need for prospective validation of electronic record/signature capability during computer system implementation is stressed in two of the six Warning Letters, making an observation on validation. The computer systems concerned were Computer Aided Drawing, Process Control Systems, Record Keeping Systems, and EDMS.

The next most cited group of observations concerned backup and archive. Systematic backups are required to meet defined schedules. Backups and archives must be maintained for the duration of the record retention requirements and for records readily retrievable. The Warning Letters making these observations referred to CDS, Spreadsheets, electronic drawings and to the implied use of Computer Aided Drawing (CAD) application, complaint files, and Device History Records.

Security as a topic is referred to the same number of times as backup and archive. Security issues raised stress the need to limit access to computer systems to protect records, and in one instance deficient password controls are mentioned. Computer systems referred to include CDS, CAD, Record Keeping Systems, and Spreadsheets.

Failure to submit certification to the FDA that the use of electronic signatures in pharmaceutical and healthcare company's organization has the same legal standing as handwritten signatures accounts for just under one in ten Warning Letter observations. This is a simple observation to correct with the issue of a single letter of declaration given to the FDA as described earlier in this chapter.

Three of the Warning Letters referred to a wider organizational review of electronic record/signature requirements beyond the scope of the particular computer systems that were the focus of the original inspection. Pharmaceutical and healthcare companies should ensure that they have a compliance plan that covers the whole part of their organization subject to 21 CFR Part 11.

The remaining Warning Letter observations covered a variety of topics that only appeared once or twice as an observation and did not group naturally with the analysis above. These observations concerned human readable copies of electronic records for electronic drawings and compliant files, taking paper copies of electronic change control records, and continuous session controls in relation to integrity of batch records recording operator actions and detecting invalid records.

REFERENCES

1. FDA (2002), Agency Information Collection Activities; Submission for OMB Review; Comment Request; CGMP Regulations for Finished Pharmaceuticals, Federal Register Notices, 67 (95), May.
2. FDA (1997), *Preamble to Electronic Signatures and Electronic Records*, Code of Federal Regulation Title 21: Part 11, Food and Drug Administration, Rockville, MD.
3. FDA (2003), Electronic Records, Electronic Signatures — Scope and Application, 21 CFR Part 11 Guidance for Industry (www.fda.gov).
4. Pharmaceutical Inspection Co-operation Scheme (2003), *Good Practices for Computerised Systems in Regulated GxP Environments*, Pharmaceutical Inspection Convention, PI 011-1, August.
5. ISPE (2002), Risk-Based Approach to 21 CFR Part 11, White Paper, published by ISPE (www.ispe.org), December.
6. GAMP Forum (2003), Risk Assessment for Use of Automated Systems Supporting Manufacturing Processes Part 2 — Risks to Records, *Pharmaceutical Engineering*.
7. Taylor, J., Turner, J., and Munro, G. (1998), Good Manufacturing Practice and Good Distribution Practice: An Analysis of Regulatory Inspection Findings, *The Pharmaceutical Journal*, The Pharmaceutical Press, The Royal Pharmaceutical Society, London, November.
8. PDA (2002): *Good Practice and Compliance for Electronic Records and Signatures: Part 1 — Good Electronic Record Management (GERM),* published by ISPE and PDA, available from www.ispe.org.
9. ISPE/GAMP (2001), *Good Practice and Compliance for Electronic Records and Signatures: Part 2 — Complying with 21 CFR Part 11 Electronic Records; Electronic Signatures*, published by ISPE and PDA, available from www.ispe.org.
10. World Health Organisation (2000), *WHO Expert Committee on Specifications for Pharmaceutical Preparations*, 32nd WHO Technical Report, Geneva.
11. Compliance Policy Guide (1987), Computerized Drug Processing, 7132a: *Source Code for Process Control Application Programs* (Guide 15), Food and Drug Administration, Rockville, MD.
12. PDA (2003), *Good Practice and Compliance for Electronic Records and Signatures*: Part 3 — Models for System Implementation and Evolution, published by ISPE and PDA, available from www.ispe.org.
13. Directive 1999/93/EC of the European Parliament and of the Council of 13th December 1999 on a Community Framework for Electronic Signatures, Official Journal of the European Communities, January 19, 2000.
14. Selby, D. (2000), Practical Implications of Electronic Signatures and Records, in *Validating Corporate Computer Systems: Good IT Practice for Pharmaceutical Manufacturers* (Ed. G. Wingate), Interpharm Press, Buffalo Grove, IL.
15. European Union Guide to Directive 91/356/EEC (1991), *European Commission Directive Laying Down the Principles of Good Manufacturing Practice for Medical Products for Human Use.*
16. GAMP Forum (1999), *Complying with 21 CFR Part 11 Electronic Records and Electronic Signatures*, First Draft: Consultative Document to Solicit Feedback, December.

APPENDIX 15A
EXAMPLE ELECTRONIC RECORDS

Electronic records can be identified by searching regulatory requirements for the key words "record" and "document." This appendix is based on the U.S. Code of Federal Regulations and EU Directives, and is not intended to be exhaustive. More definitive listings are expected to be published by industry groups such as ISPE/GAMP.

SUMMARY OF REFERENCES IN GCP

- Consent documents (informed and Institutional Review Board)
- GCP protocols and amendments
- Clinical investigation and changes
- Financial disclosure forms and reports
- Investigator statement
- New drug application forms and submission statements
- Clinical study data and ownership statements
- Investigational drug shipment and disposition

SUMMARY OF REFERENCES IN GLP

- Equipment maintenance and calibration records
- GLP protocols and amendments
- QA audit records
- Standard Operating Procedures
- Final Study Reports and QA Statements
- Training records
- Job descriptions

SUMMARY OF REFERENCES IN GMP

- Equipment cleaning maintenance records
- Master production and control records
  - Components specifications
  - Drug product containers and closures specifications
  - In-process materials
  - Packaging material
  - Labeling specifications
  - Drug products specifications
  - Procedures and specifications
- Batch production and control records, including
  - Products from contractors
  - Production records
  - Packaging records
  - Laboratory tests results (QC Records)
  - Reprocessing of batches
- Biological sterilization
- Laboratory tests
- Out of specification investigations
- Customer complaints
- Standard Operating Procedures

- Training records
- Job descriptions

Summary of References in GDP

- Distribution and shipment records
- Adverse event reports
- Recall records
- Customer complaint records
- Standard Operating Procedures

APPENDIX 15B
EXAMPLE ELECTRONIC SIGNATURES

The regulated use of signatures can be determined by searching regulatory requirements for the key words “signature,” “initial,” “approval/approved,” “authorization/authorized,” and “certify.” This appendix is based on the U.S. Code of Federal Regulations and EU Directives, and is not intended to be exhaustive. More definitive listings are expected to be published by industry groups such as ISPE/GAMP.

Summary of References in GCP

- Consent documents (informed and Institutional Review Board)
- GCP protocols and amendments
- Clinical investigation and changes
- Financial disclosure forms and reports
- Investigator statement
- New drug application forms and submission statements
- Clinical study data ownership statements

Summary of References in GLP

- GLP protocols and amendments
- Exact transcripts of raw data and changes to raw data
- QA audit records
- Authorization for animal treatments
- Changes to, and deviations from, standard operating procedures
- Final Study Reports and QA Statements

Summary of References in GMP

- Major/critical equipment cleaning, maintenance, and use
- Master production control and batch records
  - Components
  - Drug product containers
  - Closures
  - In-process materials
  - Packaging material
  - Labeling
  - Drug products
  - Procedures and specifications
  - Products from contractors
  - Final batch production record
- Laboratory tests
- Out of specification investigations
- Significant steps in production (e.g., dispensary and weighing)
- In-process controls
- Formal checks, where appropriate
- Deviations and unusual event records
- Rejection of batches
- Reprocessing of batches
- Recovery of batches
- Standard Operating Procedures

Summary of References in GDP

- Distribution and shipment records
- Adverse event reports
- Return records
- Recall records
- Customer complaint records
- Standard Operating Procedures

16 Regulatory Inspections

CONTENTS

Regulatory inspections are conducted before a new drug or device can be approved, to verify production method and technology changes, and periodically verify every 2 or 3 years that GxP practices are being maintained. Inspections are used to determine if processes are adequately validated with documentary evidence that provides a high degree of assurance that a specific process will consistently produce a product meeting its predetermined specifications and quality characteristics.[1]

This chapter discusses what to expect during inspections, how inspectors approach their work, and how to manage the process of receiving an inspection. Specifically, inspections by the U.K. Medicines and Healthcare products Regulatory Agency (MHRA) and the U.S. Food and Drug Administration (FDA) are explored. Preinspection questionnaires and inspection checklists used by the regulatory authorities are attached as appendices to this chapter.

INSPECTION AUTHORITY

The inspection authority of the FDA, MCA, and other regulatory authorities is broadly the same although specifics vary. Taking the FDA as an example, it has legal authority to gain access to all regulated companies' facilities including vehicles that carry regulated products. This remit covers the use of equipment, computer systems, and personnel with production, warehouses, packaging, and distribution facilities.

The FDA has the authority to inspect records, files, papers, processes, controls, and facilities bearing on whether prescription drugs are adulterated, misbranded, or in some other way violate GxP regulations. No distinction is made between active pharmaceutical ingredients (APIs) and finished pharmaceuticals, and failure of either to comply with cGMP constitutes a failure to comply with the requirements of the Federal Food, Drug, and Cosmetics Act. It is policy not to examine internal audit and supplier audit reports without due cause because the FDA does not want the company to compromise the detail in these reports on the premise that it might be inspected. The FDA, however, is not allowed access to financial data and information, sales data (other than shipping and distribution), pricing information, personnel records (except training records and CVs), and research data (other than for product being inspected). While this distinction in theory is quite clear, it is sometimes difficult in practice to split items of GxP and non-GxP information that may exist together in a single record.

INSPECTION PRACTICE

The FDA is sometimes quoted as saying, *"In God we trust, everyone else needs documentation."* This phrase neatly captures a strong and common theme to GxP inspections conducted by the various national regulatory authorities around the world. Computer validation requires the documentary evidence that a system was developed, and is operated and maintained in accordance with predefined acceptance criteria, i.e., demonstrably fit for purpose. The FDA is primarily looking for evidence of bad practice and fraud. This stringent approach was reinforced by the "Generic Drug Scandal" in the late 1980s when the FDA uncovered instances of fraud by pharmaceutical companies. Other regulatory authorities such as the MHRA have much more of a "partnership" approach. Each approach has its merits.

Approach to Organizational Capability

The emphasis of inspections is moving away from particular products toward general operational capability. This move was first evident in the Quality Systems Inspection Technique (QSIT) adopted by the FDA for medical device inspections in January 2000. Companies are considered "out of control" if any one of the main quality management controls inspected is found noncompliant with regulatory requirements:[2]

- Complaint handling
- Corrective and preventative action
- Management oversight
- Production and in-process controls (including design)

The success of the inspection technique led to the development of the Systems Based Approach for full and abbreviated inspections of pharmaceutical and healthcare companies. Full Inspections are conducted for the initial inspection of a facility, or where a facility has a history of poor compliance, or where significant changes have taken place, or for any other cause deemed appropriate. Abbreviated Inspections are applicable when a pharmaceutical or healthcare company has a record of GMP compliance, with no significant recall or product defect or alert incidents, or with little change in scope or processes comprising the manufacturing operations of the firm within the last two years. Both full and abbreviated inspection will satisfy biennial inspection requirements.

Full inspections will cover all, and abbreviated inspections at least two, of the following:

- *Quality System* (including status of required computer validation/revalidation, change control, and training/qualification of QA staff)
- *Facilities and Equipment Systems* (including equipment IQ/OQ, computer qualification/validation, security, calibration and maintenance, and change control)
- *Materials System* (including qualification/validation and security of computerized or automated processes, change control, and training/qualification of personnel)
- *Production System* (including contemporaneous and complete batch production documentation, validation and security of computerized or automated processes, change control, and training/qualification of personnel)
- *Packaging and Labeling System* (including validation and security of computerized processes, change control, and training/qualification of personnel)
- *Laboratory Control System* (including calibration and maintenance programs, quality and retention of raw data, validation and security of computerized or automated processes, system suitability checks, change control, and training/qualification of personnel)

These focal points should be rotated in successive Abbreviated Inspections. The frequency of Abbreviated Inspections will be based on the pharmaceutical or healthcare company's specific operation, history of previous coverage, and other priorities determined by the FDA. The manufacturing operations of some firms may be limited, and an Abbreviated Inspection may itself comprise inspection of the entire firm (e.g., contract laboratory, in which case Abbreviated Inspections are synonymous with Full Inspections).

The FDA District Office managing an inspection is responsible for determining the depth of coverage given to each pharmaceutical or healthcare company and whether a computer validation inspection expert is required to assess the state of compliance.

In order for a pharmaceutical or healthcare company to be considered in a state of control, there should be no "objectionable" deviations identified in any one focal point covered during an inspection. Whether or not a Warning Letter is issued will depend on the seriousness and frequency of the problems found. It should be possible to determine from a FDA 483 whether or not a Warning Letter is likely based on the following guidance:

- Quality System
  - Pattern or failure of QA personnel to review/approve procedures/documentation
  - Pattern of failure of QA personnel to assure compliance with SOPs
- Facilities and Equipment
  - Pattern of failure to qualify equipment including computers
  - Pattern of failure to establish/follow change control process
- Materials System
  - Lack of validation of computerized processes
  - Pattern of failure to establish/follow change control process
- Production System
  - Lack of validation of computerized processes
  - Pattern of failure to establish/follow change control process
- Packaging and Labeling
  - Lack of validation of computerized processes
  - Pattern of failure to establish/follow change control process
- Laboratory Control System
  - Lack of validation of computerized and/or automated processes
  - Pattern of failure to establish/follow change control process
  - Pattern of failure to retain raw data

Full Inspections may be recommended as a consequence of an adverse Abbreviated Inspection. The issuance of a Warning Letter or undertaking of other significant regulatory action will normally warrant a Full Inspection to verify remedial actions as satisfactorily completed and thereby close out immediate FDA concerns. Failure to satisfy regulatory authorities such as the FDA can result in heavy fines (see Chapter 1) and restrictions on future product approvals and marketing licenses.

An important aspect of this new approach is the expectation that pharmaceutical and healthcare companies will implement any corrective actions identified as the result of a site inspection across the whole of their operations. Effective coordination of corrective actions is vital for large multinational organizations. An example form that might be used to collate computer validation inspection history is presented in Table 16.1. The FDA and MHRA already have access to inspection databases and have the ability to readily trend data and track repeated offences on particular topics across multiple sites in a firm's organization. Indeed, regulatory authorities may in the future share inspection findings with the MRA partner regulatory authorities.

Approach to Individual Computer Systems

Most regulators follow a top-down approach similar to the four-level review process described by the FDA:[3]

Level 1: Recognize how the computer system interacts with operations.
Level 2: Evaluate the quality procedures used by companies to control their operations.
Level 3: Examine documentation in the validation package supporting and computer system.
Level 4: Review software source code as appropriate.

The first review level is necessary to confirm the inspector's understanding of the criticality of computer systems and set the inspection priorities. This will involve discussions with the pharmaceutical and healthcare company's senior technical management and a tour of the facility.

TABLE 16.1
Example Inspection History Form

		Inspection Type					Inspector's Classification of Findings				Computer System Affected							Current Status
Inspection Date(s) (dd/mm/yy)	Regulatory Authority/ Inspector(s)	GxP	PAI	For Cause	Inspection Scope and Site	Total Number of Inspection Findings	Critical	Major	Minor	Brief Description of Significant Commitments	Analytical Laboratory Instrumentation	Process Control or Monitoring System	Spreadsheet or Database Application	Corporate Computer System	Infrastructure or Service	Medical Device	Electronic Records/Signatures	N = Not Started P = Planned I = In Progress C = Completed
12-1-98	MCA — A. Person	✓			Sterile Manufacturing, Brighton	4	0	1	3	Validate BMS		✓						C
5-7-98 to 12-7-98	FDA — B. Person, C. Person		✓		New Drug Product X, Bordeau	7	0	1	6	Configuration change control	✓							C
10-7-98	MCA — D. Person	✓			General Manufacturing, Manchester	3	0	0	3	None	✓	✓						C
25-11-98 to 3-12-98	FDA — E. Person, F. Person			✓	API Manufacturing, Trenton	21	5	2	14	System backups, validate MRP II/LIMS				✓	✓		✓	I
8-1-99 to 11-1-99	TGA — G. Person	✓			Sterile Manufacturing, Darwin	6	0	1	5	Validate spreadsheets		✓	✓					I
23-10-99 to 24-10-99	MCA — D. Person	✓			Y2K, Manchester	0	0	0	0	None	✓	✓	✓	✓	✓	✓		N
17-8-00	MCA — D. Person	✓			Distribution, Manchester	3	0	1	2	Validate warehouse distribution				✓				P
30-10-00 to 1-12-00	FDA — E. Person, H. Person	✓			API Manufacturing, Trenton	13	0	1	12	Training records against user profiles	✓			✓	✓		✓	P

The second review level should identify poorly defined or missing procedures within the pharmaceutical and healthcare company's quality system. This will affect the expectations of the third review level and the scope and detail of validation documentation.

The third review level examines the document sets for particular computer systems identified in the first review level. Validation Plans and Validation Reports are typically among the first documents to be inspected. If the review of a computer system is not superficial, the main life-cycle documents identified in Chapter 4 may be inspected. The inspector is likely to ask to see evidence of system specification and qualification, supplier evaluation, data maintenance, change control, training, and security. Sometimes inspectors will ask for supplementary information to be sent onward to them if they are seeking clarification of an issue.

The fourth review level is usually only invoked by specially trained inspectors for software configurations and customizations, but may be extended to standard software packages where deficiencies are identified.

Throughout the review process where customary or reasonable validation evidence is lacking or incomplete, inspection scrutiny may be increased. Conversely, if the preliminary review of the validation evidence does not raise apparent or suspect problems, the scrutiny may be reduced. Once identified, inspectors will pursue weak spots such as lack of documentation or inconsistencies. They will examine employee performance for common errors (training or ways of working at fault). The inspector will establish the degree of any compliance gap between company practice, company procedures, and regulatory requirements. It is worth presenting information to inspectors in a form that is readily understandable and meets their expectations. Use industry terminology wherever possible.

Mutual Recognition Agreements

The concept behind the MRA is that one regulatory authority will accept the findings of another authority with confidence in the rigor of the inspection process and hence negate the reason to conduct its own inspection of the same pharmaceutical or healthcare company. This is all good theory but requires harmonized inspection standards, practices, reporting, and training.

Regulatory inspections conducted under the MRA have already begun although progress on individual agreements is often a start/stop affair as various issues are worked through. Initial pilots are almost always based on inviting an inspector from one authority to participate as an observer in an inspection by the other authority. Budget constraints are being imposed by most national governments on their respective regulatory authorities and it is not likely to be long before MRA inspections become a regular occurrence. In the interim, it is reasonable to expect inspection findings to be shared between different regulatory authorities. FDA inspection findings are available to the MHRA anyway under the U.S. Freedom of Information Act. A reciprocal arrangement, other than the MRA, does not exist to give the FDA open access to MHRA inspection findings.

INSPECTION PROCESS

Receiving an Inspection Request

When a request to conduct an inspection is received, the pharmaceutical or healthcare company's senior management should be immediately notified. Notice of an inspection may be received by a number of people in a pharmaceutical or healthcare company, so it is important that a procedure exists describing how and to whom the request is passed onto. Usually the focal point is the Head of Quality.

After receiving an inspection request, the Head of Quality will appoint an Inspection Response Team Manager. The Inspection Response Team Manager should contact the regulatory authority concerned to confirm the date, time, duration, site, and topic of the inspection. It is not unknown for inspectors to arrive at the wrong site or to try to inspect systems or product that are not located at the site proposed for inspection. The inspector may request advance information and documentation. The response to these requests must be carefully considered as information may be interpreted out of context by the inspector.

At this stage the pharmaceutical or healthcare company may wish to consider asking the inspector to sign a confidentiality agreement. During the inspection proprietary information must be respected.

Preparing for an Inspection

An SOP should be prepared to describe how inspections are to be managed from the notification of an inspection through its completion. Such procedures are usually applicable to multiple sites within a pharmaceutical and healthcare company's organization, ensuring inspectors are treated in the same fashion no matter which site they inspect. Advice on how to handle inspection scenarios (good and bad) and particular inspectors should be captured in training materials rather than the SOP.

The structure and membership of the Inspection Response Team should be agreed upon in accordance with predefined internal guidelines. Inspection Response Teams are usually established at a site level. The Inspection Response Team Manager should not have to negotiate release of key personnel. Table 16.2 suggests Inspection Response Team roles and responsibilities. One individual may fulfill more than one role, but careful consideration should be given to whether certain mixes of roles actually conflict. Named deputies should be recorded in case primary nominations are not available for whatever reason.

Key preparation steps for an inspection include the following:

1. Prepare personnel to receive audit, possibly including training in how to interface with inspectors for those who are unfamiliar with inspection requirements. Notify site of inspection so that general preparations can be put in place. A site briefing may be appropriate.
2. Obtain room/office for the inspector that is isolated from employees: the Inspection Room. In parallel allocate a room or office as the Inspection Response Team's Control Room. The Inspection Room should not be too close to the Control Room.
3. Identify what information and resources may be needed during the inspection: what was reviewed and outcome of previous inspections, and what corrective actions are closed, in progress, and not started. Review problem logs and change control records. Consider if there are any topics the company would like to take the opportunity to brief the inspector with.
4. Gather documentation for key computer systems together in the Control Room. Arrange files into a logical accessible order. Typical documentation to get ready should include
   - Organizational charts
   - Training records
   - Validation Master Plans
   - Change control records
   - Problem logs
   - System requirements and overviews
   - Development methodology
   - Validation Plans and Reports
   - Testing records
5. Perform a quick walk-through of key computer systems and user workstations in the facility at their point of use. Consider conducting a mock inspection. Pull the records from archives (can information be retrieved in a timely manner?). Review documentation for obvious errors — a fresh pair of eyes! Identify potential problem areas and have answers prepared. Final computer validation reports should be available in English for the FDA.

TABLE 16.2
Inspection Response Team Roles and Responsibilities

Roles	Responsibilities
Team Manager	• Manages Inspection Response Team • Acts as company's direct interface with inspector when organizing logistics for inspection
Inspection Coordinator	• Manages Control Room • Coordinates Scribe and Runner
Host and Deputy	• A senior manager • Represent site management • Welcome inspector and establish commitment of company to support inspection and its outcome
Quality Assurance Representative	• Own inspection process • Agree company position on inspection topics • Agree response to inspection findings • Provide knowledge of how computer systems are used in support of GxP
Quality Control Representative	• Provide knowledge of how computer systems are used in quality control processes
Regulatory Affairs Representative	• Provide knowledge of regulatory submissions with direct and indirect reference to use of computer systems
Technical Representative	• Provide technical backup on deployment and maintenance of computer systems (IT, process control, and laboratory applications)
Operations Representative	• Provide knowledge of how computer systems are used
Validation Group Representative	• Support inspection of validation documentation from retrieval of appropriate documents to walking through validation conducted
Scribe/Secretary	• Keeps minutes of inspector's comments and observations • Keeps a record of documents requested and provides to inspector
Escort	• Accompanies the inspector during the inspection at all times
Runner	• Brings and removes documents requested by the inspector

Note: Typically the Host for an inspection is the site QA Manager. It is usually polite for the Site Director to attend opening and closing meetings.

The preparation for inspections should include a risk assessment based on the drug product being processed, the production process involved, and the technology mix including the use of computer systems and a review of the company's internal audit and regulatory inspection history.

Hospitality

Hospitality must not be perceived as influencing the inspection. Regulators are typically required to pay their own accommodation costs and usually have a fixed daily allowance. Suggest suitable local hotels that fit their pocket. Hotel reservations can be made on their behalf but check they are comfortable with the arrangements. The pharmaceutical and healthcare companies should also consider local transport requirements from the airport or train station to the site, and daily commuting to and from the hotel. If the inspector is making his/her own way to the site under inspection, then reserved car parking would be courteous.

Only company representatives hosting the inspector should stay at the hotel to avoid accidental discussions being overheard — it is not unknown for inspectors to overhear conversations in the hotel bar! Company administration staff should check that no company employees or suppliers are booked into the hotel for the duration of the visit. Make sure there are not too many company representatives acting as host at any one time as it gives the opportunity for the investigator to play one representative off another. It also makes for a more congenial atmosphere.

The pharmaceutical and healthcare company should consider establishing a policy whereby personnel are to decline to comment on inspectors' queries outside company premises. Indeed personnel should be required to notify site security who will mobilize an official company response to off-site queries. Only nonwork issues should be discussed out of work; otherwise, personnel should say that the run of the conversation is inappropriate for discussing or chatting and should, if necessary, walk away.

Arrival of the Inspector(s)

Site security should be briefed on the expectation of an inspection. First impressions count, so security should be courteous, and the site needs to be generally tidy and in a state of good repair.

Upon arrival the inspector should present himself/herself to the site reception or gatehouse. The nominated Host will usually go to meet the inspector and take him/her to the designated Inspection Room. Once on site, an Escort and a Scribe should accompany the inspector at all times. The Scribe will record all remarks, observations, questions, and responses made by both the inspector and company staff. If other authorities arrive with the inspector, note who they are and why they are there. This information should be relayed to the Control Room.

When at the designated Inspection Room, try to agree to use it as a base for the inspector. Confirm the purpose and scope of the inspection. How long will the inspection last? Is this a routine or "for cause" inspection? What documentation would they like to see? Who would they like to speak with during the inspection? Do they have any other requirements? Create an agenda for the inspection with the inspector. An inspector will not always have a predefined agenda, and an agreed plan will help the inspector structure the inspection as well as help the host organize logistics to make the inspection as efficient as possible. Request daily wrap-up meetings during the inspection and final closure meeting.

Conducting the Inspection

Company personnel need to perform well during the inspection. Presenters and supporters need to be alert and ready throughout. The inspection is not over until the regulatory inspector is traveling back home.

Do not assume anything; always repeat inspector questions and ask for clarification if required. Inspectors may ask open-ended questions or make nonspecific requests. This may be because they themselves are unsure of what exactly they want and are just fishing around. A sense of balance should pervade. Do not question every request in detail as this will almost always annoy the inspector. Only address the specific point being raised by an inspector when answering questions — do not elaborate. Do not explain your answer unless specifically requested to do so. Let the inspector follow through his or her process. It might seem like helping but it might end up confusing the situation. Beware of informal "off the record" questions because everything is on the record. Do not get "friendly" with the inspector. Further, do not be tempted to speak when the inspector is quiet. Silence is generally good, not bad. Inspectors may employ long gaps between questions to encourage loose talk. Do not argue with inspection observations. Instead, prepare evidence to present to the inspector to address his or her concerns.

Inspectors will typically assume everything is GxP-critical unless justified with rationale, and even then they are likely to spot check and challenge such justifications. Types of inspection questions related to computer systems include the following (based on Reference 4):

- Quality management system and system development methodology
- Use of tools and standards
- Use of supplier (roles and responsibilities)

- Document control (draft, review, approve, superseded, withdraw)
- Change control
- Access controls (passwords and user log-ons)
- Data sources and entry/capture including contemporaneous transcription
- Data processing
- Data archiving, storage, and retrieval
- Information security management (including virus checking)
- Internet links
- Remote access
- Electronic records and audit trails
- Signatures and status control
- IT Infrastructure (including network firewalls)
- E-mail transactions/interactions
- Configuration and version control
- User training

There will be uninitiated questions, inquisitive questions, skeptical questions, adversarial questions, and long, pregnant pauses from the inspector. Personnel should be instructed to state only what they know to be true, and not to guess or speculate. Personnel should be firm and sincere when answering questions. This does not mean they must not become adversarial. If they do not know the answers, they should let the inspector know this and that they will get back to him or her to follow up on their request. It is perfectly acceptable to admit you do not know, but make sure the question is not left unanswered. Open issues should be noted by the Scribe and logged by the Inspection Response Team. Follow-up responses should be discussed with the Inspection Response Team and positioned accordingly before the inspector is given the answer or information.

Above all, there should be a consistent approach by personnel to the inspector. There should be an objective of thoroughness and clarity — of trying to do the right thing and not shirking responsibility. Be sensitive to the responsibilities and demeanor of the inspector — he or she may be just having a bad day! Make best out of deficiencies, concentrating on positive aspects; what has been done to put situation right and what is planned. Avoid the use of jargon: do not use undefined terms during the inspection. It is also important that personnel are briefed and made sensitive to possible national language differences, e.g., "warm feeling," which means *in control* in the U.K., means *out of control* in the U.S.

Inspectors may ask for documentation that is outside their inspection authority. Do not provide such documents without due consideration. They will have some reason for the request, and if you are unsure about the validity of the request, gently explore this with them. Be careful not to refuse documentation by citing strict interpretation of the regulations; be cooperative where possible. Consider if the inspector's line of enquiry could be pursued without documentation — is alternative proof available? For instance, share audit schedules rather than audit reports as proof of auditing.

Make a list and copies of all documents provided to the inspector during the inspection. Mark documentation given to the inspector as appropriate (confidential, restricted, uncontrolled, controlled, etc.).

Only provide documentation specifically requested. Provide copies of requested documents as per company SOP in a timely manner. Lengthy lag times in responding will make the inspector suspicious that there is a problem. Some questions are appropriate to answer quickly such as SOPs; some require slower response such as technical detail. Inspections of computer systems are predicated on the assumption that pharmaceutical and healthcare companies have effective record retention and retrieval systems.[5] Significant problems may arise during inspections where these systems are inefficient or ineffective.[6]

Pharmaceutical and healthcare companies should have a company policy that no cameras, videos, or recording devices can be used without prior written permission. This policy should apply to inspectors too. If pushed by an inspector the company should take and process photographs and send a copy on to the regulatory authority concerned.

Do not employ any delay tactics. On the contrary facilitate a swift inspection and let the inspector go home — that is what both parties are really after. The company should not want a repeat visit.

The majority of inspectors working for regulator agencies do not have specialist knowledge of computer systems and technology. Should the assigned regulatory inspectors responsible for an inspection be particularly anxious about the validation of computer systems, then advice and assistance can be requested from a specialist inspector within the agency. Remember that if the discussion of an issue is getting bogged down in technical detail, it might be useful to position a commonsense type of explanation. This approach is after all what many inspectors will use to determine if there is a potential problem in the first place.

Demonstrations may prove useful to an inspector by facilitating a less time-consuming overview of functionality. Obviously the demonstration should reflect how the system is used in real life. The availability and suitability of demonstrations (including simulations) should be carefully planned. Demonstration software needs to be validated in its own right.

Keep control of the inspection by leading the inspector as much as possible through the agreed upon agenda and processes being audited. Remain calm and cordial at all times. Do not let company staff argue amongst themselves in front of the inspector; make sure the staff put in front of an inspector do not have an axe to grind. Do not make hasty commitments; some inspectors make lots of suggestions, and this might just be an indication that they do not understand fully how the company manages issues.

Sometimes inspectors will ask for supplementary information to be sent to them once their site visit is finished. Such documentation must be controlled in the same fashion as documents given to the inspector during the inspection. Remember to agree on timings of delivery of any documentation. Timings should not be agreed to that cannot be achieved. The inspector will generally be understanding of reasonable time constraints.

Daily Washup with Inspector

While inspectors are under no obligation to conduct daily washup meetings, they can be very useful to the inspectors themselves and the inspected. Such meetings can provide a useful means of getting/giving early feedback on the good and the not-so-good from the inspectors' perspective. In particular, washups offer pharmaceutical and healthcare companies two main benefits:

- The opportunity to provide requested information to the inspector that could not be supplied earlier and thereby possibly close what might otherwise be issues left open.
- The opportunity to clarify outstanding questions/issues that are not satisfactorily closed so that closure can be planned.

The attendance at daily washup meetings should be limited to the Host, senior members of the Inspection Response Team, and a Scribe. A separate site washup can be held afterward with the full Inspection Response Team and other invitees as appropriate. The daily washup should be used as the beginning of preparations for the next day's inspection.

Do not volunteer "war stories" about fixing the system. You may think this will impress the inspector but it will not because the inspector will be worried that the project is out of control. A good project is one that is well managed so that there are not situations warranting heroic action!

After the Inspection

The Inspection Response Team will normally conduct an internal debriefing immediately after the inspector has left the site. A more formal Inspection Report should be written soon afterward. The

Inspection Report will summarize the inspection and include an index of all documentation provided to the inspector. In addition the Inspection Report will capture the corrective actions that the pharmaceutical or healthcare companies will share with the regulatory authority to close any adverse observations made by the inspector. There may also be other lessons that will be acted upon that will not be openly shared with the regulatory authority.

It is important that the inspection findings be presented to senior management in an honest, direct, and timely fashion. It may be many weeks, even months, before the inspector officially presents inspection findings back to the pharmaceutical or healthcare company. This is too long to wait to keep senior management informed of the implications of the inspection.

Inspection Findings

The inspector will normally write back to the pharmaceutical or healthcare company after the inspection to confirm significant findings (positive and negative). The letter can take many weeks to arrive. Observations concerning the validation of computer systems might be logged as specific items or incorporated within the text of the system's associated equipment/process.

A citation of noncompliance, known as a "483," may be drafted by the FDA at the close of the on-site inspection with a pharmaceutical or healthcare company. An opportunity to clarify issues is given before the close of the inspection and the formal issue of the citation. Similar reports are written by the EU agencies and the Australian TGA, but unlike the FDA citations that are available to the public in accordance with the U.S. Freedom of Information Act, these reports are confidential to the inspected company and the regulatory authority.

The FDA will consider the lack of computer validation as a significant inspection finding and log it as a 483 noncompliance citation. The MHRA may take a more lenient view depending on the criticality of the system on GxP operations. The lack of a detailed written description of an individual computer system (kept up to date with controls over changes), its functions, security and interactions (EU GMP Annex 11.4); a lack of evidence for the quality assurance of the software-developed process (EU GMP Annex 11.5), coupled with a lack of adequate validation evidence to support the use of GxP-related computer systems may very well be either a critical or major deficiency. Ranking will depend on the inspector's risk assessment.

Decisions on whether or not noncompliance merits pursuit of regulatory action will be based on a case-by-case evaluation. The general criteria for regulatory action is the same for most regulatory authorities:

- Nature and extent of deviations
- Effect on product quality and data integrity
- Adequacy and timeliness of planned corrective measures
- General compliance history

Regulatory citations for computer compliance by the FDA should reference the applicable predicate regulations. Enforcement by the MCA and other European regulatory authorities is through Annex 11 on computerized systems in the EU GMP Directive. They too will generally refer to the governing GMP requirement when citing computer system noncompliance.

Global Commitments

Care must be taken when making commitments to regulatory authorities not to inadvertently imply a global commitment to universal corrective action across an entire organization. While pharmaceutical and healthcare companies have an obligation to share learning across their organizations, this is not the same as making a formal commitment to specific corrective actions. Most noncompliances will be location specific to an individual site or facility. Only systemic issues should be considered for global commitments. Indeed, regulatory authorities will expect global commitments for such issues. Global commitments should be made in a timely fashion as part of a proactive recognition and management of an issue. Regulatory authorities will generally take further regulatory censure if they feel like they are having to persuade an organization to make a global commitment. This could mean issuing a Consent Decree for instance (see Chapter 1).

Poor Excuses

Many excuses have been given to GxP regulatory authorities when inspections have found validation to be deficient. Sam Clark, a former FDA investigator now working for Kempers-Masterson, listed some of the excuses offered to him when he was inspecting computer systems.[6] Clark listed these excuses under two categories. First, some responses were from pharmaceutical companies that simply did not validate the computer system that was inspected. The first category of excuses offered included the following:

- We do not have the resources.
- We have used the system for years.
- We do not have anyone who can do that.
- It was done, just not documented.
- We got the system from a reputable supplier.

None of these excuses could be accepted. Pharmaceutical and healthcare companies should not release drug products whose manufacturing practice is not completely validated.

Second, other excuses were presented by pharmaceutical companies for incomplete, inconsistent, or missing documentary evidence supporting validation:

- We do not need written procedures — all our people are professionals.
- We have excellent training programs.
- We have done it this way for years and have not had any problems.
- It was done, just not documented.

Without documentation, there is no physical evidence that validation took place, regardless of whether it was sufficient. Hence the saying "If it ain't written, it ain't done." GxP regulatory authorities may well believe on a personal level that a pharmaceutical or healthcare company did conduct suitable validation to accept GxP compliance without documentary evidence. It is imperative that pharmaceutical and healthcare companies collate documentation supporting their validation as evidence to be presented to GxP regulators on inspection.

ISO 9000 and Validation

Questions are often raised concerning the acceptability of ISO 9000 accreditation of pharmaceutical and healthcare companies and their suppliers in lieu of validation. GxP regulators do not accept this position. ISO 9000 and other software development processes do provide foundation for validation, but they do not replace the specific needs of GxP validation. This perspective is supported by recent research which suggests that ISO 9000 and other software development processes help improve bad practices rather than improve good practices, although good practices may improve a little (see Figure 6.4).[6] Those who are familiar with ISO 9000 will also know that the annual follow-up audits supporting an organization's ongoing certification by an accredited body almost always uncover problems with management procedures and their application, even though some of these audits are very brief — perhaps only a day long. Holding an ISO 9000 certificate does

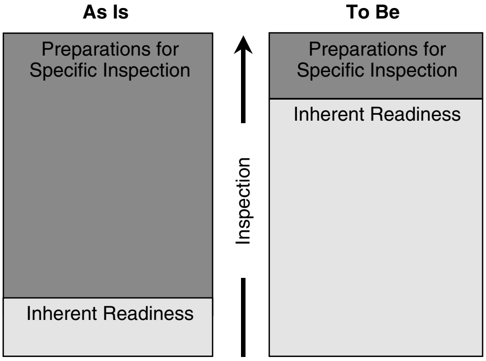

As Is
To Be
Preparations for Specific Inspection
Inherent Readiness
Inspection
Preparations for Specific Inspection
Inherent Readiness

FIGURE 16.1 Improving Inspection Readiness.

not guarantee high quality work; it is just an indicator of capability. GxP regulators would seem to be tight in their cautious attitude toward ISO 9000.

ENSURING A STATE OF INSPECTION READINESS

No matter how well a pharmaceutical or healthcare company believes it conducts validation, it will count for nothing unless during an inspection the regulator understands what has been done and can easily find his or her way around supporting documentation. Pharmaceutical and healthcare companies need to demonstrate they understand their responsibilities and are actively controlling compliance. To this extent a key feature in any validation exercise is inspection readiness (see Figure 16.1).

INVENTORY OF SYSTEMS

An inventory of systems and knowledge, of which one is GMP-critical, must be maintained and available for inspections. An MHRA preinspection checklist has this as one of its opening topics. The availability or otherwise of this information is a clear indicator of whether management is in control of its computer systems validation. The use of an inventory need not be limited to inspection readiness; it could also be used for determining supplier audits and periodic reviews, etc. Many pharmaceutical and healthcare companies use a spreadsheet or database to maintain this data. Where a site's inventory is managed between a number of such applications (perhaps one per laboratory, one for process control systems, one for IT systems), care must be taken that duplicate entries are avoided and, equally, that some systems are missed and not listed anywhere. It should be borne in mind that where spreadsheets and databases are used to manage an inventory, it should be validated just like any other GxP computer application.

SYSTEM/PROJECT OVERVIEWS

Management overviews should be available for systems and projects, giving a succinct summary of the scope of the system, essentially drawing boundaries and identifying functionality and use of the system/application concerned. Top-level functional diagrams and physical layout diagrams are highly recommended. It is also worthwhile considering developing some system maps showing various links between systems, dealing with both manual and automatic interfaces. Care must be taken to keep system maps up to date as new systems are introduced, old systems are decommissioned, and as the use and interfaces of some systems are modified to meet evolving user demands. Regulators are often interested in system interfaces, manual and electronic, and the validation status of connected systems. As a rule of thumb, all systems providing GxP information (data, records, documents, instructions, authorizations, or approvals) to a validated computer system should themselves be validated together with the interface.

Some regulators have requested guidance be given by pharmaceutical and healthcare companies on what is of particular relevance in terms of GxP functionality within their corporate computer systems. Such GxP assessments often fit neatly in the system overview. The reason for this request by regulators is to help them concentrate on key aspects of the system during an inspection without their getting bogged down in aspects of the system which are not of prime concern. It is easy for a regulator who is unfamiliar with a corporate computer system to get lost in its extensive and complex functionality (information overload). Needless to say, any GxP assessment information presented to a regulator must be understood and carefully justified.

Validation Plans/Reports and Reviews

It is likely that during a GxP inspection a regulator will ask whether or not a particular system has been validated. This line of investigation may stop with a yes/no response from the pharmaceutical or healthcare company. The line of investigation may, however, lead to a follow-up request to see the Validation Plan and Report for a system described as validated. Many of the computer systems used today have been in use over many years, and the regulator may also ask for any evidence of any Validation Reviews. These documents are, not too surprisingly, vital in demonstrating GxP compliance. It is not very clever to let a regulator discover a system in use with a Validation Plan but an incomplete or nonexistent Validation Report. Equally, if the system has been used for many years, it is more than reasonable to expect a recent Validation Review. Validation Plans, Reports, and Reviews should be checked to make sure they exist, are approved, and meet current regulatory expectations. In some instances pharmaceutical and healthcare companies, when considering this point, may put in place a review program to check that the items discussed above are complete and in place.

Documentation

It is vital to be able to easily locate documentation. Validation documentation that exists but cannot be retrieved as required during an inspection is worthless; it might as well not have been prepared in the first place. To this end an index to documentation should be maintained. All documentation supporting validation should be available at the site during inspections.

A procedure should be developed describing how to handle requests by regulators for documentation. Where requested, access to master (or copies of) documents (including raw data such as test evidence) should be provided within reasonable time frames, normally 24 to 48 h depending on circumstances. The Canadian Health Products and Food Branch Inspectorate, for instance, requires records to be accessible within 48 h.[7] The FDA has similar requirements for off-site paper-based archives.[8,9] Service Level Agreements between central support functions and sites should define the service levels for access to documentation.

Controlled copies of centrally held Validation Plans and associated Validation Reports should be issued to sites in advance of any regulatory inspection. Access to electronic copies of centrally held protocols and reports can be facilitated during regulatory inspections to avoid unnecessary delays waiting for paper master copies to arrive. Such access can be facilitated through e-mail or a shared system directory. In such circumstances it should be clearly stated to the regulator that these electronic copies may not adhere to regulatory electronic record/signature requirements but are being provided to assist the inspector in advance of hard copies being delivered to site.

In addition to documentation, access should be provided to support personnel with knowledge of the central application and documentation during regulatory inspections. Inspectors will not normally be authorized to access systems as a point of policy. An inspector who asks to see electronic documentation or electronic records can watch an authorized user query a system and make a printout.

Presentations

In practice computer systems are not perfect, and projects implementing applications will typically raise many management issues — that is life in the real world! The validation of any system/application will present its own special problems and solutions. Rationales need to be prepared and documented to demonstrate how problems and solutions have been managed. It is important to present a system/application in a positive light. Knowing how to effectively position problems and solutions will dramatically enhance the overall perception of the standard of validation on a system/application. The aim must be not to mislead an inspector but just to present validation issues in the vein of a glass half full rather than a glass half empty. If all reasonable endeavors have been made by a pharmaceutical or healthcare company to validate a system/application, this should normally be sufficient to satisfy an inspector, remembering that reasonable endeavors might include replacement where an original system/application cannot be validated to meet current regulatory expectations.

It is useful to prepare a brief presentation of each system subject to an inspection, which can be offered during the inspection. Remember, however, that some inspectors will not want an introductory briefing. Presentations should consist of perhaps four or five slides — certainly less than a dozen. The presentation slides should not be too detailed but should provide a broad picture describing a system/application and facilitate discussion. It is worthwhile letting the legal department look over the slides because there may be a danger of too high a level of information being interpreted as misleading if the detail of a system/application is examined. There is a careful balance to be struck between too much information and concise clarity. The slides should be in a suitable state that the inspector could be provided with a copy if requested.

Internal Audit Program

An internal audit program should be established if it does not already exist to cover the use of computerized systems. A schedule of audits should be planned placing priority on key topics subject to inspection such as data centre, laboratories, and manufacturing lines. It is useful to create a set of metrics to benchmark audit outcomes and monitor progress against audit actions. The audit should only mandate corrective actions where company policy, procedures, or regulatory requirements are not fulfilled. The audit can also be used to make recommendations for sharing examples of best practice with other sites or adopting best practice from other sites. Recommendations should not be included in audit metrics.

Mock Inspections

A mock inspection program should be developed if one does not already exist. Mock inspections should be as realistic as possible. Mock inspections on computer systems validation may be conducted as part of a more wide-ranging exercise or as a topic of a mock inspection in its own right.

The opportunity should be taken to actively coach personnel receiving the mock inspection, clearly identifying areas for improvement. If necessary, be prepared to withdraw individuals from the front line of a potential inspection if they are not readily capable of fulfilling this role. Sometimes doing yet more training will not be enough. It is important to accept that not everybody is suitable to place before an inspector.

Trained Personnel

Last but by no means least, the availability and use of trained presentation personnel during inspections is key. Those who present to an inspector should be permanent employees otherwise there may be an impression of dependence on quality from temporary staff whose loyalty and long-term commitment to a pharmaceutical or healthcare company could be questioned. Presenters need to be knowledgeable about systems/projects they are asked to front. They need to understand the validation approach and appreciate why certain project and validation decisions were made. The position papers, slide packs, and Validation Plans/Reports/Reviews should all help in this respect as long as the individuals concerned have enough time to study and digest the information they contain.

Individuals can feel quite exposed when they are informed they may be required to participate in an inspection, especially if they are likely to be asked to answer an inspector's questions. Individuals will benefit from training in this regard, and senior management can have confidence in how company members will interact with an inspector.

Presenters should be educated as to what to expect in the way of inspection protocols and regulatory practice. This aspect of training is likely to be tailored to the individual regulatory authorities. For instance, the FDA has a very different approach compared to many EU national regulatory authorities such as the MHRA. Those who front during an inspection need to be aware of these differences. Mutual recognition agreements should also be understood as information presented to one regulator in one context could be shared with another regulator out of context. Fronting an inspection can be a complex affair!

Training courses should be considered for:

- How to respond to inspector's questions
- How to escort/host an inspector
- How to provide copies of documentation to the inspector
- How to conduct yourself in front of an inspector
- How to report inspection findings to senior management

Training must cover what to say and what not to say: How to react when asked a question? How questions might be asked or phrased by an inspector? How to ask for clarification if requests are unclear? The aim is to remove any unnecessary fear.

Knowledge Management

Pharmaceutical and healthcare companies often rely on the personal knowledge and skills of individuals without formally managing this knowledge as a key corporate asset. Projects often do not employ suitable measures to safeguard and retain knowledge and skills particular to discrete project phases. Project documentation can become difficult to understand if it is overtaken by numerous change control records. For large systems documentation may become so complex in terms of number of documents or in terms of location of storage that it becomes very difficult to retrieve them in a timely manner. Change control records may become fragmented and give insufficient information to retrospectively understand change. Old and new computer system documentation may not be reconcilable if audit trails are not clearly maintained during changes to terminology or development methodologies. Furthermore, changes made over time may also inadvertently move system functionality away from its original intent.

The release of permanent staff from projects back into the business and their subsequent interdepartmental movements make their return to support inspections difficult and unreliable. Inspection readiness can be further frustrated by key staff taking on external positions or leaving the business for other reasons, e.g., voluntary redundancy. Many projects depend greatly on contracted resources, and turnover of such staff can be high. Once staff are dispersed, there may be an irretrievable loss of knowledge.

Succession plans need to be established and proper handovers arranged when staff leave. Refresher training should be considered for support staff. The reasons and benefits of historical changes in system functionality, terminology, and development methodologies must be documented in an easy-to-access and readily understandable way. An understanding of technological issues throughout the life of the system must be retained. Any outsourcing must clearly define user compliance accountabilities and mutual user/supplier responsibilities.

PROVIDING ELECTRONIC INFORMATION DURING AN INSPECTION

Regulatory authorities such as the FDA and MHRA may make requests to access electronic copies of documentation and records. The FDA, for instance, has a legal right to access such information electronically under the 21 CFR Part 11 (Electronic Records and Electronic Signatures) regulation. It is important to distinguish the difference between electronic documents/reports, electronic copies of desktop applications such as spreadsheets and databases, and electronic records that might be held on distributed/relational databases. The first two are relatively easy to extract as an entity to give to the inspector/investigator. The third is much more difficult.

Provision of Electronic Documents and Reports

The provision of electronic copies of documents/reports should be defined in procedures describing the general approach taken. Many inspectors may find authorized paper copies of documents/reports more useful as they are often easier to read than electronic text.

Provision of Electronic Copies of Desktop Applications

The provision of electronic copies of desktop applications such as spreadsheets and simple databases should be defined in procedures describing the general approach taken. Many inspectors will be able to execute these applications on their own computer systems. Because of this, authorized copies of relevant operating procedures and associated validation should normally be provided with the copy of the desktop application.

Provision of Electronic Records

The provision of electronic copies of records held on distributed/relational databases will need technical support to extract the right information to meet the regulators' needs without the regulator having to have sophisticated and expensive computer technology to read the information in a meaningful way. It is unlikely that inspectors/investigators will have the technical capability to read such information (e.g., they do not have their own SAP system to load data onto for investigation). For this reason provision of electronic records from distributed/relational databases is not typically useful to the inspector/investigator, and alternative ways of providing relevant information should be explored. A high-level procedure should describe the general process.

Direct Access to Electronic Information by Regulators

Direct access to electronic documentation and records should not be offered to the inspector/investigator. If direct access is requested by the inspector/investigator, the legal department should be informed. The inspector/investigator is not an employee of the company and would have to be properly approved, involving authorization, suitable training, and competency to have access. Such access could also violate the security (e.g., "closed system" status) of the company's computer systems. Similarly, inspectors/investigators should not be permitted to connect their own computer systems to pharmaceutical or healthcare companies' systems.

Use of Computer Systems by Regulators

Operational use of a computer system should not be offered to the inspector/investigator. Inspectors/investigators do not have the right to use company computer systems by themselves to access electronic information. Inspectors/investigators can watch an authorized user access a computer system, but they must not themselves directly use it. If direct access is requested by the inspector/investigator, the legal department should be informed. The inspector/investigator is not an employee of the company and would have to be properly approved, involving authorization, suitable training, and competency to have access.

Electronic Copies of Information

During inspections an inspector/investigator may request to see archived documents, or documents not held on the site under inspection. As discussed earlier in this chapter, pharmaceutical and healthcare companies should provide information in a timely manner and allow the inspection to flow naturally in accordance with the expectations of the inspector/investigator. If the physical transport of original documentation is not fast enough, then fax copies could be presented with the concurrence of the inspector/investigator. The time to fax large documents may not make this approach practical even with high-speed fax machines. In this situation it may be that just the main body of documents are faxed without appendices and attachments. If this is still too slow, then again, with the agreement of the inspector/investigator, electronic copies might be retrieved directly from company databases or sent to the site under inspection by e-mail and printed locally. In this latter situation the inspector/investigator must understand that the printed copies are being presented to aid the inspection by removing delays. These printed documents are not claimed to be compliant with electronic record/signature requirements. This approach must only be taken upon a specific request/authorization from the inspector/investigator.

Where copies of electronic documentation and records are provided to an inspector/investigator to take away, they should be provided on read-only media (preferably write-once, read-only). The same issuance process should be followed as for paper documentation (e.g., signed handover of copy to inspector/investigator, and an exact duplicate copy made on the same media as provided to the inspector/investigator for retention by site). Ad hoc electronic reports from computer systems specifically requested during inspections do not have to be validated.[10]

INSPECTION ANALYSIS

BioQuality analyzed FDA inspection observations on computer compliance made in 2002 by category of system (see Figure 16.2).[11] Recent FDA Warning Letters referring to computer systems validation that were issued between 1999 to 2002 are listed in Appendix 16E. The 176 observations relating to computer systems are analyzed by system type in Figure 16.3.

A life-cycle analysis of the FDA Warning Letters listed in Appendix 16E is presented in Figure 16.4. Of the total observations, 15% relate to Project Initiation, typically a lack of validation. Another 19% of the observations related to System Specification, Design and Development, System Build, and Development Testing. This figure may seem low given that 28% of observations relate to User Qualification. It could be argued that many User Qualification observations could be avoided if there were better system development. The medical device regulatory authorities have already recognized this trend from their inspection analysis and are putting more emphasis on system development during their inspections. Pharmaceutical and healthcare companies should also expect increasing regulatory focus on system development.

Operation and Maintenance account for one in three FDA Warning Letters observations related to computer systems. The majority of these are for data integrity and system security. Many observations are examples of bad practice and highlight the need for ongoing compliance activities

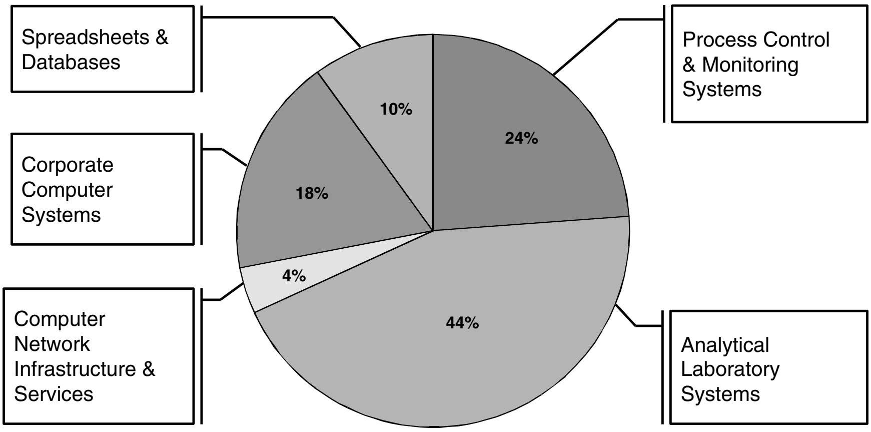


FIGURE 16.2 FDA 483 Observations on Computer Applications.

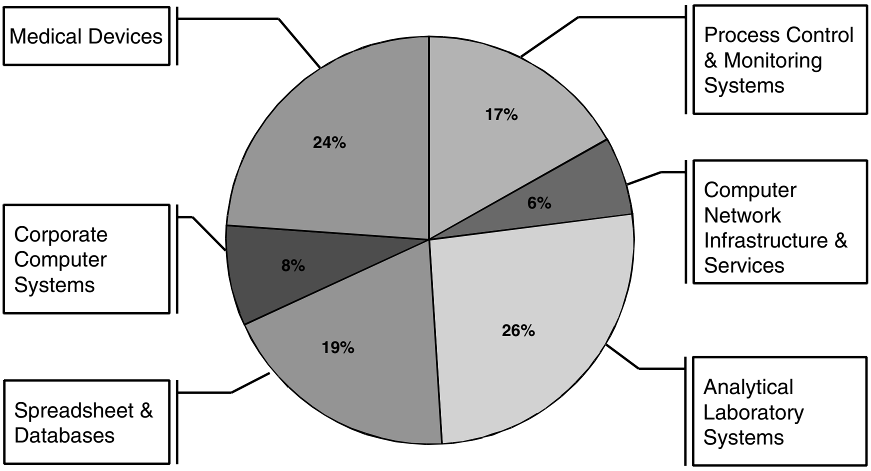


FIGURE 16.3 FDA Warning Letter Observations on Computer Applications.

during the operational life of a computer system. Validation does not end when a system is authorized for use.

Potential Causes of Validation Failure

ACDM/PSI have summarized common causes of validation failure.[12] Although their review is based on clinical systems the summary is equally applicable to all GxP computer systems.

- Inadequate documentation of plans.
- Inadequate definition of what constitutes the computer system.
- Inadequate definition of the expected results.
- Inadequate specification of the software (e.g., user requirements, functional specification).
- Software does not meet specification.
- The source code for the software is not available.
- Inadequate specification of the computer hardware and operating environment for which the system is designed to work.
- The computer hardware or operating environment differs from the specification.

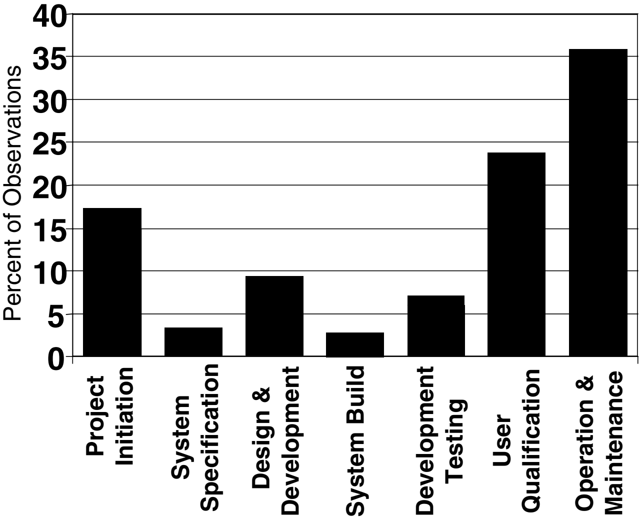


FIGURE 16.4 FDA Warning Letter Observations by Life Cycle.

- The way the system should be used is not defined.
- Inadequate consideration given to centralized IT infrastructure, e.g., network management, procedures, and responsibilities.
- The intended use of the system is clearly defined, but users are not aware of it or do not adhere to it.
- The system has been inadequately tested, or the testing has been inadequately documented.
- Documented standard procedures for the development, maintenance, operation (including security), or use of the system are inadequate.
- Documented procedures for disaster recovery are inadequate.
- System developers or other personnel involved with system implementation and use are not properly qualified, trained, or competent.
- Documentary evidence to demonstrate qualification, training, and competence level of personnel involved with the system is not available.
- Documentation for all or part of the validation process does not exist, or cannot be located.
- Evidence of review and approval of validation documentation by qualified staff is not available.
- Inadequate change control over any element of the system (i.e., hardware, software, procedures, people).

REFERENCES

1. FDA (1987), *General Principles of Process Validation,* Food and Drug Administration, Center for Drug Evaluation and Research, Rockville, MD.
2. Food and Drug Inspection Monitor (2000), "FDA To Do System-Based Audits of Drug Companies," 5 (10), published by Washington Information Source Co., October.
3. *The Gold Sheet* (1996), 30 (7), July.
4. MCA (2002), Computerised Systems and GMP — Current Issues, Top Ten GMP Inspection Issues, Royal Garden Hotel, London, September 24.
5. FDA (1992), Compliance Program Guideline Manual, 7346.832, Pre-Approval Inspection Investigations, Food and Drug Administration, Rockville, MD.
6. Wingate, G.A.S. (1997), *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice*, Interpharm Press, Buffalo Grove, IL.

7. Canadian Health Products and Food Branch Inspectorate, Draft Good Manufacturing Practice Guideline (2001).
8. U.S. Code of Federal Regulations Title 21: Part 203, *Prescription Drug Marketing.*
9. U.S. Code of Federal Regulations Title 21: Part 205, *Guidance for State Licensing of Wholesale Prescription Drug Distributors.*
10. FDA (2002), *Investigations Operations Manual*, Office of Regulatory Affairs, May.
11. Private Communication with Scott Lewis, February 2003.
12. ACDM/PSI (1998), "Computer Systems Validation in Clinical Research: A Practical Guide," Association of Clinical Data Management (ACDM) and Statisticians in the Pharmaceutical Industry (PSI), Version 1.1, December.
13. Pharmaceutical Inspection Co-operation Scheme (2003), Good Practices for Computerised Systems in Regulated GxP Environments, Pharmaceutical Inspection Convention, PI 011-1, Geneva, August.
14. Regina Brown (2001), Inspecting a Laboratory Computerized System, GAMP Americas Meeting, Philadelphia, March 22.
15. FDA (1998), *Guideline to Inspections of Computerized Systems Used in Food Processing Industry,* October.
16. Janis Halvorsen (2000), Georgia GMP Conference, *Gold Sheet*, November.

APPENDIX 16A
PREINSPECTION QUESTIONNAIRE[13]

The following information is sometimes requested by regulatory authorities prior to an inspection.

1. Details of the organization and management of IT and other Computer Services (from business IT systems to process control) on site.
2. State corporate policy on procurement of hardware, software, and systems for use in GxP areas.
3. IT/Computer Services standards and SOPs? (Attach list.)
4. Provide a list of all GxP-related computerized systems on site by name and application for business, management, information, and automation (equipment and process control) levels. Indicate the totality of the inventory of computerized systems and indicate links with other sites/networks, etc.
5. For the systems identified as GxP related, has the company identified the critical systems, interfaces, subsystems, modules and programs that are relevant to GxP, product quality, and safety? If so, please cross-reference lists provided for Question 4 above.
6. What documentation generally exists to provide up-to-date descriptions of the systems and to show physical arrangements, data flows, interactions with other systems, and lifecycle and validation records? Comment as to whether all of these systems have been fully documented and validated.
7. Comment on the qualifications and training aspects of personnel engaged in design, coding, testing, validation, installation, and operation of computerized systems (Specifications, Job Descriptions, Training Logs).
8. What is the firm's approach to assessing suppliers of hardware, software, and systems?
9. How does the firm determine whether purchased or "in-house" software has been produced in accordance with a system of quality assurance?
10. What project management standards and procedures are in place for the development of applications and validation work? (List key titles and reference numbers.)
11. What approach is taken to the validation and documentation of older systems where original records are inadequate? (Summarize and list systems undergoing retrospective documentation and justify the continued use of these systems.)
12. Has the firm determined whether GxP-critical systems conform to electronic data processing needs, accuracy, and controls (including retrieval of archived records) for quality records as required by 91/356/EEC Article 9 and EU GMPs 4.9 (*inter alia*)?

APPENDIX 16B
GLP INSPECTION CHECKLIST[14]

- The main focus is on quality of drug products and integrity of associated data.
- The integrity of the data and how it is maintained gives her an overall judgment of product quality. Therefore, procedures should be in place to assure the integrity of all processes and data. During an inspection, missing data are cues that something is amiss and will cause the inspector to search further in this area.
- Key questions during an inspection:
  - Who has access to the data?
  - How is access controlled?
  - What operations are permitted (read, write, edit, and delete)?
  - How can you demonstrate that what is reported is the same as that stored?
  - Have you evidence that backup and restore of data has been tested and can be demonstrated?
- A company policy and guideline on Computer System Validation (CSV) is expected. Documentation, SOPs are reviewed; diagrams, flowcharts on the systems are requested.
- All systems should be validated and calibrated before implementation.
- Change Control of the system is reviewed; if it is lacking, this is viewed as a QA oversight.
- Audit trails must exist, and restrictions on delete functions are required.
- Passwords on the system must be controlled and changed periodically.
- If electronic signatures are used, procedures must be in place for how they are used and maintained.
- 21 CFR Part 11 — the adequacy and timeliness of planned corrective measures. The company is expected to have a reasonable timetable and must be able to demonstrate progress and see the corrective actions that have been executed.
- Lot systems and data are typically reviewed.
- For a chromatograph system, stability tests results are examined to compare the paper record with the electronic record.
- All raw data should be retained.
- Security measures must be in place, especially for HPLC systems.
- Quality Control personnel must know everything about the system, the validation, training records, etc. For example: An individual may be asked about data that reside on a system and then asked to retrieve the archived data in question. This is to ensure that the individual knows what he or she is talking about.
- GMP training is required for all people involved in the manufacturing process.
- Overall the functional, operational, and security features are investigated.

APPENDIX 16C
GMP INSPECTION CHECKLIST[15]

- Determine the critical control points (base investigation on FMEA or other hazard analysis technique). Examples are:
  - Pasteurization
  - Sterilization
  - pH control
  - Temperature control
  - Cycle timing
  - Record keeping
  - Control of microbiological growth
- For those critical control points controlled by computerized systems, determine if failure of the computerized system may cause drug adulteration.
- Identify computerized system components including:

- Hardware Inventory
  - Input devices
  - Output devices
  - Signal converters
  - Central Processing Unit
  - Distribution system
  - Peripheral devices

- Hardware

  Obtain a simplified drawing of the computerized system (covering major computer components, interface, and associated system/equipment). For computer hardware determine the manufacturer, make, and model number.

- Software Inventory
  - Inventory of files (program and data)
  - Documentation
  - Manuals
  - Operating procedures

- Software

  For all critical software determine:
  - Name
  - Function
  - Inputs
  - Outputs
  - Set-points
  - Edits
  - Input manipulation of date
  - Program overrides

- Version Control
  - Who developed the software (standard, configured, customized, bespoke)?
  - Software security to prevent unauthorized changes.
  - Computerized systems input/outputs are checked.

  - Obtain simplified drawing of overall functionality of collective software within computerized systems

- Data
  - What data are stored and where?
  - Are data distributed over a network — how is it controlled?
  - How is compliance to electronic record regulations achieved?
  - How is data integrity verified?

- Personnel
  - Type (developer, user, owner)
  - Training records

- Observe the system as it operates to determine if:
  - Critical processing limits are met
  - Records are accurate
  - Input is accurate (sensor or manual input)
  - Timekeeping is accurate
  - Personnel are trained in systems operations and functions
- Determine if the operator or management can override computer functions. How is this controlled?
- How does the system handle deviations from set or expected results? Check all alarms, calculations, algorithms, and messages.

- Alarms
  - Types (visual, audible, etc.)
  - Functions
  - Records

- Messages
  - Types (mandate action?)
  - Functions
  - Records

- Determine the validation steps used to insure that the computerized system is functioning as designed.
  - Was the computerized system validated upon installation?
    - Under worst case conditions?
    - Minimum of three test runs?
  - Are there procedures for routine maintenance?
    - User manual
    - Vendor-supplied manual
    - Third-party support manual
    - Management manual
  - Does the equipment in place meet the original specifications?
  - Is validation of the computerized system documented?
  - How often is system
    - Maintenance performed
    - Calibrated
    - Revalidated
  - Check scope and records of any Service Level Agreements.

  - Are there procedures for revalidation? How often is revalidation conducted?
- Are system components located in a hostile environment that may affect their operation (ESD, RFI, EMI, humidity, dust, water, power fluctuations)? Are system components reasonably accessible for maintenance purposes?
- Determine if the computerized system can be operated manually. How is this controlled?
- Automated CIP (cleaning in place)
  - How is automated CIP verified?
  - Documentation of CIP steps
- Automated SIP (sterilization in place)
  - How is automated sterilization verified?
  - Documentation of SIP steps
- Shutdown Procedures
  - Does the firm use a battery backup system?
  - Is the computer program retained in the control system?
  - What is the procedure in the event power is lost to the computer control system?
  - Have backup and restore procedures been tested?
- Is there a documented system for making changes to the computerized system?
  Is there more than one change control system (hardware, software, infrastructure, networks)? Document for each challenge:
  - The reason for the change
  - The date of the change
  - The changes made to the system
  - Who made the changes

  How do they interface? Challenge change history, verify audit trail?
- What are the auditors' impressions of:
  - Presentation of validation
  - State of documentation
  - State of compliance
  - Maintaining validation
  - Requirements for revalidation

APPENDIX 16D
ELECTRONIC RECORD/SIGNATURE INSPECTION CHECKLIST[16]

- Review firm's record-keeping requirements.
- Predicate record-keeping requirements even if not electronic.
- Determine if the firm has procedures for providing electronic and paper copies of records.
- What is the overall security of the electronic record-keeping system?
  - Can records be altered without a trace?
  - Do systems by design fail to record noncompliant information?
  - Are password systems robust (sticky notes, same as user names, easily guessed strings)?
  - Is access to the system restricted? Normally or when station is unattended?
  - What are the procedures in the event passwords or tokens are compromised?
- Documentation of the following?
  - Functional specifications
  - Design specifications — high level and detailed
  - Code documented and commented
  - Testing plans, documented test results
  - Review of all documentation by knowledgeable people
  - Release criteria and maintenance plan
  - Validation plans, procedures, and report
- Does the firm know its own deficiencies and have specific corrective action plans?
- Can the firm document progress toward achieving its corrective action plans?
- Does the firm "maintain a validated state"? Is validation documentation current and readily available?
- Has the firm trained IT and technical personnel on FDA regulations?
- Have administrative controls been put into effect?

APPENDIX 16E
RECENT FDA WARNING LETTERS

	Life Cycle								Application							
Company Name	**Project Initiation**	**System Specification and Supplier Selection**	**Design and Development**	**System Build**	**Development Testing**	**User Qualification & Authority to Use**	**Operation and Maintenance**	**Phase-Out and Withdrawal**	**Analytical Laboratory System**	**Process Control or Monitoring System**	**Spreadsheet and Database Applications**	**Corporate Computer System**	**Computer Network Infrastructure and Services**	**Medical Device**	**Electronic Records and Electronic Signatures**	**Number of Paragraphs in Warning Letter Dealing with Different Computer Issues**
02/99 Hydro Medical Sciences Inc.						**X**	**X**		**X**							1
04/99 Fairbanks Memorial Hospital							X				X					1
04/99 General Electric Company					X	X	X							X		9
04/99 Florida Blood Services			X	X			X							X		4
05/99 Glenwood LLC						X	X		X							2
05/99 Picker International Inc.			X	X	X	X	X							X		12
06/99 Cypress Bioscience Inc.						X	X				X					1
06/99 Solvay Pharmaceuticals B.V.							X			X		X				6
07/99 Gensia Sicor Pharmaceuticals Inc.						X	X		X							2
08/99 Drager Medizintechnik GmbH	X										X					1
08/99 Linweld Inc.					X	X	X					X	X		X	11
10/99 Synthes						X	X							X	X	5
11/99 Apheresis Technologies Inc.							X				X				X	1
12/99 Hoffmann-LaRoche			X				X		X	X						3
03/00 Schein Pharmaceutical Inc.			X		X	X			X				X			4
03/00 Johnson Matthey							X		X							1
04/00 Harper Hospital					X	X			X							4

	Life Cycle								Application							
Company Name	Project Initiation	System Specification and Supplier Selection	Design and Development	System Build	Development Testing	User Qualification & Authority to Use	Operation and Maintenance	Phase-Out and Withdrawal	Analytical Laboratory System	Process Control or Monitoring System	Spreadsheet and Database Applications	Corporate Computer System	Computer Network Infrastructure and Services	Medical Device	Electronic Records and Electronic Signatures	Number of Paragraphs in Warning Letter Dealing with Different Computer Issues
05/00 Intersurgical Ltd.	X													X	X	2
05/00 Schering Laboratories							X		X							1
06/00 Medical Industrial Equipment Ltd.							X							X	X	4
06/00 Sani-Pure Food Laboratories						X	X		X							4
06/00 A&L Laboratories			X			X					X					2
06/00 Jiangsu Hengrui Medicine						X	X		X							2
07/00 Integrity Pharmaceuticals Corp.						X	X		X (system not specified)							2
07/00 Rhodia Inc.							X		X							1
08/00 Baxter Healthcare Corp.						X			X	X					X	3
09/00 Leiner Health Products	X												X		X	2
10/00 Spolana a.s.	X					X	X		X							6
10/00 Contract Pharmacal Corp.						X	X			X						1
11/00 Alcon Laboratories Inc.		X	X		X	X										2
11/00 SOL Pharmaceuticals Ltd.						X	X		X							3
11/00 Sybron Chemicals						X	X		X							1
12/00 Societa Italiana Medicinali	X					X	X			X		X				1
12/00 Chemrich Holdings							X		X							1
01/01 Pharmacia Corp. (Sterile)		X	X			X	X			X	X		X			11
01/01 Pharmacia Corp. (API)		X	X			X	X			X			X			9
01/01 Aventis Behring							X					X				1
01/01 Biological Research Solutions							X		X							2
01/01 Allergy Laboratories	X										X					1
01/01 DSM N.V.							X		X			X				3

03/01	Eli Lilly and Company							X			X						1
03/01	Zeus Scientific Inc.	X											X			X	1
04/01	Stough Enterprises						X	X		X						X	2
04/01	Cardiomedics Inc.	X		X											X		2
04/01	Neurocontrol Corp.	X										X			X	X	1
05/01	Zenith Goldline Pharmaceuticals							X		X							1
06/01	Meridian Bioscience	X										X			X	X	2
07/01	Cardinal Health						X	X		X		X					5
07/01	SeQual Technologies Inc.	X									X						1
07/01	Esolyte Inc.							X				X					1
07/01	Kaken Pharmaceuticals							X		X	X						2
07/01	EP MedSystems	X										X					1
07/01	Aventis Bio Services							X				X					1
07/01	American Blood Resources Assoc.		X	X		X		X						X	X		4
08/01	Paradigm Medical Industries							X				X			X	X	2
08/01	Farouk Systems Inc.	X								X			X				1
08/01	Medical Instruments Technology	X	X	X	X						X				X		3
09/01	Pharmakon Labs	X										X					1
09/01	Utah Medical Products	X													X	X	1
09/01	Braun Medical	X													X		1
09/01	Christ Hospital	X						X			X					X	2
09/01	Cleveland Medical Devices						X	X							X		3
09/01	Dentsply International Inc.					X									X		1
09/01	Total Medical Info. Mgt Systems		X	X				X				X			X		1
10/01	Northeast General Pharma							X		X							1
10/01	Michigan Instruments							X							X	X	1
10/01	Bunnel Inc.	X										X			X		1
10/01	Luneau	X						X				X			X	X	2
10/01	Neil Laboratories						X			X							2
10/01	Sorenson Development Inc.	X						X			X	X			X	X	1
12/01	Natural Technology Inc.						X				X						1
12/01	Medical Device Services	X										X	X				3
12/01	Cardinal Enterprises						X			X	X					X	2
01/02	Sysmex Corp.				X										X		1
01/02	Pharmaceutical Distribution Sys.						X				X						1
02/02	GOJO Industries							X			X						1

	Life Cycle								Application							
Company Name	**Project Initiation**	**System Specification and Supplier Selection**	**Design and Development**	**System Build**	**Development Testing**	**User Qualification & Authority to Use**	**Operation and Maintenance**	**Phase-Out and Withdrawal**	**Analytical Laboratory System**	**Process Control or Monitoring System**	**Spreadsheet and Database Applications**	**Corporate Computer System**	**Computer Network Infrastructure and Services**	**Medical Device**	**Electronic Records and Electronic Signatures**	**Number of Paragraphs in Warning Letter Dealing with Different Computer Issues**
04/02 American Dental Technologies						X	X							X		1
04/02 A-Vox Systems Inc.					X									X		1
07/02 Earlham College						X	X		X						X	1
10/02 Abbott Laboratories	X									X						1
Medical Devices	11	3	6	4	5	5	13	0	0	3	7	0	1	24	10	63
Pharmaceutical and Healthcare	12	2	6	0	4	27	34	0	27	15	13	8	5	0	8	127
TOTAL	23	5	12	4	9	32	47	0	27	18	20	8	6	24	18	190

17 Capabilities, Measures, and Performance

CONTENTS

The ability to perform validation cost-effectively is dependent on an organization's understanding of requirements and its validation capability. This chapter applies the established Capability Maturity Model (CMM) to computer validation. Examples of validation metrics and measures are examined. The metrics cover prospective validation as well as operation and maintenance of computer systems. Lean Manufacturing and Six Sigma are promoted as tools that organizations can use to streamline and improve the performance of their validation processes.

VALIDATION CAPABILITY

Experience has shown significant advantages for suppliers of computer systems (internally within pharmaceutical and healthcare company organizations, systems integrators, and equipment vendors) who improve their validation capability. In particular, the risk of noncompliant validation is reduced, and conducting validation itself becomes more cost effective and time efficient.

A framework recognizing the symbiosis between both process and product was first proposed by the Software Engineering Institute (SEI) at Carnegie Mellon University and called the Capability Maturity Model (CMM). CMM is based on five evolutionary levels of capability from *ad hoc*, chaotic processes to mature, disciplined processes:[1]

Level 1: The quality process is characterized as *ad hoc* and occasionally even chaotic. Few processes are defined, and success depends on individual efforts and heroics.

Level 2: Basic project management processes are established to track cost, schedule, and functionality. The necessary process discipline is in place to repeat earlier successes on projects with similar applications.

Level 3: The quality processes for both management and engineering activities is documented, standardized, and integrated into a standard quality process for the organization. All projects use an approved, tailored version of the organization's standard quality process for developing and maintaining systems.

Level 4: Detailed measures of the quality process and product quality are collected. Both the quality process and products are quantitatively understood and controlled.

Level 5: Continuous process improvement is enabled by quantitative feedback from the process and from piloting innovative ideas and technologies.

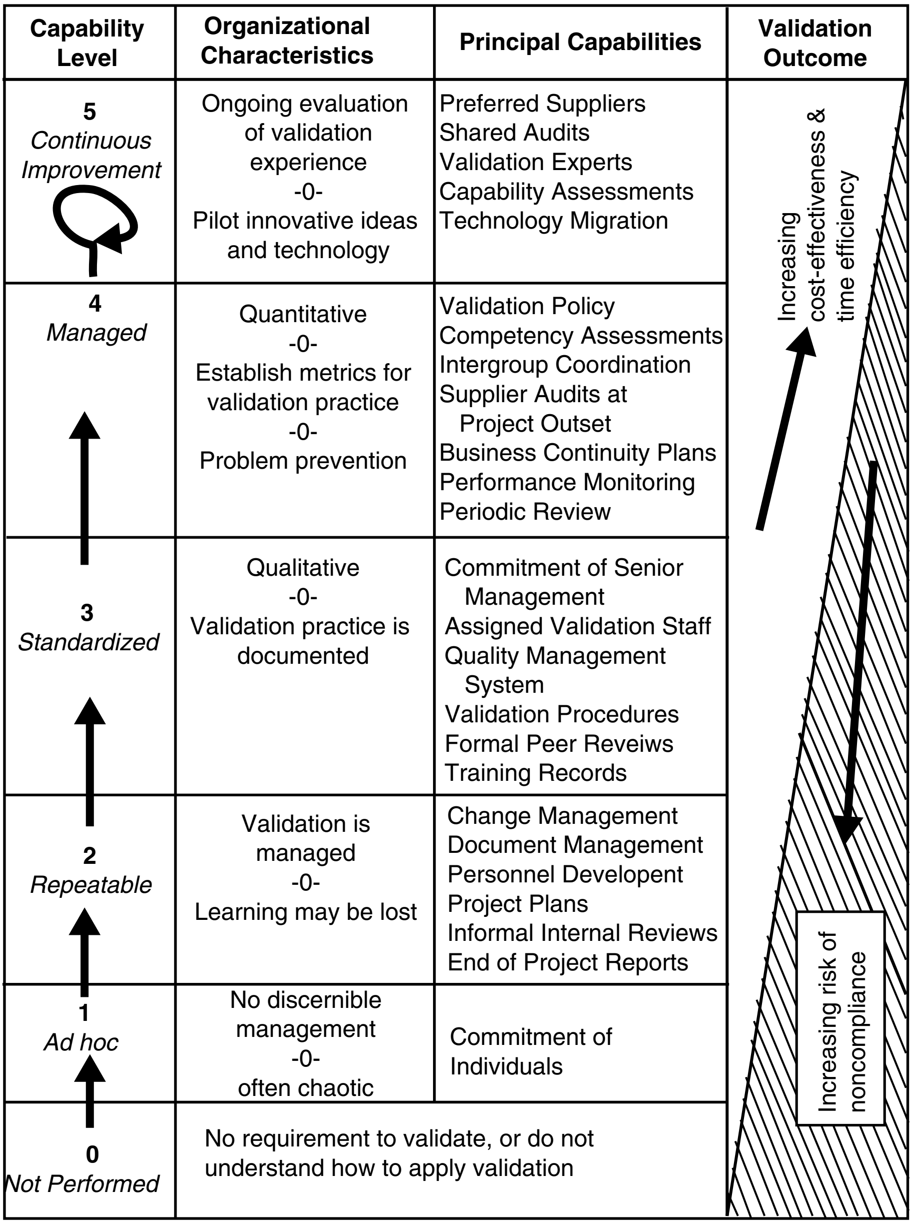
An assessment of an organization against CMM can be used to generate a profile from which an organization can identify necessary initiatives to support an improvement in its quality assurance capability. In the same manner, the profile can help organizations prevent management missing crucial activities supporting a level of capability. Walter Royce suggests that only 25% of companies can be considered Level 3 or above,[2] that is, at a competency level similar to ISO 9000.

An adaptation of SEI's CMM for application to the validation of computer systems is shown in Figure 17.1. This is not a definitive adaptation and is based on the principles of CMM. An additional sixth level (0) has been added to deal with organizations that have not yet embarked on any validation capability.

CAPABILITY APPRAISALS

Pharmaceutical and healthcare companies should consider placing themselves within the framework. Organizations will often have a capability profile that includes elements of capabilities from several levels. The assessed level of capability will be that with which an organization is entirely compliant.

A sample questionnaire is given in Appendix 17A to help evaluate which level of validation capability an organization fits into.[3] The best way to conduct an appraisal is by an unannounced surprise audit. Mature organizations should by their nature be inspection ready. When conducting an appraisal, specific examples should be documented to demonstrate a capability and a note made as to whether the capability is readily observable in more than one context. During any appraisal care must be taken to assess the true organization capability rather than massaging the assessment outcome. If the whole audit process takes longer than 5 to 10 man-days effort inclusive of auditor and auditee, then it probably indicates that evidence is not readily available.



Capability Level	Organizational Characteristics	Principal Capabilities	Validation Outcome
5 *Continuous Improvement*	Ongoing evaluation of validation experience -0- Pilot innovative ideas and technology	Preferred Suppliers Shared Audits Validation Experts Capability Assessments Technology Migration	Increasing cost-effectiveness & time efficiency
4 *Managed*	Quantitative -0- Establish metrics for validation practice -0- Problem prevention	Validation Policy Competency Assessments Intergroup Coordination Supplier Audits at Project Outset Business Continuity Plans Performance Monitoring Periodic Review	
3 *Standardized*	Qualitative -0- Validation practice is documented	Commitment of Senior Management Assigned Validation Staff Quality Management System Validation Procedures Formal Peer Reveiws Training Records	
2 *Repeatable*	Validation is managed -0- Learning may be lost	Change Management Document Management Personnel Developent Project Plans Informal Internal Reviews End of Project Reports	Increasing risk of noncompliance
1 *Ad hoc*	No discernible management -0- often chaotic	Commitment of Individuals	
0 *Not Performed*	No requirement to validate, or do not understand how to apply validation		

FIGURE 17.1 Computer Validation Capability Model.

Capability Characteristics

The characteristics associated with each level of validation capability can be summarized as follows:

Level 1: Unpredictable performance in terms of costs, schedule, and quality. Possibly less that 20% of issues raised will be resolved. There are often difficult team interactions, mainly because there are no defined processes to assist implementation of computer systems.

Level 2: Repeatable performance from project to project in terms of costs, schedule, and quality, but no performance improvements. Typically less than 60% of issues raised will be resolved. There are likely to be some difficult team interactions, but basically the team will support each other.

Level 3: Better performance on successive projects in terms of costs, schedule, and quality. Less than 25% of issues raised remain unresolved. Team members mutually support one another.

Level 4: Significant performance benefits in successive projects in terms of costs, schedule, and quality. Less than 10% of issues raised remain unresolved. Computer implementations are trustworthy and consistently delivered with full functionality, within budget, and on schedule. Highly cooperative team interactions.

Level 5: Continually improving performance benefits on successive projects in terms of costs, schedule, and quality. Of issues raised, 100% are resolved. Teams are cohesive and seamless. Level 5 organizations typically are specialized into niche expertise.

In the capability framework presented, it has proven quite difficult to align validation activities between Level 3 and Level 4. The divide suggested is based on an established project-by-project capability of Level 3 compared to the ongoing inherent organizational capability of Level 4. In the framework, Level 4 equates to a fully compliant regime of validation for GMP.

Capability Assessment Outcomes

Level 1 and Level 2 assessment outcomes usually denote pharmaceutical and healthcare companies whose senior management are still not committed to the implementation of validation and rely on their subordinates to enact validation without the practical support they could offer. Computer validation is often characterized by firefighting.

Pharmaceutical and healthcare companies typically like to think of themselves at Level 3. A compliant validation capability would rank between Level 3 and Level 4. A validation capability below Level 3 will almost certainly be regarded by GMP regulatory authorities as insufficient for GMP. Noncompliance may not be identified on an initial or limited inspection. Pharmaceutical and healthcare companies must not become complacent and should prepare for further and detailed inspections. The regulators' position with individual cases of GMP noncompliance will vary with the severity of the deficiencies they find. Generally, they will give the pharmaceutical or healthcare company a period of time to take corrective actions before they take the matter further.

Less than 1% of organizations have a Level 5 capability. Level 5 signifies the opportunity for pharmaceutical and healthcare companies and their suppliers to reap the reward of tangible benefits discussed earlier in Chapter 1. Principal capabilities associated with Level 5 might include selecting preferred suppliers, conducting joint Supplier Audits with other organizations and sharing audit reports, developing in-house validation experts, conducting internal capability assessments to identify improvement opportunities, and planning for technology migration to exploit any new innovations.

Supplier Capability Assessments

A slightly different situation exists with suppliers of computer systems to the pharmaceutical and healthcare industry. Suppliers generally understand the benefits of a quality approach, but unless the pharmaceutical and healthcare industries form a significant proportion of their sales, then it is unlikely they really understand how their quality approach relates to the requirements of validation, despite what the suppliers' salesmen might say. For this reason, while a supplier might be Level 3 or 4 on the CMM, the same supplier is likely to be Level 2 on the validation capability framework. It is very important for pharmaceutical and healthcare companies to undertake a Supplier Audit to determine the actual capability of their suppliers, and, in particular, to assess the competence of personnel assigned to their project.

Benefits of Improving Capability

Annual returns on original investment of an enhanced quality assurance capability for computer systems should be over fourfold. Stepping up one level in the capability framework should reduce costs by 20 to 25%.[4] It typically takes an organization about 2 years of concerted effort to go up a level of capability. This is because capability is linked with culture. It is relatively easy to establish policies and procedures; it is much harder to build a complementary inherent quality culture. For instance, QA groups are often perceived as striving for 100% perfection on computer validation to mitigate all risk of noncompliance ("zero tolerance"). Consequently, development groups often push back on QA, sometimes to the extent of compromising basic quality assurance practices ("dumbing down"). This is because they lose sight of the fact that "fit for purpose" in the pharmaceutical and healthcare industries means not only that the system works and fulfills industry standards, but also that the computer system satisfies regulatory requirements. A developing organizational capability must break down these barriers and foster a collaborative working environment.

PROJECT VALIDATION METRICS

Pharmaceutical and healthcare companies have the opportunity to use validation to reduce the cost of ownership for the computer systems they use. The cost of validation of a project represents an investment that will be more than recouped in lower maintenance costs which, anecdotally, can be reduced by 50 to 80%. With maintenance perhaps responsible for half the lifetime cost, this could give a return on investment of 1 to 3 years.

Figure 17.2 collates some project validation metrics from recent publications and conferences (data sources listed in Appendix 17B). These metrics have been collected to help practitioners understand validation and the allowance that should be made during project planning. Equally, the metrics will help challenge project planning where resource requirements seem excessive or too low to be credible. Ways of reducing validation costs are explored later in this chapter.

When reviewing Figure 17.2, remember that there was no standard definition between the sources identified as to what exactly constituted validation. The percentage costs are a rough indicator as part of overall project cost and, not surprisingly, they increase as the complexity and customization of systems increase.

Analytical laboratory systems in Figure 17.2 include analytical instruments with coupled laptops or personal computers (e.g., HPLC, GC, LC) and chromatography data systems. The metrics

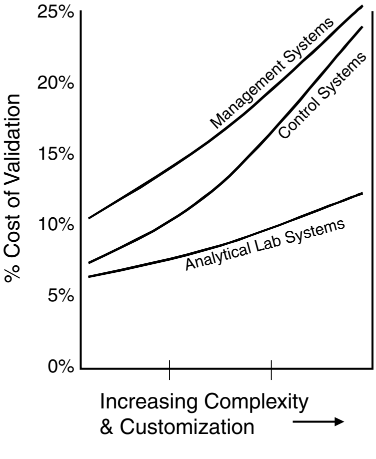


FIGURE 17.2 Relative Project Cost of Validation.

TABLE 17.1
Comparing the Cost of Validating COTS Software

Type of Application	Relative Effort to Validate
Custom (Bespoke) Application	100%
Configured COTS Application	55%
COTS Application without Configuration	25%

provided here assume that analytical laboratory systems are based on Commercial Off-The-Shelf (COTS) products. LIMS are specifically excluded here and included in the management category of computer systems since they predominantly provide an information management function. Validation costs should be low because of the standard nature of the applications and only increase slightly with the relative size and complexity of the applications.

The control systems referred to in Figure 17.2 cover Programmable Logic Controllers (PLCs) as the simplest example, Supervisory Control and Data Acquisition (SCADA) systems, and Distributed Control Systems (DCS). These systems are typically COTS-based products but have extensive configuration. As the systems get larger, there tends to be a growing number of customized interfaces to subsystems and control instruments. Control systems may have many hundreds, even thousands, of associated instruments. This leads to a larger increase in validation costs compared to analytical laboratory systems relative to the growing size and complexity of the overall application.

Management systems in Figure 17.2 include simple MRP, LIMS, MRP II, and integrated ERP systems. The relative increase in validation costs compared to growing size and complexity is relatively linear but with a greater rate of increase than analytical laboratory systems. This is because these applications typically involve extensive configuration. Customization is not such a significant factor, with system functionality being provided by plug-in modules provided by the supplier of the core product or a certified product partner.

It is important to understand too that customization will further increase validation costs (e.g., custom PLC applications will probably incur a 10% validation cost rather than the 5% indicated in Figure 17.2, which assumed a configured COTS application). Table 17.1 compares the relative increased costs associated with custom applications, configured COTS applications, and standard COTS products that do not require configuration. The further the cost of validation decreases, the more the standard software can be leveraged. The exploitation of standard software is explored further in Chapter 14.

Design and Development Metrics

An analysis of over 130 computer projects of various sizes (summarized in Table 17.2) emphasizes the benefits of conducting Design and Development Reviews. It is suggested that the combination of effective Design Reviews and Source Code Reviews reduces overall project costs by about 10% compared to projects not implementing these reviews (this saving is achieved by detecting errors before testing). Too often, however, such reviews are ill-defined and ineffective. Without effective Design Reviews the design effort may be doubled because of the need to clarify ambiguous specifications or correct errors during coding and testing. Similarly, ineffective or missing Source Code Reviews typically incur up to an additional 25% coding effort during testing to correct errors. A detailed analysis of validation costs by Murtagh emphasizes how Source Code Reviews take much less effort to perform when conducted by the software developer's organization rather than by the user. This is because the developer's organization is more familiar with the type of software and understands the design intent better.[5] Indeed, this principle holds true for all those aspects of validation that can be supported by a supplier organization. It is not uncommon to find about

TABLE 17.2
Project Metrics

	Typical Project Effort					
Life-Cycle Phase	**Analytical Lab Systems**	**Control Systems**	**Management Systems**	**Web-Based Systems**	**Typical Error Detection Capability**	**Normalized Effort to Fix Error**
System Specification & Selection	40%	35%	25%	55%	20–45%*	1
Design & Development						3–5
Coding, Configuration & Build	20%	25%	40%	15%	5–30%**	5–10
Development Testing & User Qualification	40%	40%	35%	30%	50–75%	10–50

* Design Review.
** Source Code Review.

two thirds of a pharmaceutical or healthcare company's QA department input to a project revolving around resolving supplier-related quality and compliance issues.

Testing Metrics

As previously stated in Chapter 10, testing should be designed to detect errors in the developed computer system. If the testing process itself is not robust, that, too, will induce errors and rework. The testing conducted on 85 computer systems used across primary and secondary pharmaceutical manufacturing in several companies is analyzed here to examine test failures and how they were managed to closure.

Test failures were attributed to a number of causes as illustrated in Figure 17.3. Operator error while executing the test case accounted for 1% of test failures. These tests were repeated once the error was understood. Incorrect setup also accounted for 1% of test failures. These tests too were repeated with the correct setup once the error was understood. Clarity problems with the test method and acceptance criteria accounted for 40% of test failures. Only the remaining 58% of tests did what they should have done, which is detect system errors. That is, 42% of test failure processing was avoidable if a more robust test process was adopted. Of the errors identified, 37% were classed as significant, and 63% as not significant. Resolution of these errors impacted specification and design documents.

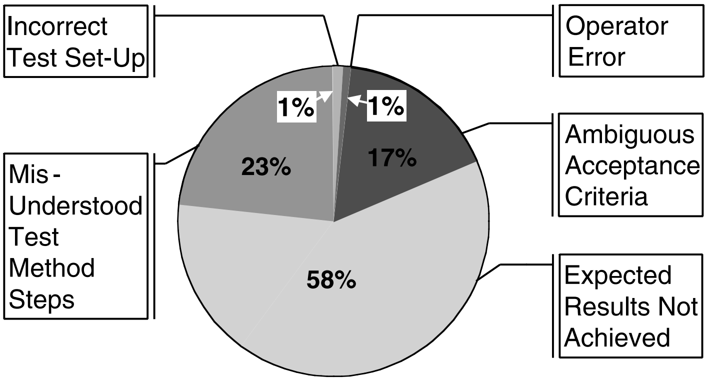


FIGURE 17.3 Test Failure Analysis.

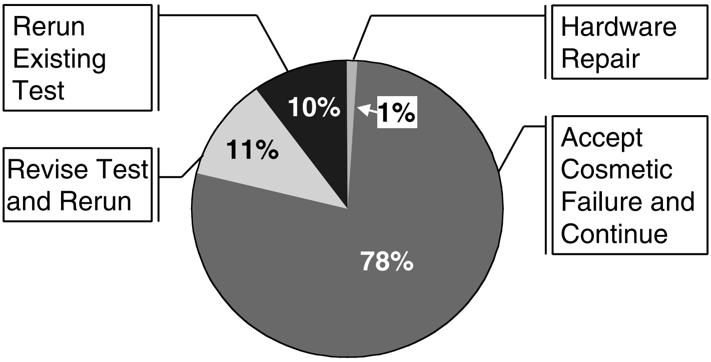


FIGURE 17.4 Test Failure Action.

Major amendments to documentation was required in order to address 18% of the errors identified; the rest only required minor document change. Remember, not all changes are limited to a single document. All document changes, however, need to go through change control, which in practical terms means rework and delays.

The follow-up actions to test failures are analyzed in Figure 17.4. The vast majority of test failures (78%) were accepted as cosmetic with no further action. The test case required revision and reissue so that the test could be repeated for 11% of the test failures. For a further 10% of test failures the test case was deemed acceptable but incorrectly executed. These tests could be rerun without modification once the tester understood where the test was misapplied. Finally, 1% of tests prompted hardware repair and a repeat test. The data collected highlight the need to train test staff to execute tests right the first time and also to quickly recognize when a test failure is cosmetic so that testing can progress without undue interruption to overall test execution.

User Qualification Metrics

The division of effort put into User Qualification is shown in Figure 17.5. About one third of the total effort is used to prepare test cases. It is important that test cases are clear and cover all the requirements of the computer system. Test execution and collation of testing evidence including preparing test reports accounts for over half of the User Qualification effort. User Qualification, however, often uncovers issues with specification and design documentation. Correcting specification and design deficiencies typically accounts for about 15% of the effort put into User Qualification. Corrective activity higher than this indicates poor development. Corrections to specifications and design documentation must not be ignored as it undermines validation.

The effort required to conduct an Installation Qualification broadly increases in a linear fashion with the size of a computer system. The effort required to conduct an Operational Qualification, meanwhile, tends to increase exponentially compared to the complexity of the computer system. The effort to conduct a Performance Qualification, like IQ, tends to increase in a linear fashion compared to the size of a computer system.

Understanding Contributory Factors

Specific factors that contribute to the overall increased effort on computer validation projects include more comprehensive procedures and training, higher level of detail in documentation, increased testing, and more rigorous document control (see Figure 17.6).

Additional documentation and testing are the primary factors that make validation more expensive than conventional quality practices. Extra "quality assurance" approvals add effort and sometimes the perception of bottlenecks to the validation process. Additional "quality and compliance"

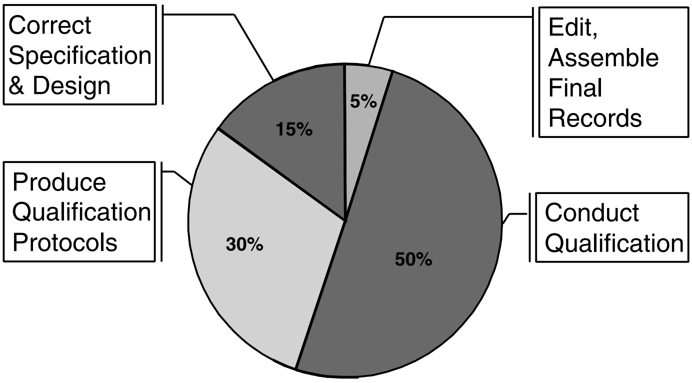


FIGURE 17.5 Split in Qualification Effort.

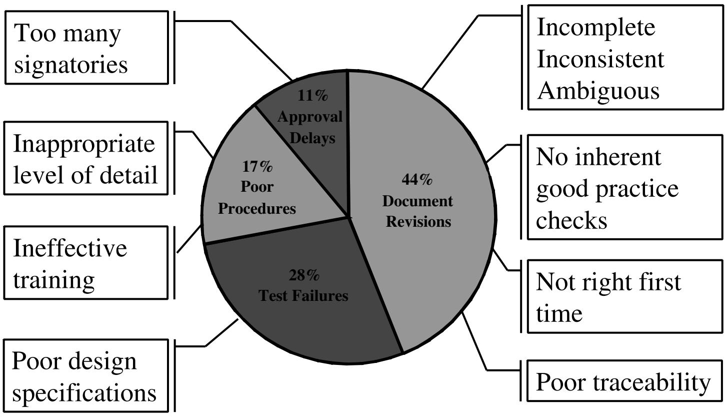


FIGURE 17.6 Factors Adding Effort to Validation.

approval signatories are often a result of various departments not agreeing on responsibilities and duplicating effort rather than being a regulatory requirement. The regulatory requirement for approvals is minimal. The same basic principle of overengineering validation contributes to the additional procedural controls associated with validation. Controls do need to be robust, but complexity is often added as a result of departmental politics and matrix organizational responsibilities rather than regulatory requirements.

Another important factor to appreciate is the impact of late change during computer system projects. It is generally understood that during computer system implementation late changes can have a very high impact compared to making modifications early. The relative impact of change during a project and operation of small and large computer systems is summarized in Figure 17.7. Collected data from 130 projects support these observations.[6]

Rules of Thumb

- Typically, projects afford about 40%–20%–40% of effort expended to (1) system specification, design, and development; (2) coding, configuration, and build; and (3) development testing and user qualification.
- The combination of effective design reviews and source code reviews should reduce overall project costs by about 10% compared to projects not implementing these reviews. (This saving is achieved by detecting errors before testing.)

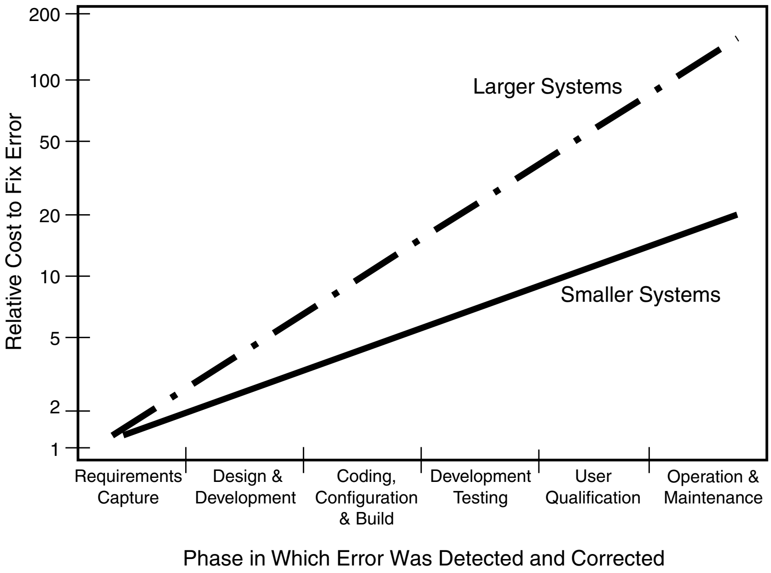


FIGURE 17.7 Relative Cost of Change.

- Increased project effort spent on system specification, design, and development should more than pay for itself in project pull-through.
- Typically 50% of design effort is expended during coding and testing either to clarify ambiguous specifications or to correct errors.
- Typically 20% of coding effort is expended during testing to correct errors.
- Typically 75% of errors are associated with 25% of the software.
- System testing typically only exercises 55% of errors without tracing tests to system requirements. With traceability to system requirements, up to 80% of errors may be challenged during system testing.
- Experience suggests that more than 10% of defects remain undetected at the point when the system is authorized for use.

OPERATION AND MAINTENANCE METRICS

Software maintenance is not limited to the correction of errors. Maintenance activities cover corrective maintenance, adaptive maintenance, perfective maintenance, and preventative maintenance.

- *Corrective maintenance* deals with the repair of errors.
- *Adaptive maintenance* deals with adapting software to changes in the operating environment, such as new hardware or the next release of an operating system. Adaptive maintenance does not lead to changes in system functionality.
- *Perfective maintenance* mainly deals with accommodating new or changed user requirements. It concerns functional elements of the computer system. Perfective maintenance also includes activities to increase the system's performance or to enhance the user interface.
- *Preventative maintenance* concerns activities aimed at increasing the system's maintainability, such as updating documentation, adding comments, and improving the modular structure (architecture) of the computer system.

It is worthwhile noting that the IEEE combines adaptive and perfective maintenance activities under the title of adaptive maintenance. Data have been published that suggest that half the maintenance effort involves correcting errors and half involves modifying the user to meet changing user needs including dealing with upgrades.[7] In reality the amount of effort directed at the latter will depend entirely on the organization's investment strategy and architecture philosophy. For this reason, it is only possible to make meaningful metric observations on those maintenance activities focused on correcting errors.

Corrective Maintenance Metrics

The annual corrective maintenance costs from approximately 250 computer applications are surprisingly consistent.[8] Not surprisingly, the maintenance effort decreased for older applications on the basis that an increasing proportion of errors is corrected over time. Annual corrective maintenance costs would seem as a rule of thumb to decrease by about one sixth every year (see Figure 17.8). The initial corrective maintenance costs were more dependent on the size of the application than on initial error rate. This is because fewer but bigger errors tend to be addressed in the early years of operation. These maintenance figures assume there are no other user-driven enhancements or system platform upgrades, etc.

When an error is to be corrected, the time to implement a change can vary enormously depending on the nature and scope of the change. The change control process should not unduly waylay changes. Ineffective change control can delay changes by many weeks or months not because of the complexity of analyzing the proposed change and assessing its wider impact on the existing computer system but because of an inability to process the paperwork in a timely fashion. The performance of the change control process can typically be greatly improved by:

- Instituting a rapid initial appraisal of change requests to filter out rejected changes
- Ensuring the change management process has no bottlenecks
- Automating the change control process with electronic review and approvals

Research has also been conducted into the so-called software death cycle. It has been suggested that in some cases up to one in three changes introduces a new error. A more typical metric might

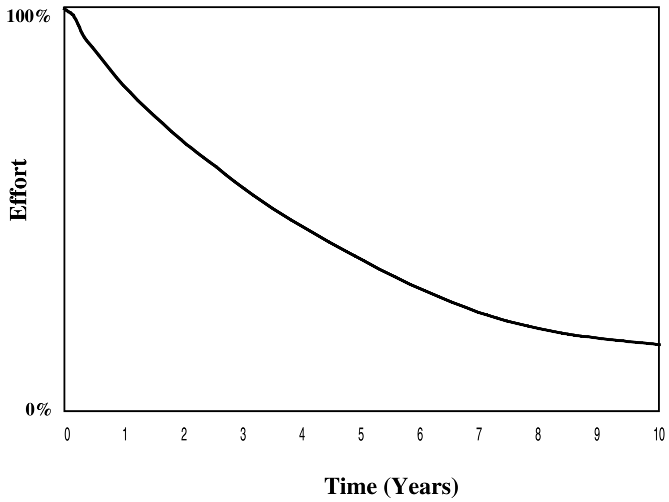


FIGURE 17.8 Corrective Maintenance Costs.

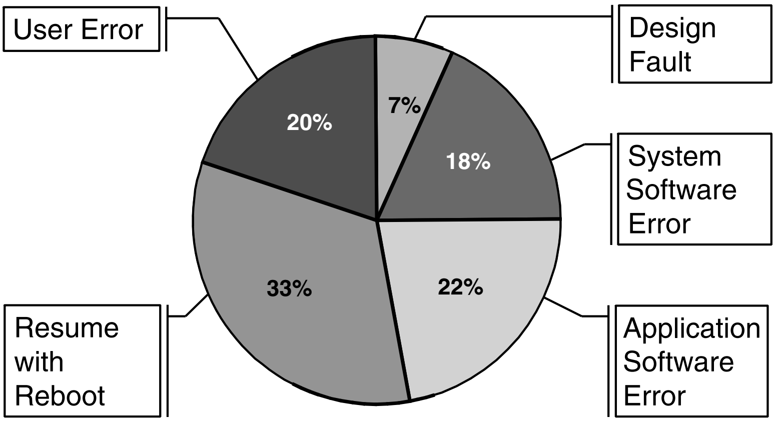


FIGURE 17.9 Malfunction Diagnosis.

be one in five changes. There is a strong dependency on specification and design documentation. Poor documentation encourages maintenance staff to hack a solution, relying on their personal knowledge of the particular application to avoid introducing new errors. Trying to avoid an appropriate level of detail in specification and design documentation during projects is a false economy.

Dependability Metrics

Operational dependability is a vital element of GMP compliance. An insight into the operational problems experienced with computer systems is given by an analysis of a validation consultancy firm's database of over 350 computer system malfunctions experienced by a number of international chemical and pharmaceutical manufacturing companies in the 1990s.[3] The results are presented in Figure 17.9.

- Poor application design and programming errors accounted for 29% of malfunctions, indicating the importance of a supplier's project capability. Some of these problems were due to poor change control of the installed computer system by the pharmaceutical company and the lack of documentation provided by servicing engineers. It is all too easy for operations staff to make changes on quiet shifts and forget to record what they did; it then comes as a great surprise to the responsible managers that the documentation describing their system is out of date.
- The importance of conducting Supplier Audits for COTS software is highlighted by the 18% of malfunctions attributed to standard software.
- The importance of training system operators is demonstrated by the 20% of malfunctions attributed to human error. Companies must ensure that training is given with approved SOPs before operators are required to use the computer system.

Unfortunately, the remaining malfunctions could not be diagnosed because a simple reboot of the software resumed normal operation, and subsequent investigation could not identify any reasons for the malfunction.

The extra cost associated with validation also affects operation and maintenance. It has been suggested that conventional quality effort for operation and maintenance processes may be doubled. However, if validation has been successful, case study evidence suggests that the overall cost of operation and maintenance may be reduced by up to 75%.

A selection of operational lessons gathered from a book considering management issues for systems dependability is listed below.[9] While there are undoubtedly other lessons relevant to pharmaceutical and healthcare systems, these would seem to convey the key points of learning:

- Management should be commensurate with the criticality of the system.
- Ensure competency of operations staff as individuals and teams.
- Control access to the system, including keys and passwords.
- Control the use of system overrides.
- Communicate learning from incidents.
- Ensure essential records are kept and maintained.
- Monitor changes and maintenance to the system.
- Ensure manufacturer's recommended operating instructions are followed.
- Ensure appropriate national and international standards are adopted.
- Audit and follow up outstanding issues with suppliers and subcontractors.
- Ensure contingency plans are practical.
- Maintain a positive attitude among operations staff.
- Regularly audit systems to verify that their specifications are still current and that they perform as intended.

It is evident that organizations must be ever vigilant of their GxP computer systems and continually develop their management capability for validation.

Rules of Thumb

- Maintenance costs often exceed original project costs.
- Corrective maintenance costs typically reduce by about one sixth every year.
- Typically annual support costs average about 10% of original project cost charged annually, index linked to rate of inflation.
- As many as one in five changes introduces a new defect.
- Effective validation should reduce maintenance costs by about 75%.

PROCESS IMPROVEMENT

Many pharmaceutical and healthcare companies are now considering process improvements for their validation practices. Two main approaches are typically adopted based on the established process improvement methodologies commonly known as Lean Manufacturing and Six Sigma (Figure 17.10). Lean Manufacturing is aimed at removing redundant steps and wait time from processes. Six Sigma is aimed at reducing process variability. Both Lean Manufacturing and Six Sigma look at actual working practices rather than what is supposed to be happening. Together, Lean Manufacturing and Six Sigma (sometimes combined and referred to as Lean Sigma) offer powerful tools to improve business efficiency including computer validation.

Lean Validation

Validation processes that have not been subjected to a focused performance review typically offer fertile ground for improvement. The basic approach to leaning validation can be summarized in the following five key steps:

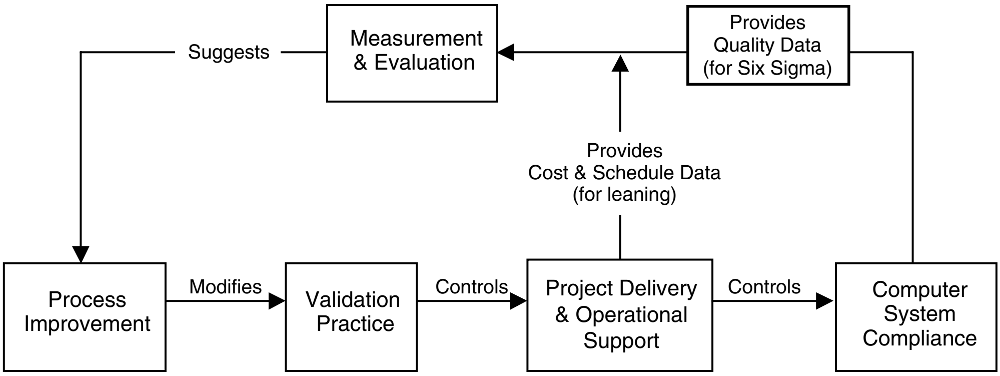

Suggests
Measurement & Evaluation
Provides Quality Data (for Six Sigma)
Provides Cost & Schedule Data (for leaning)
Process Improvement
Modifies
Validation Practice
Controls
Project Delivery & Operational Support
Controls
Computer System Compliance

FIGURE 17.10 Validation Process Improvement.

Define the Problem/Opportunity

- What are you trying to characterize?
- What are the scoping boundaries?
- What is the business case for validating?
- Who/what are the process suppliers, inputs, outputs, and customers?
- What process metrics are appropriate?

Baseline Current Way of Working

- What is my baseline?
- How should I collect data to baseline performance?
- What are the key equipment, process, and product parameters?
- How capable is the current process against what my customers require?
- How capable is the current process against what my suppliers require?
- What are the failure modes?

Analyze Opportunities

- What is the current process flow?
- What sources of variation are relevant?
- Cause and effect: what affects the key equipment, process, and product parameters?
- How can the process be systematically optimized?

Make Improvements

- What solutions help verify or improve the process?
- What are the costs, benefits, and risks associated with each solution?
- Do pilot runs confirm hypotheses?
- How best to implement improvements?

Realize Benefits

- Validate and document revised process.
- How to monitor revised process to preserve gains and maintain control?

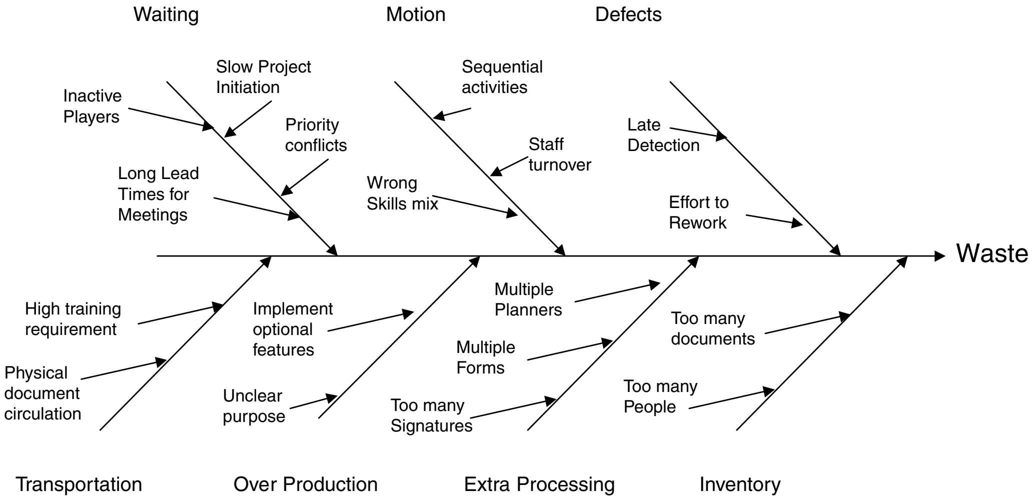


FIGURE 17.11 Validation Waste.

The "fishbone" diagram, as illustrated in Figure 17.11, can be used to structure the identification of numerous opportunities for removing waste in the validation process. Each will then have top be quantified and opportunities prioritized for implementation. There are seven basic types of waste:

- Overproduction — developing optional software features that are not critical or mandated, preparing unnecessary reports, unnecessary duplication of information between documents
- Waiting — staff unavailable when needed (meetings, reviews, and approvals), processing corrective actions monthly rather than straightaway, and delays to critical path
- Transportation — physical movement of people and documentation
- Inventory — too many documents, too many people, poor organization
- Extra Processing — conducting activities that are not necessary (e.g., too many signatories), maintaining documents that do not need to be kept current, rework to correct defects
- Motion — sequential activities that could be conducted in parallel, inability of staff to resolve issues referred to them without handoff to someone else
- Defects — data and document errors, miscommunication

Collecting data to analyze how validation personnel spend their time can provide very useful baseline information. Figure 17.12 shows an activity analysis for validation staff at two different sites. In this example less than half the available time of validation practitioners is spent actually preparing, reviewing, and approving validation documents. There would appear to be a lot of wasted time in fruitless meetings and chasing documents round their distribution for review and approval. Why is this? One reason might be that documents are being prematurely released before they are ready to hit critical path target dates in the project plan. Another reason might be that documents go through many revisions each with half a dozen or more signatories thus creating project delays. In theory there should be no need for revisions if the document is right the first time. The need for large numbers of signatories must also be challenged. Further investigation is required and corrective action taken.

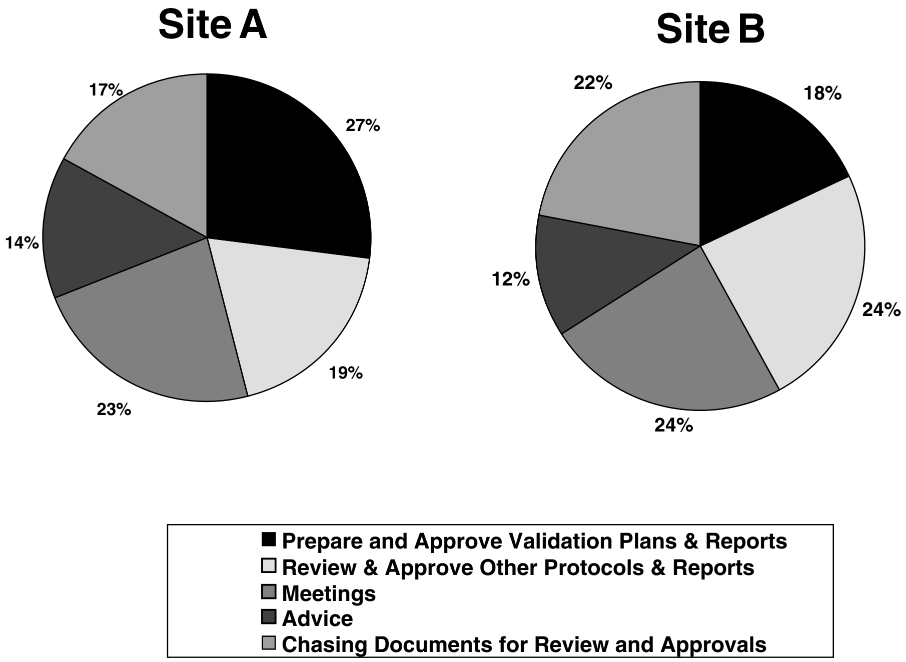


FIGURE 17.12 Example Validation Staff Activity Analysis.

Six Sigma Validation

The Six Sigma process can be used to benchmark the capability of a validation process and hence indicate the significance of any opportunity for improvement. Average capability is characterized by a Three Sigma performance. Six Sigma indicates a world class performance; anything beyond Six Sigma is not considered cost-effective.[10]

Some validation opportunities identified by pharmaceutical and healthcare companies in their software engineering processes include:

- Project start-up time
- Size of certain key documents
- Number of signatories on individual documents
- Document review and approval cycle times
- Clarity of requirements (checks on ambiguous words)
- Amount of evidence collected during testing
- Testing time for similar systems

Figure 17.13 presents a cost vs. compliance curve and is based on the compliance strategy discussion in Chapter 3. This graph takes Figure 3.1 a step further to illustrate the Six Sigma opportunity that pharmaceutical and healthcare manufacturers have to improve their level of compliance and reduce costs at the same time. The large dot on the Two Sigma plot is meant to represent point B from Figure 3.1, that is, the common-sense approach to ensuring sufficient but not too much validation is conducted to fulfill regulatory requirements and avoid major noncompliance. The same point is marked on the Six Sigma plot to illustrate how more capable processes can reduce validation costs.

Consider an example validation/quality process such as executing test cases. A project may run many hundreds of test cases. Defects observed could be based on all issues relating to ambiguous

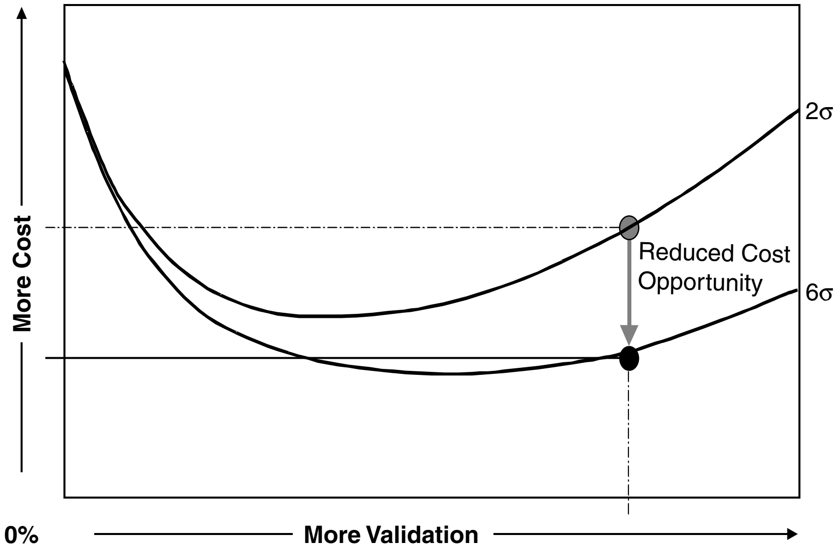


FIGURE 17.13 Six Sigma Improvement Opportunity.

test instructions and acceptance criteria. Test cases should not have residual problems; they should have been reviewed beforehand. Appendix Table 17.C1 can be used to approximate the Six Sigma capability for the validation process. Appendix Table 17.C3 is then used to indicate the cost of quality as a percentage of the cost of ownership. The cost of quality includes the cost of failure (scrap, rework), cost of appraisals (self-inspections, regulatory inspections, and supplier audits), and cost of prevention (validation procedures, validation planning, and training).

To demonstrate how the Six Sigma capability can be calculated using Appendix Table 17C.1 let us assume we have 120 test cases of which 15 have ambiguities that are not discovered until test execution. The yield of correct test cases is therefore 0.87. Two critical to quality (CTQ) characteristics have been discussed above in relation to case studies (ambiguous test instructions and ambiguous acceptance criteria), that is, $N = 2$. Assuming the CTQ characteristics are evenly split, the defect rate per CTQ characteristic is $[(1 - 0.87)/2] = 0.065$, and consequently the defects per million opportunities (DPMO) is 65,000. This equates to an approximate Six Sigma value of 3 using Appendix Table 17C.2. Now examine Appendix Table 17C.3, which indicates that subjecting overall validation to the same sigma level will result in the cost of quality of about 25 to 40% of the cost of ownership of the computer system. This is similar to some of the anecdotal examples given for the cost of quality in Chapter 1. There would seem to be plenty of opportunity for improvement.

Another example might be to improve the review and approval process. Some pharmaceutical companies have successfully reduced cycle times by an order or magnitude. The breakthrough is usually made when the team looking at process improvement analyzes real data on how its processes are operating and sees what is actually happening in practice. A realistic cycle time targeted for improvement should be based on actual current practice. In this case as a result of Six Sigma improvements project managers should be able to better anticipate and schedule document review and approvals.

Plotting the distribution of activities is another useful way to illustrate the starting situation and the impact of any process improvement. Figure 17.14 provides an example of how a cycle time for the general review and approval of documents might be drawn. The review and approval cycle times seem excessive. Project critical paths are likely to bottleneck in these circumstances. Unnecessary complexity and effort are probably being added to projects to manage late approvals. Figure 17.15 shows how a control chart can be used to measure existing practice for particular document types, and how target improvements might be set.

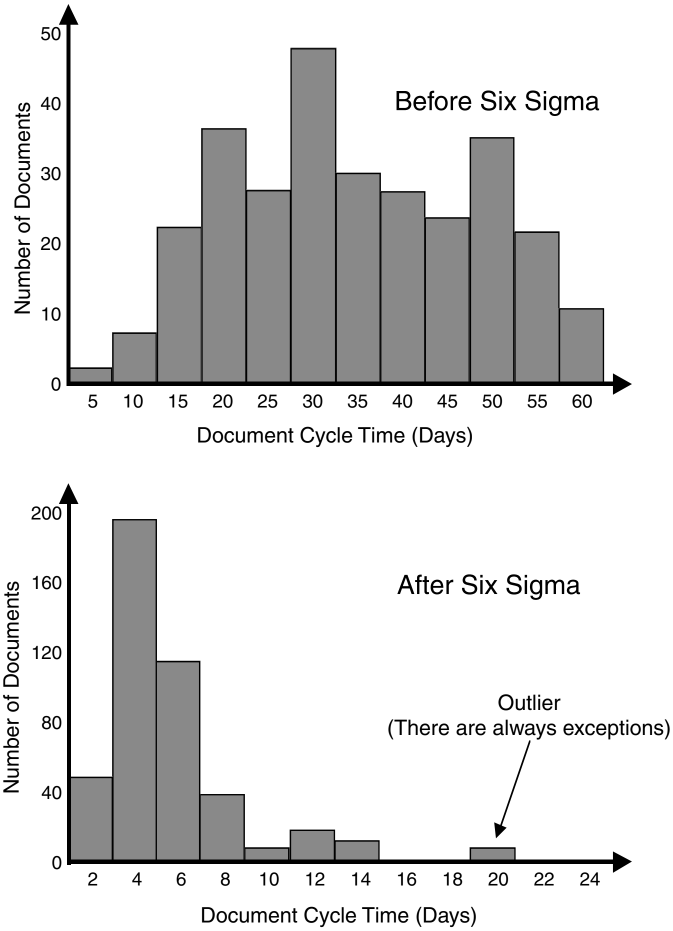


FIGURE 17.14 Example Document Cycle Time Distributions.

The improvement to be made could be to institute a weekly approval meeting for documents. Documents need to be circulated in advance of the meeting (a minimum advance circulation time should be set). Attendees at the meeting must review documents before the meeting. Any revisions to documents need to be agreed upon in the meeting and changes made directly to documents so that documents can be concluded (signed) at the meeting. Nominated attendees must assign deputies authorized to approve documents on their behalf when they cannot attend meetings themselves. This process requires a lot of self-discipline. Of course, just letting project managers know that document review and approval cycle times are being monitored may be enough in itself to prompt improvements.

Best Practice Expectations

Many pharmaceutical and healthcare companies have challenged the typical current costs associated with validation projects (refer to Figure 12.2) even though there is a payback to this investment. As discussed, Lean Manufacturing and Six Sigma offer an opportunity to reduce costs while maintaining or even improving the robustness of the validation process. Few formal case studies have been published; however, recent experience from a large multinational pharmaceutical company suggests that the typical cost of validation on IT projects can be reduced by in excess of 65%. Similar experience with control systems suggests that there is less opportunity to reduce cost

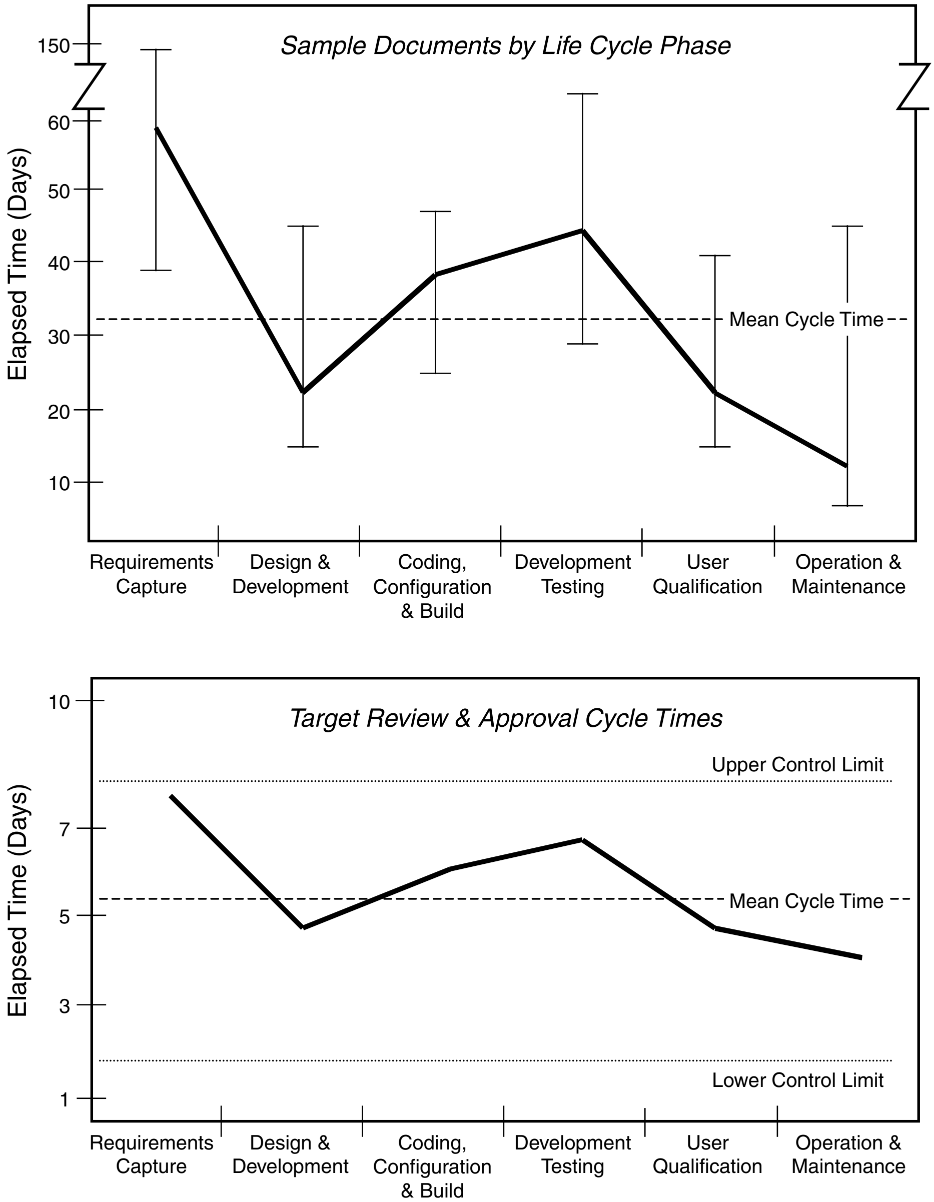


FIGURE 17.15 Example Six Sigma Control Charts.

but that costs can still be reduced by about 30%. Applying these cost reductions to Figure 12.2 provokes the need for a step change in current industry validation practices in terms of efficiency and cost-effectiveness. It should be possible to achieve the best practice quoted at various conferences that validation should account for no more than 10% of project costs for all computer system types. Pharmaceutical and healthcare companies should take seriously the benefits that focused Lean Manufacturing and Six Sigma initiatives offer. Dramatic performance improvements will only be made, however, where everybody involved in validation work together as equal parties to integrate, streamline, and optimize the validation process. It is up to industry to rise to the challenge of taking validation to the next level of maturity.

REFERENCES

1. Paulk, M.C. (1995), The Evolution of SEI's Capability Maturity Model for Software, *Software Processes — Improvement and Practice*, 3–15.
2. Royce, W. (1998), *Software Project Management: A Unified Framework*, Addison-Wesley, Reading, MA.
3. Wingate, G.A.S. (1997), *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice*, Interpharm Press, Buffalo Grove, IL.
4. Herremans, P. (2000), *Manage System Development Capabilities*, Institute of Validation Technology Conference on Computer & Software Validation, London, February 21 and 22.
5. Murtagh, R. (2002), *Identifying Improvements to Current Industry Practice for the Validation of Automated Tablet Compression Machines*, M.Sc. Dissertation, University of Manchester Institute of Science and Technology, Manchester, U.K.
6. Grady, R.B. (1992), *Practical Software Metrics for Project Management and Process Improvement,* Hewlett-Packard Professional Books, Englewood Cliffs, NJ.
7. Vliet, H.V. (2000), *Software Engineering: Principles and Practice*, Second Edition, John Wiley & Sons, New York.
8. Maxwell, K.D. (2002), *Applied Statistics for Software Managers*, Software Quality Institute Series, Prentice Hall, Upper Saddle River, NJ.
9. Redmill, F. and Dale, C. (1997), *Life Cycle Management for Dependability*, Springer-Verlag, New York.
10. Harry, M. and Schroeder, R. (2000), *Six Sigma*, Doubleday Press, Garden City, NY.
11. McDowall, R. (2000), *Validation of Chromatography Data Systems*, Henry Stuart Conference Studies Conference, London, September 13.
12. Wingate, G.A.S. (1997), *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice*, Interpharm Press, Buffalo Grove, IL.
13. GAMP Forum (2000), Industry Board Meeting, Antwerp, Belgium, July.
14. Samways, K. (1997), *Validating an MRP II System*, Validating Automated Manufacturing and Laboratory Applications (Ed. Guy Wingate), Interpharm Press, Buffalo Grove, IL.
15. Clark, C. (2001), *Validation of MRP II Systems*, Business Intelligence Conference on Computer System validation for cGMPs in Pharmaceuticals, London, March.
16. Cleave, R. (2001), *Cost Effective Validation of LIMS*, Business Intelligence Conference on Computer System validation for cGMPs in Pharmaceuticals, London, March.
17. Sephar, R. (2002), *Laboratory Case Study: Validation of LIMS*, Institute of Validation Technology's Computer and Software Validation Conference, Tokyo, February 18 and 19.
18. Accenture (2001), *21 CFR Part 11: Achieving Business Benefits*, Pharmaceuticals and Medical Products White Paper, June.
19. Perez, R. (2001), *Applying GAMP 4 Concepts to Determining Validation Strategy for an IT System*, ISPE GAMP 4 Launch Conference, Amsterdam, December 3 and 4.
20. Fiorito, A. (2002), *Qualifying and Managing Workstation Arrangements*, Institute of Validation Technology's Network Infratructure Qualification and Software Validation Conference, Philadelphia, October 8 and 9.
21. Selby, D. (2000), David Begg Associates Training Course on Computers and Automated Systems: Quality and GMP Compliance, York, U.K., July 3–7.
22. Wyrick, M.L. (2000), *Assessing Progress Towards Harmonisation of Validation Governance in the Global Pharmaceutical Industry*, Developing a Business Driven Approach to Computer System Validation for cGMP in Pharmaceuticals, Business Intelligence, London, March 29 and 30.

APPENDIX 17A VALIDATION CAPABILITY QUESTIONNAIRE[3]

Level 2 Questions

- Does the project follow a formally documented project planning process?
- Are estimates of cost and scheduling (including intermediate milestones) documented for use in planning and tracking project progress?
- Do project plans identify work packages and responsibilities for their delivery?
- Do all affected groups and individuals agree on their responsibilities?
- Are adequate resources and time provided for project planning?
- Does the project manager review planning both on a periodic and event-driven basis?
- Is the actual project performance (e.g., cost and schedule) compared with original plans, and are corrective actions taken when they differ?
- Do all affected group and individuals agree to any change in their responsibilities?
- Is someone on the project specifically tasked with tracking and reporting progress?
- Are measurements used to determine the status of activities and deliverable on the project?
- Are project tracking activities and results periodically reviewed with senior management?
- When changes occur, are the necessary amendments made to project plans?
- Do projects follow project and quality management policy requirements?
- Are project team members trained in the procedures they are expected to use?
- Is progress on project deliverables subjected to periodic review?
- Is a documented procedure used for selecting suppliers and subcontractors?
- Are changes to subcontractors notified to the pharmaceutical or healthcare company?
- Are periodic technical interchanges held with subcontractors?
- Are performance issues followed up with suppliers and subcontractors?
- Does the project manager review supplier and subcontractor performance on both a periodic and event-driven basis?
- Is a defined quality management system used on projects?
- Do quality plans identify quality assurance activities and deliverables?
- Are internal audit results provided to affected parties?
- Are software quality assurance issues not resolved by the project addressed by senior management?
- Are adequate resources and time provided for quality assurance activities?
- Are measurements used to determine the cost and schedule status of quality assurance activities?
- Are quality assurance activities reviewed with senior management on a periodic basis?

Level 3 Questions

- Does the organization follow a written policy for both the development and maintenance of computer systems?
- Does the organization have a documented and maintained quality management system?
- Does the organization collect, review, and make available performance data related to the use of the quality management system?
- Do users of the quality management system receive adequate training?
- Is the review and maintenance of the quality management system planned, monitored, and audited?
- Is there a training policy?

- Are training requirements planned covering both management and technical skills?
- Are adequate resources put into training?
- Are measurements used to determine the quality of training?
- Is training reviewed with senior management on a periodic basis?
- Are projects planned in accordance with the quality management system?
- Are project activities and deliverables reviewed and audited by quality assurance personnel?
- Is consistency maintained across different projects?
- Is there a written policy that guides the establishment and management of multidisciplined teams?
- Do internal groups work together in collaboration?
- Are inter-group issues identified, tracked, and resolved?
- Are measurements used to determine the status of inter-group coordination activities?
- Are inter-group relationships reviewed with senior management on a periodic and event-driven basis?
- Is effort focused on critical aspects of the computer system development and maintenance?
- Is the change control process defined, proceduralized, and robust?
- Do personnel understand and receive training to enable them to discharge their change control responsibilities?
- Are the volume and nature of changes measured and monitored?
- Is there a mechanism for verifying that the originator of a change request is satisfied by the change implementation?

Level 4 Questions

- Is there a written policy for quantitatively measuring management of development and maintenance of computer systems?
- Is there a defined quantitative measurement process?
- Is the management performance of development and maintenance of computer systems controlled quantitatively?
- Are adequate resources provided for quantitative measurement process activities?
- Are quantitative measurements reviewed with senior management on a periodic and event-driven basis?
- Are documented correlations made between historical management and actuals?
- Are historical management and actuals used to improve planning on current projects?
- Do projects use measurable and prioritized goals for managing quality?
- Are measurements used to determine the status of activities for managing quality (e.g., the cost of poor quality)?
- Are the activities for managing quality planned in advance for projects?
- Are the activities performed for quality management reviewed with senior management on a periodic basis?
- Is return on investment evaluated, monitored, and reported to senior management?

Level 5 Questions

- Are defect prevention activities planned?
- Is there a formal process to identify common cause defects?
- Once identified, are common causes of defects prioritized and systematically eliminated?
- Is training given in defect prevention?
- Are defect prevention activities subject to quality review and audit?
- Does the organization follow a defined process to management technology changes?

- Are new technologies evaluated to determine their effect on quality and productivity?
- Does senior management sponsor the introduction of new technology?
- Do people throughout the organization participate in process improvement initiatives?
- Are improvements continually made to process management?
- Are process improvement initiatives reviewed with senior management on a periodic basis?

APPENDIX 17B
REFERENCES FOR COST OF VALIDATION METRICS

Type of System	Source of Information	Validation (% of Project Effort)	Reference
	Analytical Laboratory Systems		
Standard COTS Laboratory Systems	Rule of Thumb	5%	
Chromatography Data Systems	Best Practice	8–10%	McDowall[11]
	Control Systems		
Process Control Systems (e.g., PLC)	Best Practice Case Study	5–8%	Wingate et al.[12] Murtagh[5]
Configured SCADA System	Case Study	10%	
Distributed Control System	Workshop	20%+	ISPE Meeting[13]
	Management Systems		
Basic MRP II System	Case Studies	8%[c]–10%	Samways[14] Clark[15]
LIMS Systems	Case Studies	10%[e] –15%	Cleave,[16] Sephar[17]
IT Systems	Rule of Thumb	15%	Accenture[18]
Integrated ERP System	Case Studies	15–20%+	Clark[15] Perez[19]
	IT Infrastructure		
Computer Network Infrastructure	Best Practice Rule of Thumb	20%	Fiorito[20]

Note: Based on information from Accenture, AstraZeneca, Aventis, Boots, GlaxoSmithKline, ICI, ISPE/GAMP, Jansen, Napp, Novartis, PDA, and Roche. Overall best validation practices have been reported 5–10% project costs.[9,27]

APPENDIX 17C
SIX SIGMA TOOL BOX

TABLE 17C.1
Six Sigma Capability

Step	Action	Equation
1	Identify a validation/quality process.	Not applicable
2	How many times was the process run?	Not applicable
3	How many process runs did not exhibit defects?	Not applicable
4	Calculate yield.	(Step 3)/(Step 2)
5	Calculate the defect rate from Step 4.	1 − (Step 4)
6	Determine the number of things that could potentially cause the observed defects.	N = Number of critical to quality (CTQ) characteristics
7	Calculate the defect rate per CTQ characteristic.	(Step 5)/(Step 6)
8	Calculate the defects per million opportunities (DPMO).	(Step 7) × 1,000,000
9	Convert the DPMO into a Six Sigma value using Appendix Table 17C.2.	Not applicable
10	Draw conclusions using Appendix Table 17C.3.	Not applicable

TABLE 17C.2
Six Sigma Conversion Table

Sigma Value	Defects per Million Opportunities (DPMO)	Sigma Value	Defects per Million Opportunities (DPMO)
0.0	933,193	3.2	44,565
0.2	903,199	3.4	28,717
0.4	864,334	3.6	17,865
0.6	815,940	3.8	10,724
0.8	758,036	4.0	6,210
1.0	691,462	4.2	3,467
1.2	617,911	4.4	1,866
1.4	539,828	4.6	968
1.6	460,172	4.8	483
1.8	382,088	5.0	233
2.0	308,537	5.2	108
2.2	241,964	5.4	48
2.4	184,060	5.6	21
2.6	135,666	5.8	9
2.8	96,800	6.0	3
3.0	66,807		

Note: This table includes 1.5 Sigma shift.

TABLE 17C.3
Cost of Quality

Sigma Level	Defects per Million Opportunities	Cost of Quality
2	308,537 (Uncompetitive)	>40% of cost of ownership
3	66,807	25–40% of cost of ownership
4	6,210 (Industry Average)	15–25% of cost of ownership
5	233	5–15% of cost of ownership
6	3.4 (World Class)	<1% of cost of ownership

18 Concluding Remarks

CONTENTS

Validation practices for computer systems have been presented in line with current expectations of GxP regulatory authorities and industry practice. This chapter concludes the first part of this book by reviewing some fundamental concepts and industry trends that will affect how we validate in the 21st century. Specific guidance on the validation of different types of computer system can be found in the case studies presented in Chapters 19 through 42 of this book. The authors of these case studies are themselves experienced practitioners, and they have been encouraged to focus their papers on the key issues affecting their case studies.

COMPUTING ENVIRONMENT

Today's computing environment is illustrated in Figure 18.1. Four tiers of computer systems are considered:

1. Measurement and Control — providing data acquisition and actuator directives
2. Process Management — providing process monitoring and control
3. Operations Management — providing management control across multiple processes
4. Enterprise Management — providing management control across multiple processes at multiple locations

The computer systems used within each of these levels tend to have common characteristics that will influence their validation. For instance, measurement and control systems are generally configurable Commercial Off-The-Shelf (COTS) instruments. Examples include control instrumentation, analytical instrumentation, and medical devices including blood processing systems.

Process management systems include real-time control systems, spreadsheets, and databases. They often include bespoke programming although they may be based on COTS products. Examples of process management systems include Programmable Logic Controllers (PLC), Supervisory

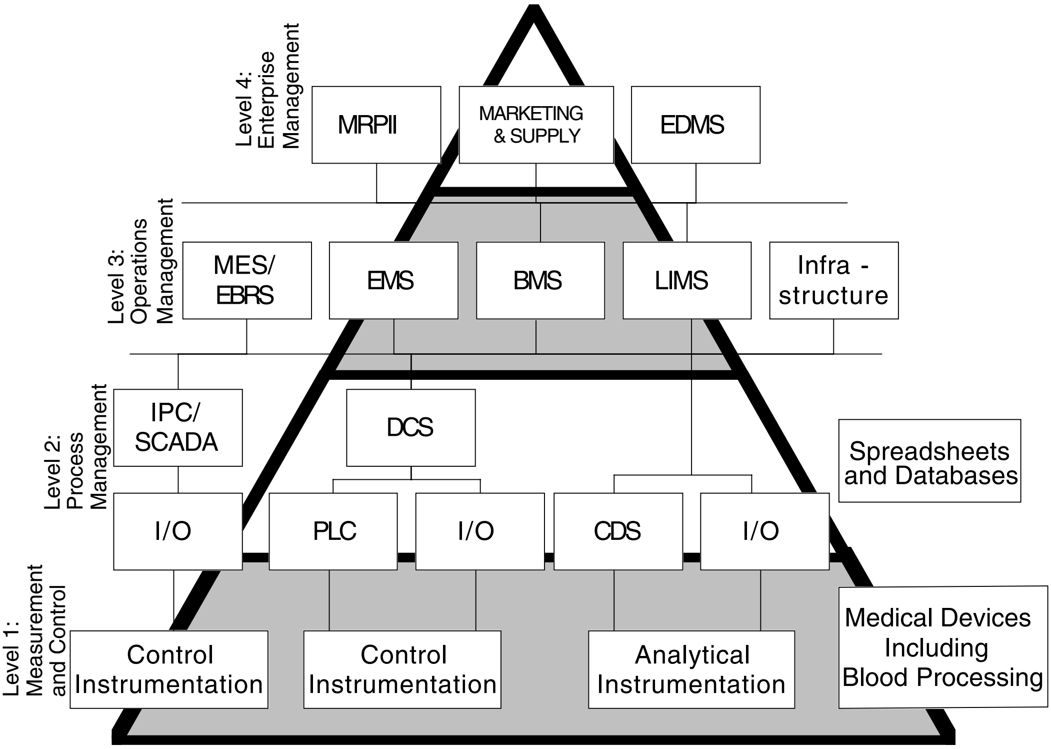

Level 4: Enterprise Management
MRPII
MARKETING & SUPPLY
EDMS
Level 3: Operations Management
MES/ EBRS
EMS
BMS
LIMS
Infra - structure
Level 2: Process Management
IPC/ SCADA
DCS
Spreadsheets and Databases
I/O
PLC
I/O
CDS
I/O
Level 1: Measurement and Control
Control Instrumentation
Control Instrumentation
Analytical Instrumentation
Medical Devices Including Blood Processing

FIGURE 18.1 Computer System Hierarchy.

Control And Data Acquisition (SCADA), Industrial PC systems, and Distributed Control Systems (DCS). Measurement and control systems and process management systems are often integrated together. Process Analytical Technology (PAT) is a special case implementing nonintrusive process instrumentation.

Operations management systems include Manufacturing Execution Systems (MES), Laboratory Information Management Systems (LIMS), Building Management Systems (BMS), and Engineering Management Systems (EMS). These systems are not necessarily dedicated to a particular plant or laboratory and so may be shared, for instance, between several manufacturing units on a site. While operations management systems may have response-time dependencies, these are generally much less stringent than the real-time operations required by process and measurement and control systems. Computer Network Infrastructure plays a very important role is supporting these as well as enterprise systems.

Enterprise management systems include Manufacturing Resource Planning (MRP II) systems, commercial Marketing and Supply applications, and Electronic Document Management Systems (EDMS), and are typically based on large IT packages. These packages often claim to be configurable, although such configuration may be extensive and almost akin to bespoke programming. Enterprise management systems are often implemented to coordinate operations across multiple sites, possibly in different countries.

THE BUSINESS CASE FOR VALIDATION

A key influence in today's business environment is the return on investment for computer systems and the cost of ownership for what are often expensive assets. Automation strategies need to exploit the following benefits that computer systems offer:

- More flexibility
- Higher efficiency
- Faster operations
- Improved consistency
- Less human error
- Real-time performance data

All this means nothing if either the system does not operate correctly or if it has insufficient validation. The cost of noncompliance can be huge, as discussed in Chapter 1. The challenge is to determine how much validation is enough. Excessive validation may increase confidence in regulatory compliance, but it is expensive and will not necessarily bring any further assurance in the process being validated.

INDUSTRY CONSENSUS

An established industry approach has emerged providing the basis for the international acceptance of validation work by various GxP regulatory authorities. The basis for consensus has been the GAMP Guide. The benefits of a widely adopted industry framework for validating computer have been extolled by Anthony Trill of the MHRA and include:[1]

- Stabilizing standards and their interpretation at a realistic level
- Linking validation to existing standards (e.g., ISO 9000)
- Reducing the costs of validation
- Shortening the time required for validation
- Ensuring appropriate documentation is produced for validation projects
- Improving the quality and reliability of delivered systems

All this makes good business sense and should eliminate the need for corrective work because of misunderstood validation requirements. Similar harmonization would appear to be occurring on the topics of electronic records and electronic signatures. Mutual Recognition Agreements (MRAs) between various national regulatory authorities such as the FDA, MHRA, TGA, and MHLW and the work of the International Conference for Harmonization (ICH) offer an opportunity to formally consolidate harmonized computer validation requirements.

GOLDEN RULES REMAIN UNCHANGED

Many light-hearted publications have published lists of golden rules for validation practitioners. None of these lists, however, is carved in stone, and there is some flexibility in deciding what should be included in the way of guidance. The checklist given below was published in *Validating Automated Manufacturing and Laboratory Applications*,[2] and essentially remains unchanged. It should help practitioners concentrate on ten key validation issues:

- Plan and monitor validation — adopt a proactive project management style.
- Use competent personnel, and train where necessary.
- Document validation including collating raw data as supporting evidence — ensure everything is reviewed and approved.
- Implement a regime of change control covering projects and operational use of the computer system.
- Specify procedures for validation and follow them.
- Develop system specifications with testing in mind and test using preapproved qualification protocols.

- Use the approval of summary validation reports to authorize the use of a computer system.
- Operate and maintain validated computer system in a state of control.
- Periodically review the validation status of computer systems and initiate revalidation where necessary.
- Archive validation evidence for future retrieval.

Remember that the GxP requirements for computer data are the same as for manufacturing records and documentation. Similarly, computer hardware (operating platforms and supporting networks) should fulfill the GxP requirements of manufacturing equipment.

RISK MANAGEMENT

The FDA has recently highlighted the importance of risk management as part of 21st century compliance.[3] Other regulatory authorities such as MHRA share this perspective. Without risk management, computer validation costs can quickly become prohibitive. Taking the highest level of compliance for all aspects of a computer system will not necessarily lead to discernible, increased patient/consumer safety.

The ISPE has published two important concept papers for functional risk management and electronic record/signature risk management.[4,5] These papers mark the start of what is likely to become a new era in applying risk management as an integral part of computer system validation.

This book has positioned the role of risk management throughout the validation life cycle and in the controls required for electronic records and electronic signatures. Design for risk management has also been discussed in terms of using backup systems, independent monitoring systems, and segregating regulated and nonregulated aspects for validation within systems.

KEY ROLE OF SUPPLIERS

There has been an increasing use of COTS software packages because of improved availability and the advantages offered over custom (bespoke) software. The advantages and disadvantages of COTS software packages and custom (bespoke) software are summarized in Table 18.1. Computer systems are also being integrated into ever more complex and sophisticated network architecture making it harder to segregate GxP from non-GxP elements. Validation requires a holistic approach so that key aspects are not inadvertently missed.

The key role suppliers have in validation is widely acknowledged. To avoid duplicating tasks in whole or in part, pharmaceutical and healthcare companies and suppliers should work together in partnership. They must be able to work efficiently as a combined team and, as such, must be able to communicate effectively to streamline validation. The mutual benefits of cooperation between customers and suppliers are outlined in Table 18.2.

ORGANIZATIONAL CHANGE

Senior management must give and hold to a clear corporate vision for compliance. Validation principles need to be incorporated into the culture of an organization. Those tasked to champion the compliance cause must believe in its intrinsic value to the business. Without sustained management backing for computer validation, skills will remain in the domain of consultants instead of being instilled as another competency of their permanent employees.

Outsourcing tends to be encouraged by many pharmaceutical and healthcare companies to reduce the number of full-time employees in a company. The Validation Department is also included in this because it uses individual contractors or managed supplier project capabilities. There is nothing wrong with outsourcing, but the pharmaceutical or healthcare company must maintain a

TABLE 18.1
Comparing COTS Software Packages and Custom (Bespoke) Applications

Approach	Advantages	Disadvantages
Custom (Bespoke) Software	Complete change freedom	Expensive, unpredictable developments
	Smaller, often simpler implementations	Typically late delivery, over budget, and reduced functionality
	Often better performance	Single platform dependency
	Control of development and enhancement	Often immature and fragile with an undefined maintenance model
	Clear understanding of user requirements	Drain on expert resources
COTS Software Packages	Predictable license costs	Up-front license fees
	Broadly used, mature technology	Multiple-supplier incompatibilities, integration is not always trivial
	Available now	Frequent upgrades
	Dedicated support organization	Dependency on supplier, limited warranties
	Hardware/software independence	Run-time efficiency sacrifices
	Rich in functionality	Functionality constraints and/or unnecessary features that consume extra resources

TABLE 18.2
Benefits of Cooperation between Pharmaceutical Manufacturers and Suppliers

Pharmaceutical and Healthcare Company	Supplier
Meet user needs	Satisfy customer
Be easier to set up	Be handed over sooner
Be in production sooner	Be paid sooner
Break down less often	Fewer warranty visits
Be easier to repair	Shorter warranty visits
Be easier to further develop	Be easier to modify and/or upgrade
Be used more effectively	Good reference sites for new customers
Cheaper overall	Cheaper overall
Preferred supplier	Repeat business

Source: Wingate, G.A.S. (1997), *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice*, Interpharm Press, Buffalo Grove, IL.

critical mass in its internal capability to effectively manage adopted validation standards. The cost of getting the balance wrong is burnout and resignation of key staff. Computer validation experts are already at a market premium and this situation is unlikely to change. Organizations must manage to retain and develop key staff for computer systems validation.

Organizational structures must not hinder exploitation of new technology. In particular, the Validation Department in many pharmaceutical and healthcare manufacturing organizations has a reputation for slowing down or even preventing the implementation of new technology. Validation must not be seen as a constraint and should not be managed as such. Pragmatic solutions should be found to enable the implementation of new technology while ensuring compliance, perhaps by refining validation approaches. Validation Departments are not known for their inspiration, but there is room for the development of novel approaches! Perhaps in the next few years we will see the emergence of new ways of validating, in particular, corporate computer systems, which tend to have their own characteristics and needs compared to process control systems and laboratory analytical equipment.

PUTTING EFFORT WHERE IT IS NEEDED

From a technical perspective, with the possible exception of some expert systems, all computer systems can be validated. Validation is about building confidence in the correct operation of a system. The degree of confidence that can be built for different systems may vary, but a case can always be developed. Beware of using the term *unvalidatable.* It is far more likely that cost is prohibiting validation rather than technicalities, and there are usually alternative validation approaches that can assist in making the exercise more cost-effective. Whatever the cost, GxP regulatory authorities will not excuse pharmaceutical and healthcare companies from using computer systems without necessary validation.

So, how much validation is insufficient, enough, or too much? Pharmaceutical and healthcare companies and suppliers alike must resist rushing into validating their computer systems without carefully considering the implications of consistently implementing their validation practice across all their systems. *Too* little validation may not necessarily be identified on a first inspection by a GxP regulatory authority. However, the longer it goes unnoticed as deficient the higher the price to bring it up to standard. Practitioners must avoid being complacent. Meanwhile, too much validation may delight inspectors, but it is costly in unnecessary time and effort. Once identified as excessive, it may also be difficult to justify a lowering of standards to GxP regulatory authorities.

So how much is enough validation? There is no panacea, but a key step is concentrating effort where it is needed. Emphasis should be placed on validating the GMP aspects of the system. There are often just a few critical functions and components that affect product quality, supported by a collection of measurements and manual interactions. The nature of any manual interactions will have a significant impact on the amount of validation required. Computer systems that only record the manual control of a process will require less detailed validation than those that directly control a process. The collection of product data is another important factor in assessing the degree of control taken by a computer system.

Critical functions and components can be identified as part of a distinct exercise, as discussed in Chapter 7, or as a task integrated within the development of a Validation Plan. Concentrating validation on critical process functions and components can realize considerable savings compared to a detailed blanket validation so often implemented as a precaution against noncompliance. Of course, there may be specific business needs that direct validation to err on the side of caution and include too much detail, such as the validation of a key drug product. Practitioners should refrain from needless validation; GxP regulatory authorities will discuss issues and options for validating computer systems, and practitioners should not be shy or worried about asking their advice. The cost of validation should always be challenged. Other computer systems may be available that are easier to validate, or hard-wired systems or manual systems may be options. Impartial corporate experts or consultants can be used to help develop the most cost-effective validation strategy.

REDUCING COSTS

The cost of validation can appear high if the total cost of ownership for a computer system is not taken into account. Validation can reportedly more than pay for itself through incurred benefits over the operational life of a system. This book has looked at validation metrics and methods for identifying and implementing validation performance improvements. Opportunities for reducing the cost of validation include the following:

- Early definition and controlled maintenance of cost-effective working practices — improved working efficiency
- Off-site contractor work packages — reduced infrastructure costs
- Experienced validation staff — reduced learning curve
- Reuse of validation experience between similar computer systems — reduced duplication of effort
- Independent checks on coordination and consistency of working practices — reduced inefficiencies
- Review validation practices, keep it simple — excessive paperwork and management is burdensome and an unnecessary cost
- Joint supplier audits between pharmaceutical and healthcare manufacturers — share the cost of audit activity

This list of options is not exhaustive, and not all will be appropriate to any one organization. Nevertheless, no organization can afford to ignore such opportunities in the current economic climate. Having said this, it is important not to be overzealous to the degree that satisfactory GxP compliance is compromised. The cost of noncompliance (e.g., nonissue, delay, or revoking of manufacturing license) can be significantly more than what, in hindsight, might appear to be marginal cost savings.

THE FINAL ANALYSIS

The validation of computer systems has been a topical issue for over a decade, and for many companies it is a routine topic for regulatory inspections. The importance of computer systems validation is set to continue with regulatory interest in the role of new technology and ever more pervasive use of computer systems to support the development, manufacture, and distribution of pharmaceutical and healthcare products.[6]

In the final analysis, pharmaceutical and healthcare companies have no choice but to validate their computer systems, otherwise their licenses to market drugs will be revoked or withheld in the first place. The cost of validation need not be excessively high if the exercise is focused.

REFERENCES

1. Trill, A.J. (1996), *MCA View on the Presented Initiatives,* ISPE/PDA Joint Conference on Computer-Related System Validation, Basel, Switzerland, May 2 and 3.
2. Wingate, G.A.S. (1997), *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice*, Interpharm Press, Buffalo Grove, IL.
3. FDA (2002), Pharmaceutical CGMPs for the 21st Century: A Risk Based Approach, www.fda.gov/oc/guideline/gmp.html.
4. GAMP Forum (2003), Risk Assessment for Use of Automated Systems Supporting Manufacturing Processes, *Pharmaceutical Engineering*, May.
5. GAMP Forum (2003), Risk Assessment for Electronic Records and Electronic Signatures, *Pharmaceutical Engineering*, May.
6. Trill, A.J. (1999), *A Current MCA View of Computerised Systems used in Good Practice Applications*, Achieving Cost-Effective Computer Systems Validation for cGMP in Pharmaceuticals, Business Intelligence, London, April 27 and 28.

19 Case Study 1: Computerized Analytical Laboratory Studies

Ludwig Huber, Agilent Technologies

CONTENTS

Computers are widely used in analytical laboratories for instrument control, data evaluation, and data management, and are subject to all validation and verification activities. Verification and validation activities to assess computer systems used in analytical laboratories cover all life-cycle phases, from user Requirements Specifications, design, development, and manufacturing to installation and operation. In this case study, users of computer systems will find guidelines on:

- How to define user requirements and Functional Specifications (FSs)
- Which type of documented evidence the vendor should provide to prove that the system was developed according to recognized standards
- How to validate and document software developed in the user's own laboratory
- How to proceed in cases where vendors do not give evidence of validation
- How to qualify a computer system at installation and for operation
- How to evaluate computer systems in analytical laboratories retrospectively
- How to ensure an ongoing performance control during routine analysis

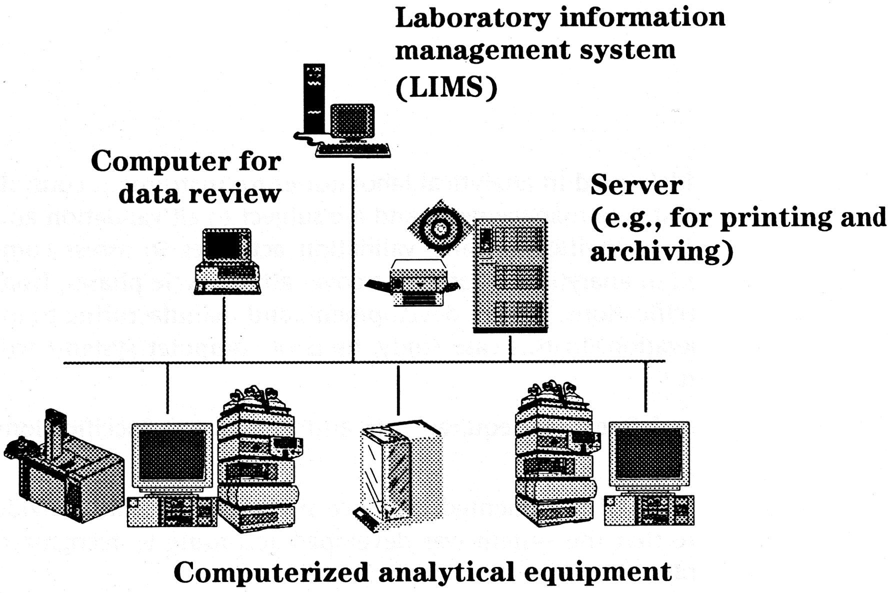

Laboratory information management system (LIMS)
Computer for data review
Server (e.g., for printing and archiving)
Computerized analytical equipment

FIGURE 19.1 Computerized Laboratory.

This case study does not cover verification activities during development at the vendor's site. This is described elsewhere.[1]

A computer system, as used in an analytical laboratory, consists of computer hardware, peripherals, and software to perform a task. It can perform different tasks:

- Instrument control, data acquisition, and data evaluation
- Laboratory information management
- Archiving of electronic records

Figure 19.1 shows an example of a complex computer system that includes computerized analytical instruments for the collection and evaluation of data, servers for data review and centralized archiving, and a laboratory management system.

On a computer used in analytical laboratories, we generally find three different software categories:

1. System software, such as operating software (Windows 2000, Windows XP, NT, or UNIX®), drivers, and file management, supplied by software vendors. These are supplied with the computer in a machine-executable format that cannot be modified by the user and is not unique to any one user's system. The correct function of this software is verified whenever an application runs under the system software.
2. Standard application software, for example, COTS chromatography software, generally supplied by an instrument vendor. The correct function should be verified during and at the end of development. The user must perform acceptance testing prior to use.
3. User specific application software, written by the user or by a third party for a specific user to meet the specific functional needs in the user's laboratory. Examples are macros to customize a system for specific user needs. This software must be validated prior to and during routine use.

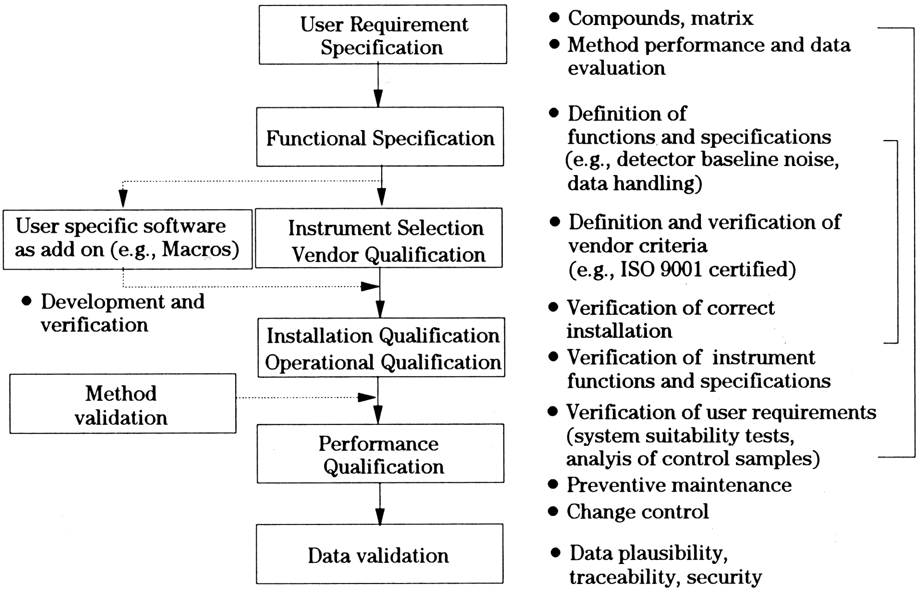

User Requirement Specification
• Compounds, matrix
• Method performance and data evaluation
Functional Specification
• Definition of functions and specifications (e.g., detector baseline noise, data handling)
User specific software as add on (e.g., Macros)
• Development and verification
Instrument Selection Vendor Qualification
• Definition and verification of vendor criteria (e.g., ISO 9001 certified)
Installation Qualification Operational Qualification
• Verification of correct installation
• Verification of instrument functions and specifications
Method validation
Performance Qualification
• Verification of user requirements (system suitability tests, analyis of control samples)
• Preventive maintenance
• Change control
Data validation
• Data plausibility, traceability, security

FIGURE 19.2 Validation Life Cycle.

Overview of Validation Steps Associated with Computer Systems

Figure 19.2 illustrates the validation steps in an analytical laboratory from a user's point of view. For a specific project, validation activities should follow a ten-step validation plan:

1. The user requirements are set. These describe the analysis problem and include instrument performance requirements for a specific analysis task.
2. From the user requirements, the type of analytical equipment, and computer system, the functions and Functional Specifications are derived.
3. The user then selects a standard instrument and appropriate options.
4. A vendor should be selected who develops hardware and software equipment in accordance with a quality assurance system, for example, ISO 9001.*
5. If the standard software supplied by the vendor does not cover all of the user's requirements, user specific software is developed as an add-on macro, either by the user, the vendor, or by a third party.
6. The modules are installed and put together as a system. Correct installation and operation should be verified against Functional Specifications as defined by the user, a process that is called Installation Qualification (IQ) and Operational Qualification (OQ).
7. The proper functioning of analytical methods should be verified on the new system. This covers testing of significant method characteristics, for example, limit of detection, limit of quantification, selectivity, and linearity. If the method has not been validated or if its scope did not cover the new instrument, the method should either be newly validated or revalidated.
8. The performance of the complete system should be validated against the User Requirements Specifications. The system combines the instrument hardware, computer hardware and software, and the analytical method. In chromatography, it also includes a column

* Steps 2 to 4 are sometimes called the Design Qualification. The user should verify that the design of the computer system meets the requirements and that the vendor meets the vendor qualification criteria.

and reference standards for calibration. This validation, usually referred to as system suitability testing, tests a system against documented performance specifications for the specific analytical method. Analytical systems should be tested for suitability prior to and during routine use, practically on a day-to-day basis.

9. When analyzing samples, the data should be validated. The validation process includes documentation and checks for data plausibility, data integrity, traceability, and security.
10. A complete audit trail that allows the final result to be traced back to the raw data should be in place. According to FDA regulation 21 CFR Part 11 on electronic records and signatures,[6] this audit trail must be computer generated and independent of the operator.

Table 19.1 lists in chronological order the steps that a user of computerized analytical equipment can follow for the entire validation process.

The type and degree of validation of a computerized analytical system depends on its complexity. For example, the functions of a simple, computer-controlled system, with little or no flexibility regarding data input or evaluation, can be verified by executing holistic tests[2] and by comparing the test results with anticipated results. On the other hand, a more complicated computerized system with on-line databases and extensive flexible data evaluation requires complex validation.

Assessment of the Need for Validation and the Validation Plan

Before validation can begin, management should assess whether the system requires formal validation. Criteria to be considered are whether the system will be used in a regulated or quality standard environment and how critical the data generated by the system are. Examples of such environments are as follows:

- Good Laboratory Practice (GLP)
- Good Manufacturing Practice (GMP)
- Good Clinical Practices (GCP)
- ISO 17025

When the decision has been made about the need for a system validation, sufficient resources should be allocated. For larger projects, the recommendation is to form a validation team consisting of Quality Assurance personnel and technical experts. All validation activities at the user's site should follow a Validation Plan.

User Requirements and Functional Specifications

Requirements Specifications define how the system will be used. For an analytical system, these can include the following:

- The type of compounds and matrix
- The expected limits for detection and quantitation
- The expected precision and accuracy
- The number of samples in a given time frame and mode of operation: manual or automated
- The type of information: qualitative and/or quantitative
- The type of computer and IT environment for instrument control, data acquisition, and data evaluation, e.g., LAN-based networked data system

TABLE 19.1
Recommended Validation Steps with Examples

Step	Explanation	Examples
Define user requirements.	Criteria: compounds, matrix, detection limit, precision, selectivity, accuracy, concentration range, qualitative or quantitative, throughput, regulatory day requirements, e.g., 21 CFR Part 11 (6).	Analysis of phenoxy acid herbicides in drinking water, detection limit: 0.01 μg/l, qualitative and quantitative information, electronic archiving of data, 30 samples/day. Method and equipment: solid phase extraction and high performance liquid chromatography (HPLC)/diode array detection. A computer and associated software control the instrument and acquire and evaluate data.
Define functional specifications.	Define intended equipment hardware, software, and system functions and operational limits; define regulatory requirements.	HPLC: binary gradient, flow rate range and system 0.2 to 2 ml/min, diode-array detector with 10-mm path length, baseline noise limits $\leq 4 \times 10^{-5}$ AU, computer for integration, quantification, peak purity check, interactive and automated spectral library search, qualitative and quantitative report, system suitability testing and archiving of method parameters with raw data file, limited and authorized access to the system and data through user-ID and passwords, electronic audit trail, electronic signatures; signatures must be bound to electronic records.
Select and qualify vendor-purchased equipment.	Develop criteria for vendor selection, and check if criteria are met.	Quality system, availability of validation documentation, local support and response time, reputation and experience; information through documentation available from the vendor.
Develop user-specific software.	Macro, spreadsheet calculations.	Customized reporting; statistical evaluation.
Qualify modules and systems prior to routine use.	Installation Qualification, Operational Qualification.	Check if shipment complies with purchase order; test equipment (e.g., test the precision of amounts and retention times); verify correct software installation.
Validate methods (optional, if the methods are not already validated).	Specify validation parameters and acceptance limits. Define and execute validation experiments.	Limit of detection, limit of quantification, selectivity, linearity, precision, accuracy, ruggedness.
Assure ongoing performance.	Develop and implement schedules and procedures for periodic preventive maintenance, for calibration, and for initial and ongoing Performance Qualification. Develop and implement procedures for error detection, recording, and handling. Develop procedures for change control.	Calibration of balances; calibration of ultraviolet grating for wavelength accuracy; exchange of lamps; system suitability testing; analysis of quality control samples and evaluation of results using control charts. ROM check at system boot up; automated shutdown of pump if leak is detected in an autosampler, or authorize changes to user-written change control software.
Assure validity, security, integrity, and traceability of data.	Develop and implement security relevant procedures.	Limited system access through user-specific passwords; data file integrity through checksum or other routines.

- The type of information calculated from original data, printed in the report and stored on electronic media
- Archiving of data

From the Requirements Specifications, the user can derive the instrument type and its minimal Functional Specifications. For example, if an instrument is scheduled to run overnight, the number of samples should be specified so that the system can inject automatically. The UV/visible detector's baseline noise specification can be determined from the specified detection and quantitation limit of an HPLC analysis. The required data evaluation will determine the demands on the evaluation software.

The next step is either to select an existing system for the analysis task or to purchase a new system. When a new system is purchased, it is often purchased not just for a specific analysis but also for use in general applications in the laboratory. In this case, the Requirements Specifications should include a representative mix of the anticipated applications, and the FSs should be set such that the instrument can handle all of the requirements. Next, the user should look for an instrument on the market that best meets these requirements. If the selected system does not provide all of the functions — for example, regarding software — the user can decide whether to develop these himself or ask the vendor or a third party to do so.

Responsibilities of Vendors and Users

New software and computerized systems in analytical laboratories are usually purchased from a vendor. Such COTS software must be validated as the FDA Part 11 validation draft guidance (7) states: "Commercial software used in electronic record keeping systems subject to part 11 needs to be validated, just as programs written by end users need to be validated." A frequently asked question is who is responsible for the validation of such a system: the vendor or the user? The Organisation for Economic Co-operation and Development (OECD) states clearly in consensus paper number 5: "It is the responsibility of the user to ensure that the software program has been validated."[3] This is also the practice of the U.S. Food and Drug Administration (FDA) as specified by Ron Tetzlaff, a former FDA investigator: "The responsibility for demonstrating that systems have been validated lies with the user."[4] Similarly, the 21 CFR Part 11 validation draft guidance (7) states: The end user is responsible for a program's suitability as used in the regulatory environment. However, it is obvious that product quality cannot be achieved by testing in a user's laboratory. This must be incorporated during design and development. Therefore, the OECD makes a further statement in a new consensus paper:[5] "There should be adequate documentation that each system was developed in a controlled manner and preferably to recognized quality and technical standards (e.g., ISO 9001)." Furman et al.[2] from the U.S. FDA also make it clear: "All equipment manufacturers should have validated their software before releasing it."

- The vendor is responsible for assuring that the system is developed, tested, and supported according to proper development and change control procedures.
- The user is responsible for the entire validation. One part of this overall validation process is to obtain documented evidence about the proper development according to documented standards. To have this assurance, a vendor assessment program should be developed that may include formal written procedures for the selection, evaluation, and qualification of vendors.

One criterion for vendor qualification should be whether the vendor has a documented Quality Assurance program that follows recognized quality standards, for example, ISO 9001. This registration is usually sufficient for those laboratories that must comply with ISO 9001 or with a laboratory accreditation standard such as ISO 17025. However, it is the author's experience that regulatory agencies do not always accept such third-party evaluation or successful registration according to ISO 9001, or the equivalent, as proof of a vendor's qualification. They expect vendors to have other proof of qualifications such as documented familiarization of their staff with software development practices, GLPs and cGMPs, procedures for software backup, archiving and periodic integrity checks, a software tracking and response system, or references from internal or external users of the system. Frequently, additional documents are requested, such as development validation certificates and an assurance by the vendor of accessibility to the source code by regulatory agencies.

Validation of Newly Purchased Systems without Evidence of Validation from the Vendor

If a vendor is not able or willing to provide documented evidence of validation, the user should consider selecting another vendor. This recommendation is easy to follow if there are a number of competitors for the same or similar products. If this is not the case, for example, when special software for an emerging technique is required, a user may purchase the software anyway, but should evaluate the vendor and perform more thorough testing and keep more detailed documentation. For example, for an evaluation of the vendor, checklists can be sent to the vendor with questions on the following topics:

- The company (history, size, financial status, number of employees)
- The organization (Quality Assurance department)
- Certifications (ISO 9001)
- Hardware and software development (quality and technical standards)
- Testing and verification of life-cycle phases (reviews of Requirements Specifications, design, code inspections, test traceability matrix from test cases back to requirement specifications, are there release criteria for the phases?)
- Source code (guaranteed accessibility to regulatory agencies, where stored, escrow account)
- The product (How many sold to the target industry?)
- Security (How are unauthorized changes prevented? Do disaster plans exist?)
- Support (response, language, phone, on-site, modem, support plan existing)
- Handling of failures and enhancement requests (formal procedures)
- Change control (Who initiates and authorizes changes, version identification, and revision history?)
- People qualification (training programs)
- User documentation (What is archived and for how long?)
- Customer training (topics, frequency, next location)
- Equipment hardware (How are specifications verified?)

A detailed checklist for vendor qualification has been developed by the author.[1] If the answers to the checklist are not satisfactory, a direct audit should be considered.

If no documentation on validation during development can be obtained from the vendor, the user should evaluate the system retrospectively. For example, the 21 CFR Part 11 draft validation guidance[7] states: "When the end user cannot directly review the program source code or development documentation, more extensive functional testing might be warranted than when such documentation is available to the user." Detailed test cases with well-characterized test data sets and known results should be developed to evaluate the correct functioning of all software programs. In the case of completely computerized analytical systems, the analysis results with reference standards or quality control samples should be compared with known results from the standard or sample. Besides this holistic test, it is recommended to verify the performance of each individual subsystem. Such checks should include the proper response of the equipment to inputs from the computer. The tests should also check the system's error-handling capabilities. The system should recognize and display any wrong entries — for example, entries that are out of the system's operating range. Another simple test would be to check how the program responds when alphanumeric data are entered into fields that are designed to accept numerical values. In addition, the mathematical formulas should be verified with alternative methods of calculation.

Validation of User-Contributed Software (e.g., Macros)

Application software developed by the user should be fully validated and documented by the user. Such software may be a stand-alone software package (e.g., for statistical data evaluation), or it may be an extension to purchased standard software (e.g., a macro to enhance functionality). The development and validation of such software should follow a documented procedure, and the source code should be available. The effort involved in validation depends very much on the size and complexity of the program. The development of large programs should follow the software development life cycle and can take several months or years. Validation can take several weeks and the documentation will be extensive. On the other hand, the validation of small programs can be done in a few hours and the documentation may be only a few pages. The development, validation, and documentation of such small programs requires the following steps at minimum:

1. Describe the problem, how the problem is solved currently, and how the newly developed program will solve it.
2. Identify responsibilities for development, test, and approvals.
3. Describe the task and the system requirements (hardware, system software, standard software).
4. Describe the program in terms of the functions it will perform.
5. Document formulas and algorithms used within the code.
6. Write and document the code in such a way that it can be understood by other people whose knowledge and experience are similar to the programmer's. Print the code.
7. Develop test cases and data sets with known inputs and outputs. Include test cases with normal data across the operating range and at the boundary, and some unusual cases with incorrect inputs. The results should be calculated by the new program and also by using alternative methods. The development of an automated test procedure that can be executed as often as possible is recommended. Test procedures and results should be documented, reviewed, and signed off.
8. Develop user documentation with information on how to install, test, and operate the program.
9. Describe and implement procedures for data backup and security routines for limited access to authorized people.
10. Develop a procedure to authorize, test, document, and approve any changes to the software and documentation.

For combined systems, vendor-updated software revisions may be critical, especially if the updated version supplied by the vendor will have an effect on the interface between the vendor's and the user's software (e.g., if the meaning of a macro command has been changed). The user should obtain information from the vendor on how the updated version may affect the interface. The user should also test his or her software after it has been integrated into the vendor's updated standard software. More details are found elsewhere about Standard Operating Procedures (SOPs) for developing and validating simple as well as complex applications software developed in the user's laboratory.[8]

Preinstallation

Before the instrument arrives at the user's laboratory, serious thought must be given to its location and space requirements.

- A full understanding of the new equipment must be obtained from the vendor well in advance: required bench or floor space and environmental conditions, such as humidity and temperature, and, in some cases, utility needs, such as electricity and compressed gases for gas chromatographs.
- Care should be taken that all of the environmental conditions and electrical grounding are within the limits as specified by the vendor and that correct cables are used.
- If environmental conditions may have an influence on the validity of test results, the laboratory should have facilities to monitor and record these conditions, either continuously or at regular intervals.
- Any special safety precautions should be considered (e.g., for radioactivity measurement devices), and the location should also be checked for any devices generating electromagnetic fields nearby.

Installation

Once the instrument arrives:

- The shipment should be checked by the user for completeness.
- It should be confirmed that the equipment ordered is what was in fact received. Besides the equipment hardware, other items should be checked (e.g., correct cables, other accessories, and documentation).
- The documentation should be checked for completeness (operating manuals, maintenance instructions, SOPs for testing, safety, and validation certificates).
- For more complex instrumentation, wiring diagrams should be generated, if not obtained from the vendor.
- An electrical test of all modules and systems should follow.
- The impact of electrical devices close to the computer system should be considered and evaluated if a need arises. For example, when small voltages are sent between sensors and integrators or computers, electromagnetic energy emitted by poorly shielded nearby fluorescent lamps or by motors can interfere with the transmitted data.
- When complex software is installed on a computer, the correctness and completeness of the installed program and data files should be verified. Vendors can assist this process by supplying installation reference files and automated validated verification procedures. In this case, the integrity of each file is verified by comparing the cross-redundancy check (CRC) of the installed file with the checksum of the original file recorded on the installation master. Modified or corrupt files have different checksums and are, thus, detected by the verification program. Verification reports include a list of missing, changed, and identical files.

The installation should end with the generation and sign-off of the installation report — in pharmaceutical manufacturing referred to as the IQ document. The hardware and software should be well documented with model, serial, and revision numbers. For larger laboratories with lots of equipment, this should preferably be a computer database. Entries for each instrument should include the following:

- In-house identification number
- Name of the item of equipment
- The manufacturer's name, address, and phone number for service call; service contract number, if available
- Serial number and firmware revision number of equipment
- Software with product and revision number
- Date received
- Date placed in service
- Current location
- Size, weight
- Condition, when received (e.g., new, used, reconditioned)
- List of authorized users and responsible person

It is recommended to make copies of all important documentation: one copy should be placed close to the instrument; the other should be kept in a safe place. An identification sticker should be put on the instrument with information about the instrument's serial number and the company's asset number.

Logbook

An electronic or bound paper logbook should be prepared for each instrument in which operators and service technicians record all equipment-related activities in chronological order. Information in the logbook can include the following:

- Logbook identification (number, valid time range)
- Instrument identification (manufacturer, model name/number, serial number, firmware revision, date received, service contact)
- Column entry fields for dates, times, and events (e.g., initial installation and calibration, updates, column changes, errors, repairs, performance tests, quality control checks, cleaning, and maintenance, plus fields for the name and signature of the technician making the entry).

Operational Qualification and Acceptance Testing

After the installation of hardware and software, the hardware should be calibrated, where required. An operational test should follow a process that is referred to in pharmaceutical manufacturing as OQ. The goal is to demonstrate that the equipment's hardware and software operate "as intended" in the user's environment.

For a computer system in an analytical laboratory, OQ can mean, for example, verifying correct communication between the computer and other hardware. As part of the product documentation, vendors should provide operating procedures for the tests, limits for acceptance criteria, and recommendations in case these criteria cannot be met. The documentation should also include algorithms for critical calculations and procedures on how to verify the algorithms in a user's environment. If the user finds the tests recommended by the vendor inappropriate or insufficient, the user can design and perform other or additional tests.

Chemical standards used for instrument calibration or qualification tests should be traceable to national standards.

The documentation of testing should include the following:

- The description and unique identification of equipment
- Test items

- Acceptance criteria
- Summary of results
- The date
- Names and signatures of persons who performed the tests

The instrument should be labeled with the calibration and qualification status, indicating the dates of the last and next calibration and OQ.

Qualification of Software

The correct functioning of software loaded on a computer system should be checked in the user's laboratory under typical operating conditions and under high load conditions. During the equipment hardware test, as described in the previous section, many software functions are also executed, such as instrument control, data acquisition, peak integration, quantitation, file storage and retrieval, and printing. Therefore, after successful completion of hardware tests, it can also be assumed that the software operates as intended. There are two situations where software verification independent of the equipment hardware may be necessary:

1. If not all critical software functions are executed during the hardware verification
2. If a verification of the software functions should be done without a need for equipment testing

This is the case after a change on the computer system — for example, if a new operating system has been installed or if new hardware, such as CD-ROMs or a hard disk, has been installed on the computer system.

The following paragraphs describe a procedure for the verification of important chromatography software functions without injecting a sample (Figure 19.3). The concept has been described in detail elsewhere.[1] It is very generic and can also be used to test and verify the correct functions of other software packages.

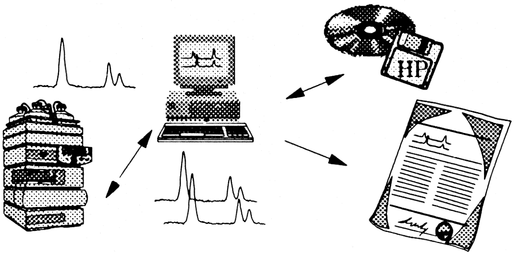
Well-characterized test chromatograms derived from standards or real samples are stored on disk as a master file. Chromatograms may be supplied by the vendor as part of the software package or can be recorded by the user. This master data file goes through normal data evaluation from

Generate Master Data

1. Generate master chromatogram.
2. Develop method for evaluation.
3. Generate master result.
4. Save results electronically and on paper.

Verification

1. Select master chromatogram and method.
2. Run test (manually or automatically).
3. Compare new results with master data.
4. Report verification results.



Test items

- Data transfer
- Data acquisition
- Integration
- Quantification
- Storage
- Retrieval

FIGURE 19.3 Verification Process of Chromatographic Software.

integration to report generation. Results are stored on the hard disk. The same results should always be obtained when using the same data file and the same method for testing purposes.

Preferably, tests and the documentation of results should be done automatically, always using the same set of test files. In this way, users are encouraged to perform the tests more frequently, and user specific errors are eliminated. In some cases, vendors provide test files and automated test routines for verification of a computer system's performance in the user's laboratory. Needless to say, the correct functioning of this software should also be verified. If such software is not available, the execution of the tests and the verification of actual results with prerecorded results can be done manually.

Successful execution of such a procedure ensures that:

- Executed program and data files are loaded correctly on the hard disk.
- The current computer hardware is compatible with the software.
- The current versions of the operating system and user interface software are compatible with the application software.
- Data are correctly transferred between the equipment and the computer (if this feature is supported by the system).

In addition to typical functions required by the application, other functions required by regulations and internal company policies should be tested. These include:

- Limited and authorized access to the system and data. This can be achieved by trying to enter the system with correct and incorrect combinations of passwords and user-IDs.
- Electronic audit trail. Check if the audit trail records events as specified in the functional requirement specifications document.
- Electronic signatures. Check if a signature includes the full name of the person who signed, date and time, and a meaning.

When data are transferred between computers through a network, the accuracy of data transfer should be verified. This can be achieved by comparing printouts before and after transfer or comparing hash factors before and after data transfer. More detailed information on the qualification and testing of networks using hash factors can be found in Reference 9.

Routine Maintenance and Ongoing Performance Qualification

When the installation is complete and the equipment and the computer system are proven to operate well, the computerized system is put on routine analysis. Procedures should exist that show that "it will continue to do what it purports to do."

Each laboratory should have a quality assurance program that is well understood and used by individuals, as well as by laboratory organizations, to prevent, detect, and correct problems. The purpose is to ensure that the results have a high probability of being of acceptable quality. Ongoing activities may include the following:

- Preventive instrument maintenance
- Performance verification and calibration
- System suitability testing
- Analysis of blanks and quality control samples
- Ensuring system security

Preventive Maintenance

Operating procedures for maintenance should be in place for every system component that requires periodic calibration and/or preventive maintenance. The idea is to replace critical maintenance parts before they have a negative effect on the quality of analytical data.

- Critical parts should be listed and should be available at the user's site.
- The procedure should describe what should be done, when it should be done, and what the qualification of the engineer performing the tasks should be.
- System components should be labeled with the date of the last and next maintenance.
- All maintenance activities should be documented in the instrument's logbook.
- Suppliers of equipment should provide a list of recommended maintenance activities and procedures (SOPs) on how to perform the maintenance.

All maintenance activities should be recorded in a maintenance logbook. To make this more convenient, modern equipment includes electronic maintenance logbooks where the user enters the type of maintenance, and the equipment records this activity together with the date and time.

Calibration

Operating devices may become miscalibrated after a while (e.g., the temperature accuracy of a gas chromatography [GC] column oven or the wavelength accuracy of a UV/visible detector's optical unit). This can have an impact on the performance of an instrument. Therefore, a calibration program should be in place to recalibrate critical items of an instrument.

- All calibrations should follow documented procedures and the results should be recorded in the instrument's logbook.
- The system components should be labeled with the date of the last and next calibration.
- The label on the instrument should include the initials of the test engineer; the form should include his/her printed name and full signature.

System Suitability Tests and Quality Control Sample Analysis for Ongoing Performance Qualification

The analysis of standards or Quality Control samples with the construction of Quality Control charts (Figure 19.4) has been suggested as a way to incorporate quality checks on results as they are being generated. Such tests can then flag those values that may be erroneous for any of the following reasons:

- Reagents are contaminated.
- GC carrier gas is impure.

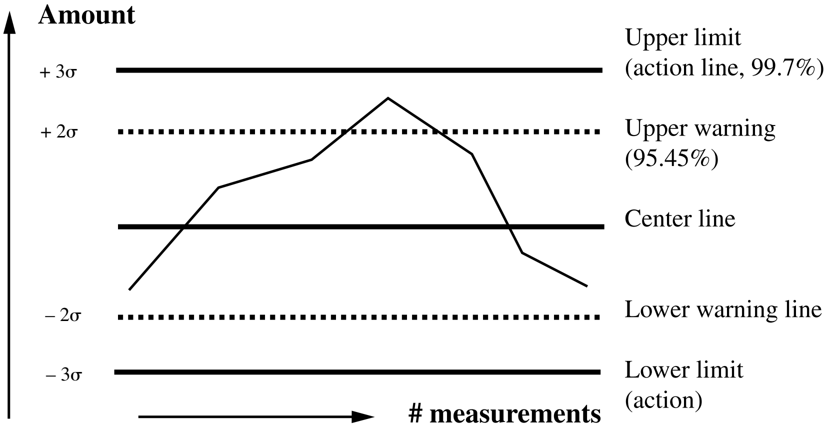


FIGURE 19.4 Quality Control Chart with Warning Lines and Control Lines.

- HPLC mobile phase is contaminated.
- Instrument characteristics have changed over time.

For an accurate quality check, Quality Control samples are interspersed among actual samples at intervals determined by the total number of samples and the precision and reproducibility of the method. The control sample frequency will depend mainly on the known stability of the measurement process — a stable process requiring only occasional monitoring.

Control samples should have a high degree of similarity to the actual samples analyzed; otherwise, one cannot draw reliable conclusions on the measurement system's performance. Control samples must be so homogeneous and stable that individual increments measured at various times will have less variability than the measurement process itself. Quality Control samples are prepared by adding known amounts of analytes to blank specimens. They can be purchased as certified reference material (CRM) or may be prepared in-house. In the latter case, sufficient quantities should be prepared to allow the same samples to be used over a longer period of time. Their stability over time should be proven and their accuracy verified, preferably through interlaboratory tests or by other analysis methods.

The most widely used procedure for the ongoing peformance control of equipment through Quality Control samples involves the construction of control charts for these samples. These are plots of multiple data points vs. the number of measurements from the same samples using the same processes. Measured concentrations of a single measurement or the average of multiple measurements are plotted on the vertical axis, and the sequence number of the measurement is plotted on the horizontal axis. Control charts provide a graphics tool to demonstrate statistical control, monitor a measurement process, diagnose measurement problems, and document measurement uncertainty. The most commonly used control charts are X-charts and R-charts as developed by Shewart. X-charts consist of a central line representing either the known concentration or the mean of 10 to 20 earlier determinations of the analyte in control material. The standard deviation has been determined during method validation and is used to calculate the control lines in the control chart. Control limits define the bounds of virtually all values produced by a system in statistical control.

Control charts often have a center line and two control lines with two pairs of limits: a warning line at m ± 2s and an action line at m ± 3s. Statistics predict that 95.45% and 99.7% of the data will fall within the areas enclosed by the ± 2s and ± 3s limits. The center line is either the mean or the true value. In the ideal case, where unbiased methods are being used, the center line would be the true value. This would apply, for example, to precision control charts for standard solutions.

When the process is under statistical control, the day-to-day results are normally distributed about the center line. A result outside the warning line indicates that something is wrong. Such a result need not be rejected, but documented procedures should be in place for suitable action. Instruments and sampling procedures should be checked for errors. Two successive values of the Quality Control sample falling outside the action line indicate that the process is no longer under statistical control. In this case, the results should be rejected and the process investigated for its unusual behavior. Further analyses should be suspended until the problem is resolved.

Change Control

Software has one distinct advantage over hardware: it does not change its performance characteristics over time. Theoretically, there should be no need to revalidate software as long as the hardware and environmental conditions do not change. However, almost 100% of all software written will be changed following its release for use. There are three reasons for a software change:

1. To correct errors
2. To adapt software to changes in its operating environment
3. To enhance the software (e.g., to add functionality)

In addition to software changes, there may be hardware changes to a computer system. The processor may be upgraded, the system may get additional disk space, or new memory chips may be installed. All software and hardware changes to a computer system may influence the performance and correct functioning of the system and may need a reverification or revalidation. A control system should exist that describes what should be revalidated after a change to the system. Procedures should be available for changes to a system purchased from a vendor as well as for software developed in-house.

Software developed and validated by the user should also be revalidated by the user. Principally, software redevelopment and testing should follow the same procedure as for newly developed software. Procedures should include information on who authorizes changes, who executes the changes, and who can finally release the changed version. Compared to new software, the amount of testing can be reduced through intensive reuse of previously developed test files and test procedures.

If computer systems are upgraded with new operating systems or when new hardware is added, the user should thoroughly document the upgrade and perform acceptance testing as for a new system. The use of existing test files and automated procedures can make this process very efficient.

If the user purchases a software upgrade, the supplier should supply documentation with a description of the change, a statement that the upgrade has been validated by the supplier during development. The supplier should provide information on:

- System requirements for the upgraded version
- How to install the upgrade
- Impact on macros written by the user
- Recommendation on what to test in the user environment

The installation should be documented and the user should perform an acceptance testing before the system is authorized for use.

REFERENCES

1. Huber, L. (2002), *Validation of Computerized Analytical and Networked Systems*, Interpharm Press, Buffalo Grove, IL.
2. Furman, W.B., Layloff, T.P., and Tetzlaff, R.F. (1994), Validation of Computerized Liquid Chromatographic Systems, *J. AOAC Intern.* 77(5): 1314–1318.
3. OECD (1992), Compliance of Laboratory Suppliers with GLP Principles. Series on Principles of Good Laboratory Practice and Compliance Monitoring, No. 5, GLP consensus document, Environment Monograph No. 49. Organisation for Economic Co-operation and Development, Paris.
4. Tetzlagg, R.F. (1992), GMP Documentation Requirements for Automated Systems: Part III, FDA Inspections of Computerized Laboratory Systems, *Pharm. Tech.* (May): 71–82.
5. OECD (1995), The Application of the Principles of GLP to Computerized Systems. Series on Principles of Good Laboratory Practice and Compliance Monitoring, No. 10, GLP consensus document, Environment Monograph No. 116. Organisation for Economic Co-operation and Development, Paris.
6. Code of Federal Regulations, Title 21, Food and Drugs, Part 11, *Electronic Records; Electronic Signatures*, Final Rule, *Fed. Register* 62(54): 13429–13466, 1997.
7. FDA, Guidance for Industry, 21 CFR Part 11, *Electronic Records; Electronic Signatures*, Validation, 2001, draft.

8. Huber, L., Macro and Spreadsheet Quality Package, Labcompliance, 2002, www.labcompliance.com/books/macros.
9. Huber, L., Network Quality Package, Labcompliance, 2003, www.labcompliance.com/books/network-quality.htm.

20 Case Study 2: Chromatography Data Systems

Bob McDowall, McDowall Consulting

CONTENTS

Chromatography is an analytical technique used in virtually all areas of the pharmaceutical and biotechnology industries to detect or measure compounds during the course of product development and manufacture. It can be used for the measurement of active ingredients, raw materials, and impurities and for determining the stability of active substances in final preparations. The chromatograms from these analytical methods are generated, displayed, and integrated, and results calculated, by a software application called a chromatography data system (CDS).

This chapter presents some approaches to prospectively and retrospectively validating client server networked CDS based on case studies; in addition the business benefits that can be exploited from the implementation of electronic signatures when remediating or upgrading a legacy chromatography data system are presented.

OPERATIONAL CAPABILITY

Figure 20.1 shows the typical workflow performed by a CDS; further details can be found in articles by McDowall[1,2] and a book by Dyson.[3]

Method Files

The start of the data acquisition operation of a chromatography data system is to build a method file. This tells the data system how to acquire data and how to process and interpret the results. A method file should control:

- The data sampling rate of the analog to digital (A/D) converter[4]
- When to start and stop the integration of the chromatogram
- Whether peak areas or heights should be used
- Retention time windows and identification of the analytes and internal standard, and should allocate the method to calculate the analyte amount or concentration

A name, number, or a mixture of both should identify individual method files within the system. In addition, the system should be able to provide facilities for version control of method files to ensure that control is maintained over the method for the lifetime of its use. Part of the control function must be access control to identify the individuals who can create, modify, or delete

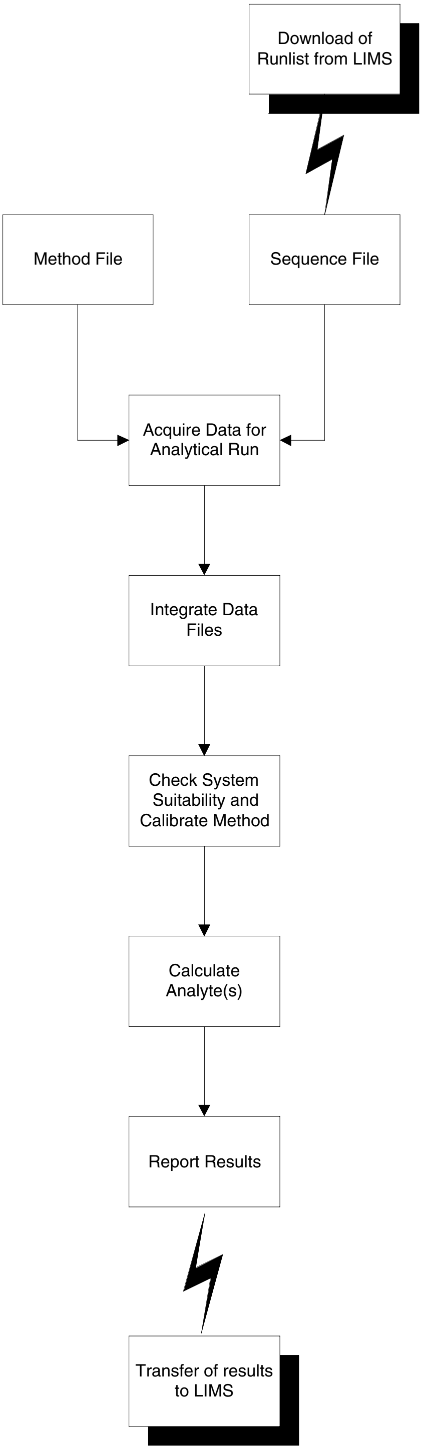

Download of Runlist from LIMS
Method File
Sequence File
Acquire Data for Analytical Run
Integrate Data Files
Check System Suitability and Calibrate Method
Calculate Analyte(s)
Report Results
Transfer of results to LIMS

FIGURE 20.1 Workflow for a Typical Chromatography Data System.

analytical methods. If a method has been modified, then copies of the modifications must be stored with the data processed by that method. This is to provide an audit trail for the data and results produced by a version of a method. However, when developing methods, flexibility with method files is essential and a default method should be available to acquire data and then feedback to a normal method.

Naming Conventions

When a laboratory uses a client-server CDS there will be an urgent need to consider naming conventions for method, sequence, and all data files within the data system. Any CDS must have sufficient capacity for naming all of the files that would be created by the system over a reasonable time period to aid efficient archiving and unambiguous identification of these files. Therefore for efficient management of data files and methods, naming conventions should be introduced. Any naming convention system must aid users, quality assurance, and regulatory inspectors.

A naming convention should be based on the workflow undertaken by a laboratory. This is to allow efficient archiving of data but just as importantly, the efficient retrieval of data. Some ideas might be to

- Organize the data around drug products or development projects. This is how the work is structured and how project teams are organized as it will help retrieve data to aid 21 CFR Part 11 compliance for ready retrieval of electronic records.
- Base major subdivisions of each project around the type of work done, e.g., method development, method validation, preformulation, etc.

Sequence File

The sequence file is the run list or order that the samples, standards, quality control samples, and blanks will be injecting into the chromatograph; this is essential as it puts in context to the individual data files. Each sequence file or each injection must be linked with a method file to process the resulting data. For laboratories with large numbers of samples for a single method, the sequence file will usually be linked with a single method. Smaller laboratories may need the flexibility to link the sequence file with several methods during the course of a single analytical run for maximal use of equipment resources.

Each sample to be analyzed should be identified in the sequence file as one of the following types:

- Unknown
- Calibration standard
- Quality control
- Blank

Depending on the data system involved, at least the first two options are available to a user. There may also be a sample number to link the injection to the physical sample used for analysis.

Interpretation of Chromatographic Data

After the method file and the sequence file have been set up, the analytical run is started and data are collected. A data file containing the A/D data slices will be obtained for each chromatographic run and sample injected. It is important from scientific and regulatory considerations that the data files must not be capable of alteration.

Moreover, they must not be overwritten either if the same sample information is assigned to an assay or if the disk becomes full. This is an area for consideration when validating the chromatography data system as it is important to know what happens to data files, especially in a regulated environment.

The data system will interpret each data file, identifying the individual peaks and fitting the peak baselines according to the parameters defined in the CDS method as shown in Figure 20.2. The data systems should have the ability to identify whether the peak baselines have been automatically or manually interpreted. This is a useful feature for compliance with Part 11 to indicate the number of times a chromatogram has been interpreted.

Most data systems should be able to provide a real-time plot, so that the analyst can review the chromatograms as the analytical run progresses. In addition, the plotting options of a data system should include:

- Fitted baselines
- Peak start/stop ticks
- Named components
- Retention times
- Timed events, e.g., integration start/stop
- Run time windows and user-defined plotting windows
- Baseline subtract

Each of these options should be enabled or disabled by a user.

An overlay function should be available to enable comparisons between results and samples. This will be used to compare and contrast chromatograms from the same run sequence as well as chromatograms from different sources. The maximum number of overlays will vary from data

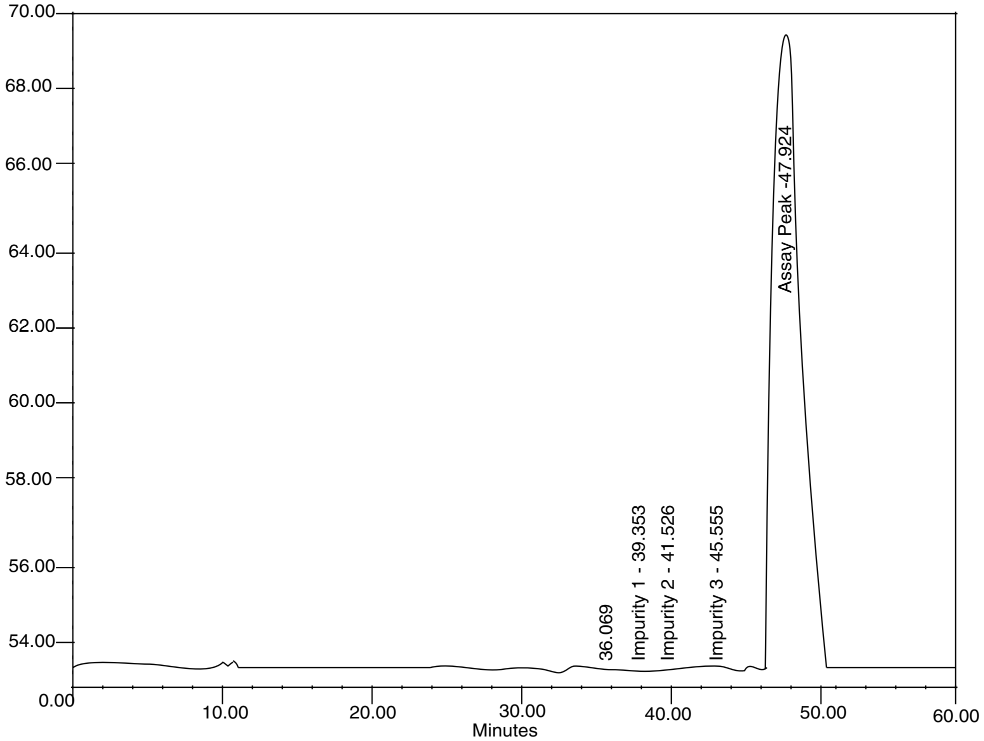


FIGURE 20.2 A Typical Chromatogram of an Active Substance Separation from Impurities/Degradation Products.

system to data system but a minimum of six to eight is reasonable and practicable. More overlays may be technically possible, but the amount of useful information obtained may be limited. Overlays that can be offset by an amount determined by the user are useful to highlight certain peak information. Ideally, the overlay screen should have hidden lines removed and be able to be printed.

Calibration

Calibration is a weak area with most data systems, as most chromatographers use many ways to calibrate their methods as evidenced by the multitude of calibration options available. Often these methods are basic and lack statistical rigor, as the understanding of many chromatographers, where calibration is concerned, can be poor.

Within a pharmaceutical analysis laboratory, the number of calibration model options that can be successfully used is usually limited to:

- Bracketed standards at one concentration or amount for bulk drug or finished product assays
- Response function for all analytes
- Average by amount for bulk drug and finished products
- Multilevel or linear regression for related substances and degradation products

Within each calibration type, the data system must be able to cope, with sufficient flexibility, with variations in numbers of standards used in a sequence and with types of standard bracketing. The incorporation of a blank standard into the calibration curve should always be an option.

Each plot of an analyte in a multilevel or linear regression calibration model must contain an identifier for that calibration line and the analyte to be determined. The calibration curve should show all calibrating standards run in any particular assay. In assays containing more than one analyte it will be necessary to interpret all the calibration graphs before the calculation of results. Again, this is an area that is poor for data system as many only offer one line fitting method for all analytes in the run, resulting in compromises.

User-Defined Analytical Run Information

The system should be capable of collating user-defined parameters (e.g., height, area, ratios, concentrations, etc.) for selected analytes from a sequence of runs. After collation, system defined and/or user defined statistical calculations will be carried out on the data generated. The type of calculations required should include mean and standard deviation an analysis of variance, and possibly significance testing.

Reports and Collation of Results

Ideally, the report following an individual chromatogram should contain both elements that are user definable and those which are standard; this should enable the laboratory to customize a report. At the end of the analytical run, a user-defined summary report containing information such as sample ID, area or height, baseline, and calculated analyte concentration should be created. This report can either be printed out or transferred to a LIMS for further analysis and interpretation.

Instrument Control

The primary interaction of the CDS with analytical instrumentation is with the output from the detector; however, there are other considerations such as instrument control. These can vary from system to system, and the following options are available:

- Contact closures for the control of chromatographic valves or associated equipment during analysis — usually available for other suppliers' equipment.
- When the same supplier makes the data system and the chromatography equipment, control is more sophisticated and more tightly integrated with the data system functions, so control of the instrument and set up of the data system can be achieved from a single workstation.
- Communication with the auto-sampler via Binary Coded Decimal (BCD) or equivalent communication for sample continuity. This is, in my view, essential, but it is usually ignored by many and offered as an option by many suppliers.
- Remote monitoring of the chromatography system output, including the instrument conditions.
- The ability to list the items of equipment (pump, detector, etc.) used for a particular analysis. This function helps to automate the administrative records associated with an analysis and to meet GMP compliance.

ARCHITECTURE OF A NETWORKED CDS

A typical networked chromatography data system will consist of several hardware components as shown in Figure 20.3:

- Chromatograph: This is the instrument that performs the analytical separation and can be a high performance liquid chromatograph (HPLC), gas chromatograph (GC), or a Capillary Electrophoresis (CE) instrument.
- Data acquisition moves via an analog to digital (A/D) converter from the instrument detector to the CDS and converts the continuous analog signal to a number of discrete digital data readings. This is an optional item; if the instrument is controlled by the CDS, data are transferred digitally via the network cables running to the data system. Often

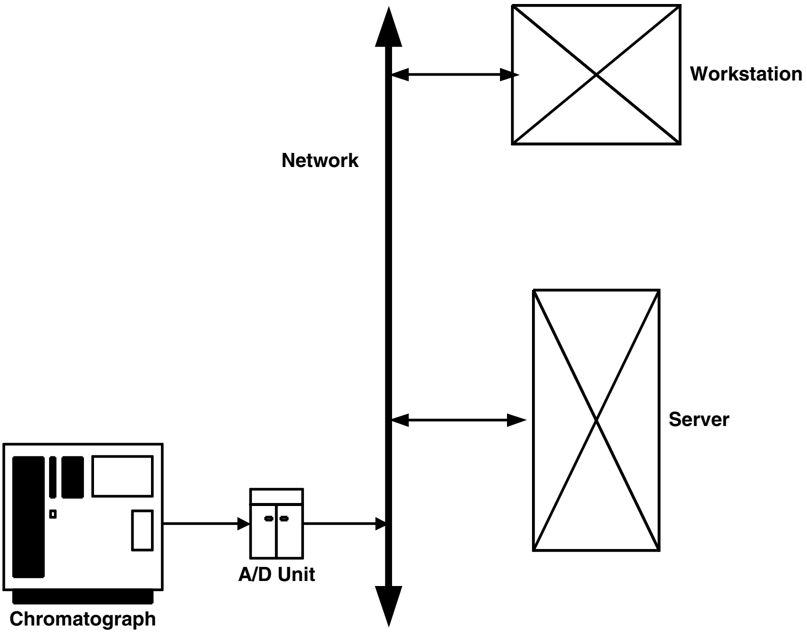


FIGURE 20.3 Schematic Diagram of a Typical Networked Chromatography Data System.

the A/D unit can have buffering capability to prevent data loss if the network is temporarily unavailable.
- Network: Transport medium for moving the data from the instrument to a server for secure data storage.
- Workstation (Client): This is for operating the CDS, setting up an instrument, checking that the separation is working correctly, and interpreting the resultant chromatograms after the run is finished, then reporting the results.

KEY REGULATORY REQUIREMENTS AND ISSUES WITH A CDS

Before discussing how to validate a chromatography data system, it is important to understand the regulatory requirements and their interpretation. The responsibility for the validation rests with the system or business process owner, but from experience it is seen that most do not understand fully the regulations they work under or the risk of mitigation strategies that need to be undertaken when validating a CDS.

The regulations and guidelines have a view of what is expected during the implementation and release of a CDS as well as what is expected when the system is operational and when it is retired. In general, the emphasis is concerned with generating the proof to demonstrate that the computerized system is accurate when validated and continues to be so when it is operational, and that there is sufficient proof of management awareness and control. To obtain proof of an action usually means that it must be documented, although the format of documentation (paper or electronic) is left open by all schemes.

The definition of Performance Qualification is *documented verification that the computer related system performs its functions in accordance with the computerized system specification while operating in its normal operating environment.*[5]

The major point to make is that the laboratory must test the CDS as it uses it and not in the way the supplier has tested it (i.e., in the laboratory's operating environment, using the laboratory's analytical methods, specifications and capacities, and using the laboratory's networks).

FDA Warning Letters and FDA 483 Observations Involving CDS

Some of the key FDA 483 observations and Warning Letters involving chromatography data systems are discussed in this section. This is not an all-inclusive list of noncompliances and the reader is encouraged to look at the FDA Web site to keep abreast of any changes in emphasis of inspections.

Gaines Chemical Company FDA 483 Observations

In December 1999[6] the FDA inspected the client server CDS operated in the QC laboratories of the company and made the following observations:

- The CDS had never been validated, and there no documentation to assure that the system can operate as intended.
- There was no change control.
- No security was enabled, and anyone could access the system.
- No record of system configuration.
- The application audit trail had been deliberately turned off by the staff.
- No documentation of calculations performed by the system.
- The application security could be bypassed by using Windows Explorer implying that files could be deleted outside of the application and with no record.
- Passwords consisted of four characters and never expired.

- When the system was operational anyone could access the application; the workstation had to be turned on otherwise data could not be acquired.
- There were no SOPs for the operation of the system.
- Backup and recovery was not demonstrated, and the storage conditions of backup tapes were not verified.

These observations reflect the situation in many small- to medium-sized companies that work in the regulated environment.

Glenwood FDA Warning Letter

In May 1999, Glenwood LLC received an FDA Warning Letter[7] that contained the following non-compliance relating to its CDS software:

> *Failure to validate the software programs, ______ and ____, that are used to run the laboratory HPLC equipment, during analysis of raw materials and finished products. The ____ software does not secure data from alterations, losses, or erasures. The software allows for overwriting of original data. There are no written procedures for the use of passwords, levels of access, or data back-up.*

Apart from the failure to validate the CDS application there was also a prominent issue with security and data integrity. Protection of electronic records created by any CDS is vitally important.

Gensia Scicor FDA Warning Letter

A Warning Letter sent to the company in July 1999[8] again reiterates the importance of protecting and preserving electronic records:

> *Failure to maintain laboratory records to include complete data derived from all tests necessary to assure compliance with established specifications and standards [21 CFR 211.194]. Specifically, your firm failed to properly maintain electronic files containing data secured in the course of tests from 20 HPLCs and 3 GCs. Additionally, no investigation was conducted by your company to determine the cause of missing data and no corrective measures were implemented to prevent the reoccurrence of this event.*

The critical problem was loss of electronic records coupled with a failure to investigate the problem to stop it happening again. Note the use of the predicate rule citation rather than 21 CFR Part 11.

Noramco FDA 483 Observations

A bulk chemical company was inspected in May 2001;[9] the FDA 483 observations highlight the detail of due diligence that any CDS validation requires in the 21 CFR Part 11 world. The observations are reproduced below:

> *There was no assurance that data acquired on the XXX chromatographic client-server data system was accurate, reliable, and reproducible for analyses of …*

- The CDS was not validated to ensure the system produced accurate and precise data.
- There was no documentation to show the system's ability to handle overload situations in an orderly fashion.
- There was no assurance of the program's behavior when working at its limit. Functional testing that includes volume and stress testing was not conducted to demonstrate the system's behavior.

- Confidential and unique user log-ins and passwords were not assigned to each analyst to ensure data authenticity and integrity. Each workstation had a single log-in name and password, which was shared with all users.
- There were no automatic computer generated time-stamped audit trails to ensure authenticity and integrity of analytical data that was acquired and processed with the CDS. Analyst's transactions were not documented to show whether the analytical data were modified, copied, or deleted.
- There was no documented evidence that the CDS was adequately configured and performed as intended.
- The firm did not have a system administrator who was responsible for system configuration and control of access to configuration tools that can modify or delete electronic records. System administrator permissions and rights were given to some QC analysts who were also responsible for analyzing samples.
- There was no control over how analysts interacted with analytical data on the system.
- The universal log-in and password system gives users rights and permissions to edit, modify, and delete data files. The system was not configured to deny analysts rights to directories and users did not have read/write access to analytical data on the system. Users could not only modify their records but all records on the server. There was no written documentation that established what limits and rights the IT groups assigned QC laboratory users.
- There was no documented evidence to show that the firm periodically restored analytical data from its tape backup medium to ensure that data files could be reconstructed and were not corrupted. IT personnel did not know how to reconstruct the graphic data on workstations and referred us to analysts in the laboratory to perform system administrator tasks.
- There was no documentation to show that analytical data on the chromatography network could not be altered or modified by authorized users of the corporate network. The networks are connected by a router, which enables data packets to move between networks. The chromatography network did not have capabilities for tracking and controlling the integrity of each sample throughout its retention period. There were no protocols that explained the logical security procedures in place to prevent unlimited and unauthorized access to chromatographic data files.

Key Inspection Learning Points

Some of the key learning points from these inspections and Warning Letters that we need to remember for the validation of any CDS include the following:

- The CDS must be validated and the scope of work should include documenting any customization or configuration of the system.
- Include in the PQ testing capacity tests for stress and overload conditions to comply with §211.63 ("adequate size"). The nature and extent of these capacity tests will vary, depending on the architecture of the individual CDS system and also how an individual laboratory uses it.
- Effective preservation of electronic records is vital to passing any inspection. Have a procedure, follow it, and retain documented evidence that it works. Use redundant hardware such as RAID disks (Redundant Array of Inexpensive Disks) and uninterruptible power supplies (UPS) as a first line of defense against electronic record loss.
- Change control is vital, and the process must include the IT department and the network.
- Security must be enabled, documented, and tested.

EXPLOITING THE BENEFITS OF ELECTRONIC SIGNATURES WITH A CDS

Rationale for Using Electronic Signatures

The Electronic Records; Electronic Signatures (21 CFR Part 11) final rule is an integrated regulation: subpart B (electronic records) has requirements for signing electronic records while subpart C (electronic signatures) has controls that are as important for ensuring the trustworthiness and reliability of electronic records as well as electronic signatures. Therefore, to use legacy systems in a hybrid mode is just a temporary solution before working completely electronically. In this section, the ways that the design of electronic signatures can be implemented into a chromatography data system (CDS) will be discussed.

A prerequisite for this approach to succeed is the need for any software to be technically compliant with the requirements of 21 CFR Part 11. Therefore, it is important that before implementing electronic signatures, the software used is technically compliant with the requirements of the regulation and the laboratory's interpretation of the regulation.

The key principle is that to implement electronic signatures on an existing paper-based process is not just a matter of electronically signing the calculated results. It requires a different philosophy and also requires a good understanding of the regulations that an organization has to comply with and also the business processes that will use electronic signatures.

It is unlikely that an organization will benefit implementing electronic signatures on an existing process unless it has been implemented to work electronically.[10]

To illustrate this principle, the interim results from a laboratory where electronic signatures have been designed into the process will be presented and discussed. The CDS is installed in a pharmaceutical quality control laboratory where the system is used for both raw material and finished product analysis; there are approximately 50 part-time users of the system. The current CDS version was not fully compliant with the technical requirements of 21 CFR Part 11 and was to be upgraded to a new compliant version of the software from the same vendor. Before the implementation of the new version, the current process was mapped and analyzed to see if there were any opportunities for improvement and to make effective use of electronic signatures.

There is also a LIMS that is operational in some of the sections within the laboratories. However, at the moment there is a mixture of both lab notebooks and a LIMS being used.

Mapping and Understanding the Current Process

The first task when considering implementing electronic signatures is to map the current process. This is relatively quick and the current laboratory high-level process is shown in Figure 20.4. We can see that there are parallel electronic and paper activities when chromatographic analysis is undertaken. For example, when a chromatograph is set up, a paper record (Lab Book) needs to be updated and checked. When results are calculated the report and chromatograms are printed out and the Lab Book updated and checked again.

It is important to analyze the current process. What are the process metrics? For example:

- How many samples are analyzed?
- What are the turn-around times?

Once this information has been obtained and the turn-around times analyzed, what were the reasons for fast or slow turn-arounds? Answers to these questions will give the information needed to start to improve the process and make it more effective and efficient.

The boundaries of the current version of the chromatography data system are also shown in Figure 20.4. In the current system the approval of results occurs outside of the chromatography data system on paper.

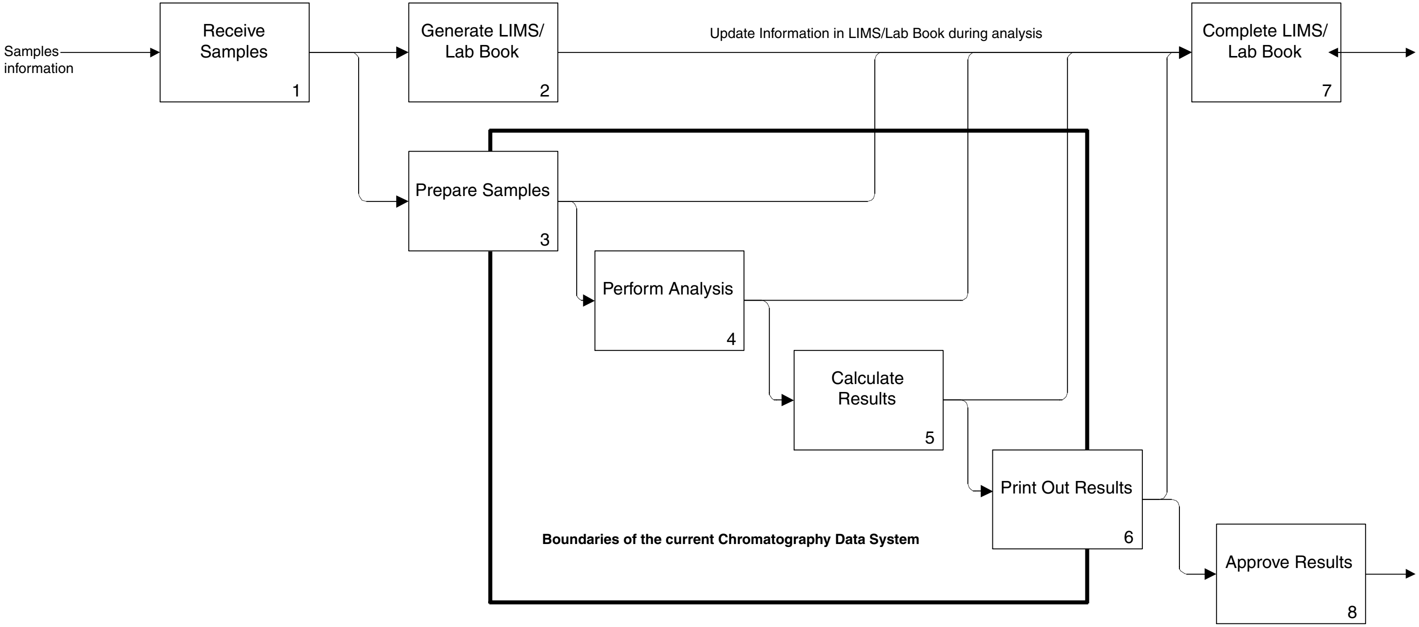


FIGURE 20.4 The Current Process Highlighting the Boundaries of the Current Version of the CDS.

Optimizing the Workflow to Use Electronic Signatures

Knowing the problems and improvement ideas from the analysis of the current ways of working, a new process can be designed to exploit the use of electronic signatures. It is important at this stage to ensure that the new process is compliant with 21 CFR Part 11 and any predicate rule requirements, and that the new version of the CDS can support the new process as well. For example, where in the process will signatures be used and where will identifications of actions be sufficient?

In the example, the redesigned process is shown in Figure 20.5. The main differences are:

- Elimination of the need to update the Lab Book for chromatographic analysis. This is a quick win that is estimated to save about 0.3 to 2.6 FTE (Full Time Equivalents or person years). This is independent of implementing electronic signatures in the CDS.
- Expansion of the scope of the CDS. In effect the approval of electronic records and calculated results takes place in the CDS and the printout is an option.
- Use of the CDS to carry out all calculations rather than using a calculator or spreadsheet. This streamlines the whole process for calculating, reviewing, and approving results.

The benefits of the process redesign when the CDS is linked to the LIMS would be in the region of 6 to 12 FTE. This is a surprising benefit but it does enable more capacity to be generated with the current laboratory resources. This is against a one-off cost of about 2 FTE for the process redesign, linking the system to the LIMS and validation of the CDS and the data link to LIMS.

LIFE-CYCLE APPROACH TO VALIDATION

GAMP Software Classification

A chromatography data system should, in the author's opinion, be classified as GAMP Category 4, and where customized macros or calculations are involved, GAMP Category 5. The rationale for this is that all commercial CDS applications need configuration at least to acquire data from the various chromatographs they are connected to or to control these instruments. Therefore the discussions here on the life cycle and the validation will be based around this premise of a GAMP 4–5 software application.

CDS Life Cycle

An International Standards Organisation (ISO) system development life cycle model is shown in Figure 20.6 and is depicted in the shape of a "V"; it is different from the GAMP "V-Model"[11] as the model below more accurately represents user and supplier relationships in regards to COTS products. The left-hand side of the V represents the design stages of the application, the bottom is the programming and the right-hand side is the testing stages of the life cycle. It is important to realize that there is a division between the user (above the line) and the supplier (below it). The qualification stages are condensed into a single stage under the control of the user below rather than presented as three distinct stages that never occur in practice.

This model can be used to generate the documentation that could be produced during the system development life cycle; the documents that could be produced are presented in Table 20.1 and the key ones are discussed in more detail in the next section. Taken together all of these documents will provide the validation package to support the contention that the chromatography data system is fit for purpose. Note that this is a suggested minimum list and may be extended. The extent that an individual validation differs from this approach will depend on the amount of regulatory risk that the organization or laboratory management wishes to carry after the validation.

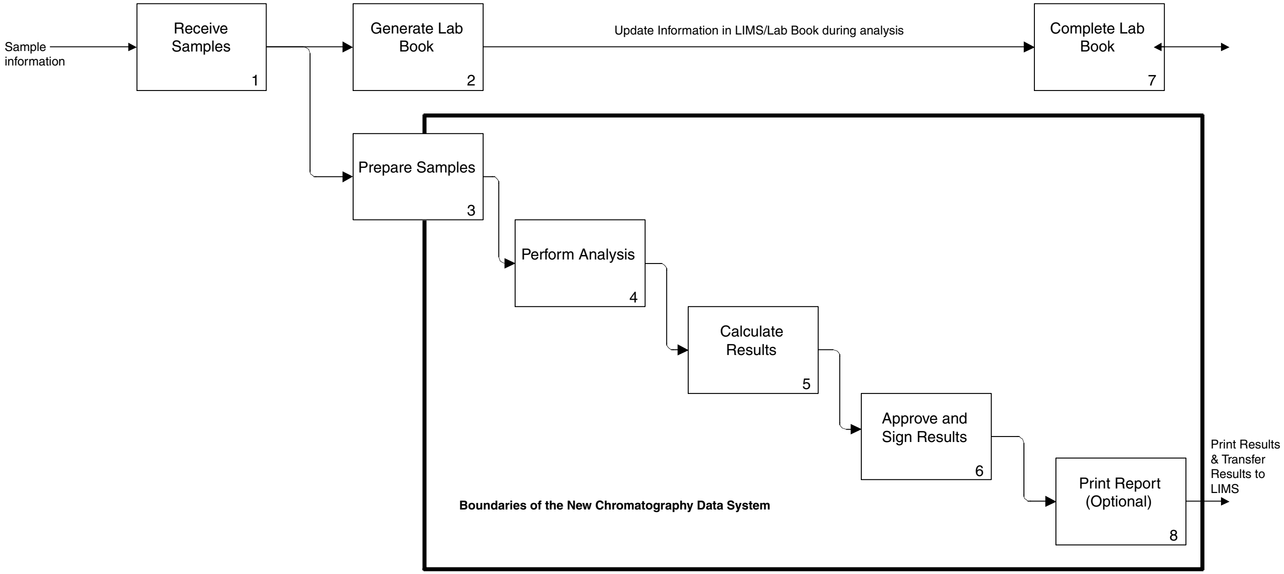


FIGURE 20.5 The Redesigned Process Highlighting the Extended Boundaries of the New Version of the CDS.

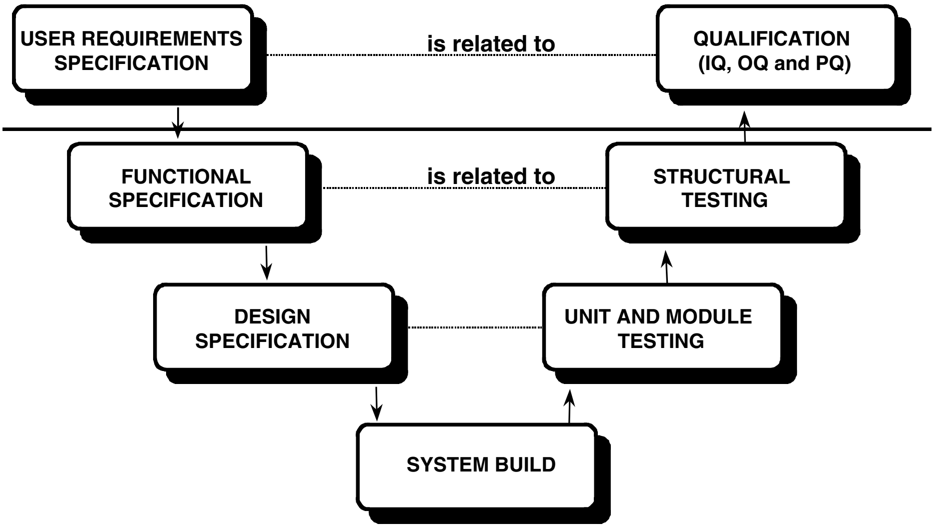


FIGURE 20.6 A System Development Life Cycle (SDLC) of a Chromatography Data System.

KEY VALIDATION DOCUMENTS FOR A CDS

The main validation documents will be presented in this section, typically in the order in which they are written and used in the system development life cycle. However, there are differences that will depend on individual circumstances.

Specifying the CDS Requirements

Defining the Basic CDS Functions

The first document in the validation is usually the URS as this can influence the validation strategy outlined in the validation plan. From Figure 20.6 it can be seen that the system requirements are related to the tests carried out in the performance qualification. Therefore, it is important to define the requirements for the basic functions of the CDS, the adequate size, 21 CFR Part 11 requirements, and consistent intended performance in the URS. Remember, the URS provides a laboratory with the predefined specifications to validate the CDS; without this document, validation cannot be conducted properly.

It is important to realize that the URS is a living document and must be updated as the system changes and evolves; for example, an URS should be written to select a system. It will then be reviewed and updated to reflect the selected CDS and version that will be validated and the functions specific to the laboratory where it will be installed.

The main elements in an URS should include the following major areas; each requirement must be individually numbered and written so that it can be tested as noted later in this section:

- Overall system requirements such as number of users, locations where the system will be used, and the instruments connected to the system; will terminal emulation be used?
- Compliance requirements from the predicate rule and 21 CFR Part 11 such as open or closed system definition, security and access configuration of the software application including user types, requirements for data integrity, time and date stamp requirements, and electronic signature requirements.
- Data system functions defined using the workflow outlined in Figure 20.1 but ensure that capacity requirements are defined such as maximum number of samples to be run, custom calculations, and reports for the initial implementation and roll-out, etc.

TABLE 20.1
Typical Documentation for a CDS Validation

Document Name	Outline Function in Validation
Validation Plan	• Documents the intent of the validation effort throughout the whole life cycle • Defines documentation for validation package • Defines roles and responsibilities of parties involved
Project Plan	• Outlines all tasks in the project • Allocates responsibilities for tasks to individuals or functional units • Several versions as progress is updated
User Requirements Specification (URS)	• Defines the functions that the CDS will undertake • Defines the scope, boundary and interfaces of the system • Defines the scope of tests for system evaluation and qualification
Risk Analysis and Traceability Matrix	• Prioritizing system requirements: mandatory and desirable • Classifying requirements as either critical or noncritical • Tracing testable requirements to specific PQ test scripts
System Selection Report	• Outlines the systems evaluated either on paper or in-house • Summarizes experience of evaluation testing • Outlines criteria for selecting chosen system
Supplier Audit Report & Supplier Quality Certificates	• Defines the quality of the software from supplier's perspective (certificates) • Confirms that quality procedures matches practice (audit report) • Confirms overall quality of the system before purchase
Purchase Order	• From supplier quotation selects software and peripherals to be ordered • Delivery note used to confirm actual delivery against purchase order • Defines the initial configuration items of the CDS
Installation Qualification (IQ)	• Installation of the components of the system by the supplier after approval • Testing of individual components • Documentation of the work carried out
Operational Qualification (OQ)	• Testing of the installed system • Use of an approved supplier's protocol or test scripts • Documentation of the work carried out
Performance Qualification (PQ) Test Plan	• Defines user testing on the system against the URS functions • Highlights features to test and those not to test • Outlines the assumptions, exclusions and limitations of approach
PQ Test Scripts	• Test script written to cover key functions defined in test plan • Scripts used to collect evidence and observations as testing is carried out • Documents any changes to test procedure and if test passed or failed
Written Procedures	• Procedures defined for users and system administrators including definition and validation of custom calculations, account management and definition of logical security • Procedures written for IT-related functions • Practice must match the procedure
User Training Material	• Initial material used to train super users and all users available • Refresher or advanced training documented • Training records updated accordingly
Validation Summary Report	• Summarizes the whole life cycle of the CDS • Discusses any deviations from validation plan and quality issues found • Management authorization to use the system

- IT Support requirements such as backup and recovery, off-line archive and restore.
- Interface requirements, such as will the CDS be a stand-alone system or will it interface with a LIMS and, if so, how?

System Specification Issues to Consider

Therefore, the first stage in the considerations for validating a CDS is to define all functions in a URS; for example, some or all of the following requirements will be included in the document:

- Data capture rates across all chromatographic techniques connected to the CDS. For example, with conventional chromatography with a run time in the order of 20 min, a data capture rate of 1 Hz is usually adequate. However, for capillary GC, 10 to 20 Hz may be appropriate, and for CE a higher rate may be required, depending on the overall migration time and the analyte peak shape.
- Several chromatographs may be linked into a collection workstation or an A/D unit. Consider that crosstalk (the interference from one channel to another) could be an issue if the A/D chip is multiplexed across two or more channels and/or total sampling capacity of the data collection and buffering unit.
- Has the maximum number of injections for an analytical run been defined? This is a critical component; if 100 vials are routinely injected in a run, the system cannot be tested with a run of only 10 samples as a user has not demonstrated adequate size. The specification must match the use of the system, including replicate injections.
- Some data systems will be configured to collect data from Diode Array Detectors (DAD). If this is required, especially to analyze product, then the data collection and analysis will need to be checked as part of the adequate size as some data files can be in the Mb range. The file delete option should not be enabled to protect the electronic records generated.
- Virtually all client server CDS systems will have a buffering capacity within their A/D or data collection units (if acquiring digital data from chromatographs via network interfaces). Therefore, part of the adequate size requirements must be the ability to capture and buffer data if the network is unavailable, followed by the successful transfer of data to the server when the network connection is reestablished.
- How many users will there be on the system at the same time, and will the system still perform its functions reliably? This number may be lower than the number of concurrent licensed users but it is still important to define this in the URS and test during the PQ. If the system becomes unreliable or unstable as the number of users increases, then the system owner cannot state that the system has adequate size or can perform as intended.

These are some of the considerations for each installation of a CDS. Once installed in a laboratory environment the CDS becomes unique. The network location, server support, operating systems, software patches, and laboratory configuration make each application different even if in just the smallest regard, and testing needs to confirm that the CDS works under each specific operating environment.

Documenting the System Requirements for Traceability

Although not mentioned in the regulations specifically, traceability of system requirements to the testing phase is important for any system including a CDS. Therefore, the way that system requirements are presented and managed is important.

It is all very well that the regulations state a user must define their requirements in a URS, but what does this mean in practice? Table 20.2 illustrates one way that capacity requirements can be documented. Note that each requirement is:

TABLE 20.2
How System Requirements for CDS Capacity Can Be Documented

Req. No.	Data System Feature Specification	Priority M/D
3.3.01	The CDS has the capacity to support 10 concurrent users from an expected user base of 40 users.	M
3.3.02	The CDS has the capacity to support concurrently 10 data acquisition channels from an expected 25 total number of channels.	M
3.3.03	The CDS has the capacity to support concurrently 10 digital acquisition channels from an expected 25 total number of channels.	D
3.3.04	The CDS has the capacity to control concurrently 10 instruments from an expected 20 total number of connected instruments.	M
3.3.05	The CDS has the capacity to simultaneously support all concurrent users, acquisition, and instrument connects while performing all operations such as reprocessing and reporting without loss of performance (maximum response time is <10 sec from sending the request).	M
3.3.06	The CDS has the capacity to hold 20 GB of data live on the system.	D

- Uniquely numbered.
- Written so that it can be tested, if required, in the PQ.
- Prioritized as either mandatory (M = essential for system performance) or desirable (D = nice to have and the system could be used without it). This prioritization can be used in risk analysis of the functions and also for tracing the requirements through the rest of the life cycle as will be discussed in a later section.

Each requirement must be written so that it can be tested if required. According to IEEE standard 1233,[12] a well-defined requirement must address capability, condition, and constraint. Remember, as shown in Figure 20.6, that the URS functions are related to the tests carried out in the qualification phase of the life cycle. If the requirements are not specified, how can they be tested? Further discussion on CDS user requirements can be found in McDowall.[13]

Review of the URS

Ideally, an independent group of users (persons not involved in writing the document) should evaluate the URS and challenge each requirement and any interfacing requirements for the chromatographs or any other computer applications. If any missing requirements or inconsistencies can be found at this stage, they are easy and inexpensive to correct. Therefore, the extra work in ensuring that the system requirement specification is correct are time and resources well spent; problems that can be rectified at this stage are far cheaper to solve than those identified later in the life cycle. When the system requirements specification is complete, outline selection tests can be generated to select a potential system then reused later in the life cycle during the PQ testing.

Validation Plan

The name for this document varies so much from laboratory to laboratory: validation plan, master validation plan or validation master plan or even quality plan. Regardless of what it is called in an organization, it should cover what steps will be taken to demonstrate the quality and compliance of the CDS in the laboratory. Ideally, it should be written as early in the process as possible to define the overall steps that are required and the documents to be produced from each (see Chapter 5 for more details).

System Selection

The purchase of a new CDS system should be a formal selection process to see if an application matches the main requirements of the URS. The outline tests can be used to screen and select the system; an in-house test can be an option if there is sufficient time and resources to do this. A selection report would be the outcome of this phase of the work and would form part of the supporting evidence for the CDS validation. This activity essentially links the URS to the supplier functional specification.

Supplier Audit

The majority of the system development life cycle for a commercial CDS will be undertaken by a third party, the supplier; this is shown in Figure 20.6 as all of the operations under the horizontal line. The European Union GMP Annex 11 on computerized systems states:[14]

> *The software is a critical component of a computerised system. The user of such software should take all reasonable steps to ensure that it has been produced in accordance with a system of Quality Assurance.*

The GAMP Guide[11] recommends that a supplier audit be undertaken to ensure that the software was developed in a quality manner.

The Supplier Audit should take place once the product has been selected and the purpose is simply to see if a quality management system (ISO 9000 or equivalent) is operated effectively. The evaluation and audit process is very important part of the life cycle as it ensures the design, build, and testing stages (which are under the control of the supplier) have been checked to ensure compliance with the regulations. The audit should be planned and cover items such as the design and programming phases, product testing and release, documentation and support; a report of the audit should be produced after the visit.

Many CDS suppliers are certified to ISO 9000 of some description and offer a certificate that the system conforms to their quality processes. This is fine but remember that there is no requirement for product quality in ISO 9000 and product warranties do not guarantee that the CDS is either fit for purpose or error free. If the system is critical to GxP operations it is better to consider a Supplier Audit.[15,16]

Requirements Traceability and Risk Assessment

The next stage in the process is to carry out a risk assessment of each function determined on if the function is business and/or regulatory risk critical (C) or not (N). Table 20.2 for URS now has two additional columns added, as shown in Table 20.3. This approach allows priority and risk to be assessed together.

Only those functions that are classified as both mandatory and critical are tested in the qualification phase of the validation.[18] Therefore in Table 20.3 functions 3.3.03 and 3.3.06 are not considered for testing as they do not meet the criteria. The remaining four requirements all constitute capacity requirements that can be combined together and tested under a single capacity test script, which in this example is called Test Script 05 (TS05). In this way, requirements are prioritized and classified for risk and the most critical one can be traced to the PQ test script.

Installation Qualification and Operational Qualification

Installation Qualification (IQ)

Establish an initial configuration baseline by taking an inventory of the whole system including hardware, software, and documentation. For networked CDS systems, the IQ should cover:

TABLE 20.3
Part of a Combined Risk and Analysis and Traceability Matrix for a CDS

Req. No.	Data System Feature Specification	Priority M/D	Risk N/C	Test
3.3.01	The CDS has the capacity to support 10 concurrent users from an expected user base of 40 users.	M	C	TS05
3.3.02	The CDS has the capacity to support concurrently 10 data acquisition channels from an expected 25 total number of channels.	M	C	TS05
3.3.03	The CDS has the capacity to support concurrently 10 digital acquisition channels from an expected 25 total number of channels.	D	N	—
3.3.04	The CDS has the capacity to control concurrently 10 instruments from an expected 20 total number of connected instruments.	M	C	TS05
3.3.05	The CDS has the capacity to simultaneously support all concurrent users, acquisition, and instrument connects while performing all operations such as reprocessing and reporting without loss of performance (maximum response time is <10 sec from sending the request).	M	C	TS05
3.3.06	The CDS has the capacity to hold 20 GB of data live on the system.	D	N	—

- Server (for data storage) installation by the IT department, server supplier, or manufacturer
- Installation of the A/D units or data collection servers to the corporate LAN
- Processing or data review workstations, either the IT department or contractors working on their behalf (typically, with an operating system configured to corporate requirements)
- Network connection of the workstations to the corporate LAN
- Installation of the CDS application software for data processing on the workstations
- Connection of the chromatographs to the A/D units or data collection servers

This work is typically supported by suppliers, system administrators from the laboratory, and the IT department, depending on the complexity of the configuration of the CDS. Planning is essential; retrospective documentation of any phase of this work is far more costly and time consuming.

Operational Qualification (OQ)

The operational qualification is carried out after the IQ and is intended to demonstrate that the application works the way the supplier says it will. Most suppliers will supply OQ scripts. These, of necessity, will only cover a subset of functions and will not be a substitute for the user acceptance tests or PQ tests.

What should be in an OQ? Here this depends on a supplier and the marketing approach to this "value-added" package. The purpose of an OQ is to show that the software and system works the way that the suppliers state it should. The amount of OQ testing can be relatively small, as suppliers have carried out the bulk of the work at their development sites. The main focus of OQ should be to test application-specific customization. Where there is a lot of laboratory customization of the application, e.g., chromatographic spectral library for specific user compounds, then a supplier's OQ package is of less or little help here.

Assess IQ and OQ Documentation

The regulations require that before execution the test protocols have to be approved by the QC/QA unit, and also that whatever is written in them needs to be scientifically sound (clause 160[17]). Here is an example of a Warning Letter sent by the FDA to Spolana,[18] a Czech company, in October 2000:

> *Furthermore, calibration data and results provided by an outside contractor were not checked, reviewed, and approved by a responsible Q.C. or Q.A. official.*

Never accept documentation from a supplier without evaluating and approving it. Check not only coverage of testing but also that test results are quantified (i.e., have supporting evidence), rather than solely relying on qualified (e.g., pass/fail) terms. Quantified results allow for subsequent review and independent evaluation of the test results. Further, it should be ensured that personnel involved with testing have been trained appropriately by checking documented evidence of training such as certificates is current at the time that the work was carried out.

Performance Qualification (PQ)

The PQ stages of the overall qualification of the system can be considered as the acceptance testing (this can also be called end-user testing), undertaken by the users and based upon the way that the system is used in a particular laboratory. Therefore, a CDS cannot be considered validated simply because another laboratory has validated the same software: the operations of two laboratories may differ markedly even within the same organization. The functions to be tested in the PQ must be based on the requirements defined in the URS and the numbering of individual requirements traced back to the system requirements. The main issue is how can users test software?

PQ Test Plan and Test Scripts

A documentation standard for the PQ test plan can be found in the IEEE standard 829-1998,[19] presented in Table 20.4. The key sections of a PQ test plan are the features to test and those that will not be tested and associated with the features to be tested are the written notes of the assumptions, exclusions, and limitations to the testing undertaken. The assumptions, exclusions, and limitations of the testing effort were recorded in the appropriate section of the qualification test plan to provide contemporaneous notes of why particular approaches were taken. This is very useful if an inspection occurs in the future, as there is a reference back to the rationale for the testing. It is also very important as no user can fully test a CDS or any other software application.

TABLE 20.4
Outline of a Test Plan from IEEE Standard 829-1998

1. Test plan identifier
2. Introduction
3. Test system/item
4. Features to be tested
5. Features not to be tested
6. Approach to be adopted
7. Pass/fail acceptance criteria for all features to be tested
8. Suspension criteria and resumption requirements
9. Test deliverables
10. Testing tasks
11. Environmental needs
12. Responsibilities
13. Staffing and training needs
14. Schedule (test order)
15. Risks and contingencies
16. Approvals

For example, the operating system was explicitly excluded from testing as the CDS application software implicitly tested this.

Release notes for the CDS application version being validated will document the known features or errors of the system. PQ tests carried out in any validation effort should not be designed to confirm the existence of known errors but to test how the system is used by the users on a day-to-day basis. If these or other software errors were found during the PQ testing, then the test scripts have space to record the fact and describe the steps that were taken to resolve the problem.

PQ Test Scripts

In the same IEEE standard[19] can be found the basis for the test documentation that is the heart of any PQ effort, i.e., the test script. In essence this document will:

- Outline one or more test procedures that are required to test the CDS functions
- Show how each test procedure consists of a number of test steps that define how the test will be carried out
- Define the expected results for each test step
- Give space to write the observed results and note if it the test step passes or fails when compared with the expected results
- Give a test log to highlight any deviations from the testing
- Collate in sections any documented evidence produced during the testing; this includes both paper and electronic documented evidence
- Define the acceptance criteria for each test procedure and indicate if the test passes or fails
- Give a test summary log collating the results of all testing
- Sign off on the test script, stating if the script has passed or failed

Testing Overview

One key point is that to ensure that the PQ stage progresses quickly, a test script should test as many functions as possible as simply as possible (with great coverage and simple design). Software testing has four main features, known as the 4Es:[20]

- Effective: demonstrating that the system tested meets both the defined system requirements and also finds errors
- Exemplary: tests more than one function simultaneously, where feasible
- Economical: tests are quick to design and quick to perform
- Evolvable: able to change to cope with new versions of the software and changes in the user interface

Write the Test Scripts

The number of PQ test scripts needed for a CDS typically falls in the range of 15 to 30 to provide adequate coverage for the important functions documented in the URS, depending on the complexity of the system.

Testing system functionality should consider:

- Data acquisition from the different types of chromatograph interfaced to the system
- Crosstalk of A/D converters[21]
- Calibration methods used within the laboratory: are they mathematically correct?
- Analyte calculation

- System suitability test parameters
- Reporting data
- Sample continuity
- Unavailability of the network: buffering of the A/D or data collection devices
- Remote processing over the network
- Data acquisition and data processing using a diode array detector (DAD) and/or dual wavelength detector
- Creation and management of DAD spectral libraries
- Custom calculations implement calculations on data
- Macros used to perform functions automatically
- System capacity tests, e.g., analyzing the largest expected number of samples in a batch, were incorporated within some test scripts to demonstrate that the system was capable of analyzing the actual sample volume that could be expected in the laboratory.
- Interfaces between the CDS and other software applications, e.g., LIMS

Testing should also consider any electronic record/signature requirements (e.g., 21 CFR Part 11) and other regulatory requirements:

- Preservation of electronic records, e.g., Backup and Recovery, Archive and Retrieve
- Data file integrity
- System security and access control, including between departments or remote sites
- Audit trail
- Date and time stamps
- Electronic signatures
- Identifying altered and invalid records

Outline Test Case Design

The considerations for designing stress and capacity tests for a CDS will be discussed here and will be based on the client–server architecture shown in Figure 20.3. Note that all aspects of the system that need to be tested must be defined in the system specification documentation.

VALIDATION SUMMARY REPORT

The validation summary report brings together all of the documentation collected throughout the whole of the life cycle and presents a recommendation for management approval when the system is validated. The emphasis is on using a summary report as a rapid and efficient means of presenting results as the detail is contained in the other documentation in the validation package (see Chapter 11 for more details).

TESTING CONSIDERATIONS

Analytical Run Capacity

First consider an analytical run and the capacity test considerations that will need to be evaluated. The maximum number of vials to be injected in a single run should be defined in the URS. Testing should include standards, samples, quality control, and blank reagents that are to be used as part of normal working procedures. A test should be designed to run the maximum samples including replicate injections.

Analogue to Digital Unit Capacity

Depending on the type of A/D unit this test can have one or more of the factors that will be discussed below:

- Crosstalk: If two or more channels are multiplexed through a single A/D chip, then a crosstalk test is recommended to see the impact of an overloaded signal on one channel impacts another.
- Data Acquisition Rate: compare the specified data acquisition rate for a data server to the data rate of chromatographs attached to the unit, including any diode array detectors.

In both or either of these instances the validation team may decide that the total data rate is close to the specification of the unit and test this to ensure that the A/D unit is not compromised during normal operation. If the data rate is far below specification, as an alternative path it may be decided that it is not necessary to test this, in which case a scientifically sound documented rationale is required.

Unavailability of the Network

There will be times when the network is unavailable and data will be buffered in the A/D unit or data server. It is important to ensure that this function works during the PQ. The worst-case example for the buffering will be defined in the URS and will be the number of injections with the longest run time. The run should be started, then the network is disconnected and the data accumulated in the A/D unit or acquisition unit until the end of the run when the network reconnected and the buffered data are transferred to the server. There should be no loss of data integrity in any of the buffered and transferred files if this test is to pass.

System Capacity

The capacity of the system needs to be tested in a way that reflects on the way the system will be used, and there are several approaches to take. For example, if there is a 30-user license, one of the simplest ways of assessing the capacity is to run all systems simultaneously. However, this will only test the data acquisition and transfer to the server via the network. As the A/D units buffer acquired data until transferred to the server this test will also implicitly evaluate the transfer with the network traffic at the time of the test. However, one of the main causes of performance degradation will be integration of data and this must also be included as part of any test of system capacity.

Logical Security and Access Control

While logical security appears, at first glance, to be a very mundane subject, the inclusion of this topic as a test is very important for regulatory reasons as it is explicitly stated in 21 CFR Part 11. Also, when explored in more depth, it provides a good example in the design of a test case.

The test design could consist of three basic components:

1. A test sequence where the incorrect account fails to gain access to the system
2. A single test case where the correct account and password gain access to the system
3. A test sequence where the correct account, because of minor modifications of the password, fails to gain access to the software

The important considerations in this test design are:

- Successful test cases are not just those that are designed to pass but also those that are designed to fail. Good test case design is a key success factor in the quality of validation efforts. Of the test cases, above 75% are designed to fail in order to demonstrate the effectiveness of the logical security of the system.
- The test relies on good practices to ensure that users change or are forced to change their passwords on a regular basis and that these are of reasonable length (minimum six to eight characters).

Other test case designs are defined below:

- Boundary test: the entry of valid data within the known range of a field, e.g., a pH value would only have acceptable values of 0 to 14.
- Stress test: entering data outside of designed limits, e.g., a pH value of 15.
- Predicted output: the function of the module to be tested being known, a known input should have a predicted output.
- Consistent operation: important tests of major functions should have repetition built into them to demonstrate that the operation of the system is reproducible.
- Common problems: both on the operational and support aspects of the computer system should be part of any validation plan, e.g., backup works and incorrect data inputs can be corrected in a compliant way with corresponding audit trail entries. The predictability of the system under these tests must generate confidence in the CDS operations (trustworthiness and reliability of electronic records and electronic signatures) and the IT support.

The format of the document and more detail of PQ testing, the articles on the retrospective, and prospective validations of CDS systems are recommended.[21,22]

Personnel and Training Records

All personnel involved with the selection, installation, operation, and use of a CDS should have training records to demonstrate that they are suitably qualified to carry out their functions and to maintain them. It is especially important to have training records and curricula vitae of installers and operators of a system as this is a particularly weak area, and a system can generate an observation for noncompliance.

Major suppliers of CDS will usually provide certificates of training for installation of the system and software. However, a major weak spot with many CDS that have the IT Department running the system is that they do not have training records or curricula vitae.

The types of personnel that could be involved in a validation are:

- Supplier staff: They were responsible for the installation and initial testing of the data system software, and left copies of their training certificates listing the products they were trained to work on. These were checked to confirm that they were current and covered the relevant products and then included in the validation package.
- System managers: Training in the use of the system and administration tasks was provided by the supplier and documented in the validation package.
- Users: These were either analytical chemists or technicians who had their initial training by the supplier staff to use the data system; this was documented in their training records.
- Consultants: Any consultants involved in aiding a validation effort must provide a curriculum vitae (résumé) and a written summary of skills to include in the validation package for the system.
- IT staff: Training records and job descriptions outline the combination of education, training, and skills that each member has.

Training records for CDS users are usually updated at the launch of a system but can lapse as the system becomes mature. To demonstrate operational control, training records need to be updated regularly, especially after software changes to the system. Error fixes do not usually require additional training; however, major enhancement or upgrade should trigger the consideration of additional training. The prudent laboratory would document the decision and the reasons not to offer additional training in this event.

To get the best out of the investment in a CDS, periodic retraining, refresher training, or even advanced training courses could be very useful for large or complex ones. Again, this additional training should be documented.

Service Level Agreement

In the case of outsourcing the support for the hardware platforms and network that run the chromatography data system software to the internal IT Department, a Service Level Agreement (SLA) has to be written. This SLA should cover procedures such as:

- Backup and recovery
- Archive and restore
- Storage and long-term archive of data
- Disaster recovery

This SLA will cover the minimum service levels agreed together with performance metrics so that they can be monitored for effectiveness.

System Documentation

Documentation

The documentation supplied with the CDS application or system (both hardware and software), user notes, and user standard operating procedures will not be discussed here as it is too specific and also depends upon the management approach in an individual laboratory. However, the importance of this system-specific documentation for validation should not be underestimated. Keeping this documentation current should be considered a vital part of ensuring the operational validation of any computerized system. The users should know where to find the current copies of documentation to enable them to do their job. The old versions of user SOPs, system, and user documentation should be archived.

Standard Operating Procedures (SOPs)

Standard Operating Procedures are required for the operation of both the CDS applications software and the system itself. SOPs are the main medium for formalizing procedures by describing the exact procedures to be followed to achieve a defined outcome. SOPs have the advantage that the same task is undertaken consistently, is done correctly, and nothing is omitted, and a written procedure means that new employees are trained faster.[23] The aim is to ensure a quality operation. Laboratory staff are used to working with SOPs — however, if a central computer group supports a large system, they may not be used to working with SOPs and even less ready to document their work. To provide a service to a regulated laboratory, a computer department must provide a suitably documented procedure. Indeed, this is a requirement under EU GMP Annex 11,[14] where a third-party supplier should have a documented operation.

According to Hambloch,[23] there is a minimum list of 12 SOPs required for the operation of a computer system in a regulated or accredited laboratory. These are:

- SOP on SOPs: This should describe the approach taken to the writing of SOPs within the functional group, the sections, who can authorize the procedure, description of the procedure, and distribution list.
- Description of responsibilities: The roles and responsibilities of staff supporting the computer system are defined.
- System description of hardware and change control procedures: This covers how the hardware components will be maintained (equivalent to the hardware configuration log) with the procedure to be adopted when the system configuration is changed.
- Preventative maintenance: The procedures for preventative maintenance of the hardware components are described.
- Prevention, detection, and correction of errors: These include the measures and procedures for finding, recording, and resolving errors in the system. This can be a complex SOP covering many different aspects of the system and may refer to sections of the technical manuals provided with the system. This SOP includes good housekeeping such as disk defragmentation or monitoring the space available on all disks.
- System boot and shutdown: This is a special SOP that should contain all the specific instructions for starting up and shutting down the system. This SOP may be required in an emergency and therefore should be written well and be easily available for use.
- Control of environmental conditions: For systems that require a controlled environment, an SOP that defines the acceptable ranges of temperature, humidity, and power supply. Other environmental considerations may be what to do in the case of electrostatic discharges, power surges, fire, and lightning strikes, or the use and maintenance of an uninterruptible power supply (UPS).
- Contingency plans and emergency operation: This is a disaster recovery plan and includes the use of alternative plans until the computer system has been recovered. It is important that any disaster recovery plan is tested and verified that it works before any disaster occurs.
- Backup and restore of data: Procedures are in place for backup of data and software programs and restoration of data to disk.
- Security: The logical (software) and physical security of the system is covered with procedures for setting up and maintaining security.
- Installation and update of software: Procedures are described to be undertaken before, during, and after installing software. This should start with the complete backup of all disks and then installation of the software and any testing and validation that may be required.
- Development and update of system software procedures: Software can be written to control the system or help execute functions. This SOP outlines the procedures for the creation, documentation, and modification of these procedures.

It is important to realize that the list above refers to a relatively large computer system that is run by a centralized IT group. Therefore, for smaller items of laboratory computer equipment the list should be reviewed for applicability and suitability. Where a system does not have the facility to store raw data, e.g., a disk drive, no SOP is required for backup and restore. The converse is also true; this is a generalized list of SOPs, and if there is a specialized application there may be the need for more SOPs than what appears above.

Manual and Automated Testing

Consider the following when deciding whether or not to use automated test tools:[20]

- Automated testing tools take longer to use the first time compared to manual testing.
- Expectation will exceed the delivery.

- To be economical the test suite must be reused many times.
- Automated tools are best used for regression testing (to see if operation of the software remains the same after change).
- Automated testing is not a substitute for manual testing.

Some suppliers offer automated tools for the IQ and OQ. These are likely to be useful as a quick means of establishing that a CDS has been installed correctly and the software functions as the supplier intended it to. The case for automated testing tools for PQ is less clear. In either case, the tool should be assessed to determine whether its use brings tangible benefits and meets GxP compliance requirements.

RETROSPECTIVE VALIDATION

The key difference between a prospective and a retrospective CDS validation is the gap and plan phase.[21]

Gap and Plan for Retrospective Validation

The Gap and Plan phase is an essential stage in the retrospective validation of any computerized system.

Collect Existing CDS Documentation

First, all of the existing documentation on the system must be collected; this could include items such as:

- Validation plan
- URS
- Documentation from the selection process
- Purchase order, packing lists
- Qualification tests and documentation
- PQ tests
- Training materials
- Operating manuals both in-house and from the supplier
- SOPs

In this example, the system was relatively new and most of the available documentation was retrieved as documentation was easily available. Furthermore, the personnel operating the system have been involved with the project from the start. This is in contrast with a system that may be much older where documentation may be nonexistent and personnel may have left the company or the company has reorganized or merged.

When all the documentation is collected, a list is made. This can be compared against the current regulations, industry guidelines, and the corporate validation policy. This generates a list of missing documents and defines the gap to be filled.

Review Existing Documents

Next, the existing documentation must be reviewed to see that each item is of suitable quality, coverage, and fitness for purpose. The mere existence of a document does not mean that its quality and coverage are good. Poor documents must be completed or otherwise discarded and new ones written that meet the current compliance requirements. For instance, if there is a current system requirements specification (URS), is it specific enough to allow qualification tests to be constructed? If a URS consists of one or two pages of general statements for a data system, such as:

- The data system performance must be fast, and
- The operation must be user friendly

this means that there is no firm requirement to allow a meaningful test to be constructed. The assessment of documents may result in more documents being added to the gap list.

Planning To Bridge the Gap

Once the gap has been defined, there must be a decision made to either write the key documents and fill the gap or for management to take the business risk not to write them, if they are not available. Time and resources must be included in this plan. This list of documents to be written and authorized by management is the output of the Gap and Plan phase. The Gap and Plan identified in a particular example included:[21]

- Validation plan
- Workflow analysis
- User requirements specification
- Test plan for the qualification of the system
- User test scripts (Performance Qualification)
- Change control and configuration management SOP
- System description

The process for the retrospective validation is to write these documents and execute any PQ testing as necessary.

MAINTAINING THE VALIDATION STATUS DURING OPERATIONAL LIFE

After operational release comes the most difficult part of computerized system validation — maintaining the validation status of the system throughout its whole operational life. Look at the challenges that will be faced when dealing with maintaining the validation of a CDS or, indeed, any system; some of the types of changes that will impact an operational CDS are:

- Software bugs, requiring associated fixes to be installed.
- Application software and operating system, plus any software tools or middleware used by the CDS, will need upgrading.
- Network improvements: changes in hardware, cabling, routers, and switches to cope with increased traffic and volume.
- Hardware changes: PCs and server upgrading or increase in memory, disk storage, etc.
- Interface to new applications, e.g., spreadsheets or laboratory information management systems (LIMS).
- Expansion or contraction of the system due to workload or reorganization.
- Environmental changes: moving or renovating laboratories.

All of these changes need to be controlled to maintain the validation status of the CDS. In addition, from a validation perspective, there are other factors that impact the system as well, such as:

- Problem reporting and resolution
- Software errors and maintenance
- Backup and recovery of data
- Archive and restoration of data
- Maintenance of hardware
- Disaster recovery (business continuity planning)
- Written procedures for all of the above

In this section, the number of measures will be discussed that need to be in place to maintain the validation status of a chromatography data system.

Change Control and Configuration Management

Changes will occur throughout the lifetime of the system from a variety of sources such as:

- Upgrades of the CDS software
- Upgrades of network and operating system software
- Changes to the hardware: additional memory, processor upgrade, disk increases, etc.
- Extension of the system for new users

This is the key item from the installation of the system to its retirement. Changes must be controlled. From a regulatory perspective, there are specific references to the control of change in both the OECD consensus document[24] and EU GMP regulations.[14]

Change control was implemented through an SOP that defined the procedure for change control. A change form was the means of requesting and assessing change:

- The change requested was described first by the submitter.
- The impact was assessed by the system managers and then approved or rejected by management.
- Changes that were approved were implemented, tested, and qualified before operational release.

The degree of revalidation work to be done was determined during the impact analysis. Changes that impacted the configuration (hardware, software, and documentation) were recorded in a configuration log maintained within a spreadsheet.

Operational Logbooks

To document the basic operations of the computer system, a number of logbooks are required. The term logbook is used flexibly in this context. The actual physical form that the information takes is not the issue. What is necessary is the information that is required to demonstrate that the procedure actually occurred. The physical form of the log can be a bound notebook, a pro forma sheet, a database, or anything else that records the information needed, as long as security and integrity of the records (paper or electronic) are maintained.

Backup Log

The aim of a backup log is to provide a written record of data backup, the location of duplicate copies of the system (operating system and application software programs), and the data held on the computer. The backup schedule for the disks can vary. In a larger system, the operating system and applications software will be separated from the data that are stored on separate disks. The data change on a fast timescale that reflects the progress of the samples through the laboratory and must be backed up more frequently. In contrast, the operating system and application programs change at a slower pace and are therefore more static; the backup schedule can reflect this.

For smaller systems, such as personal computers, the data and programs may be located on the same disk and partitioned by the directory structure. If the backup software is capable of performing selective backups, the comments in the paragraph above apply. However, if there is little sophistication, the whole disk may have to be backed up routinely. Again, for PC systems this may be an area to evaluate closely before buying. An alternative is a PC network where the programs and data are held on a central server and can be backed up more efficiently and effectively than stand-alone systems.

Some of the key questions to ask when determining the backup requirements for the CDS are:

- How long should the time between backups be? This can be answered by considering how much data the laboratory can afford to use. If it is up to a week (most unlikely), then the backups can be weekly, but typically it is daily. If criticality determines that no data can be lost, then shadowing or duplicate disks with RAID (Redundant Array of Inexpensive Disks) technology may be appropriate.
- Who is authorized to perform backups and who signs off the log? The laboratory manager in conjunction with the person responsible for the system should decide this. The authorization and any counter signature required should be defined in an SOP.
- When should duplicate copies be made for security of the data? This question is related to the security of data and programs. Duplicate copies should be part of the backup procedure at predetermined intervals. The duplicate copies should be stored in a separate location in case of a hazard to the computer and the original backups located nearby. Duplicate backups are also necessary to overcome problems reading the primary backup copies.

Problem Recording and Recovery

During the operation of a computer system, boot-up, backup, or other system functions, it is inevitable that errors occur. It is essential that these errors are recorded and the solutions also written down. Over time, this can provide a useful historical record to the operation of the computer system and the location of any problem areas in the basic operation.

Areas where this may be the case may be in peripherals where a print queue has stalled. This is relatively minor; however, there may be cases where the application fails due to a previously undetected error. In the latter case, there is a need to link the error resolution to the change control system.

Software Error Logging and Resolution

As it is impossible to completely test all of the pathways through CDS software or any software,[25] it is inevitable that errors will occur during the operation of the system. These must be recorded and tracked until there is a resolution. The key elements of this process are to record the error, notify the support group (in-house or supplier), classify the problem, and identify a way to resolve it.

Not all reported problems of a CDS will be resolved, they might be minor and have no fundamental effect on the operation of the system and may not even be fixed. Alternatively, a work around may be required which should be documented and sometimes even retraining may be necessary. Other errors may be fatal or major, which means that the system cannot be used until fixed. In these cases, the revalidation policy will be triggered and the fix tested and validated before the CDS can be operational again.

Maintenance Records

All quality systems need to demonstrate that the equipment used is properly maintained and in some instances calibrated. Computers are no exception to this. Therefore, records of the maintenance of the CDS need to be set up and updated in line with the work carried out on it. The main emphasis of the maintenance records is toward the physical components of a system: hardware, networking, and peripherals. The software maintenance is covered under the error logging system described above.

If the hardware has a preventative maintenance contract, the service records after each call should be placed in a file to create a historical record. Also, any additional problems that occurring that require maintenance will be recorded in the system log and cross-referenced to the appropriate record.

Many smaller computer systems have few, if any, preventative maintenance requirements but this does not absolve the laboratory from keeping records of the maintenance of the system. If a fault occurs that requires a service engineer to visit, then this must be recorded as well.

On sites where maintenance of personal computers is done centrally for reasons of cost or convenience, maintenance records may be held centrally. The remit of the central maintenance group may cover all areas of a site or organization including regulated or accredited as well as nonaccredited groups. It is important for the central maintenance group to maintain records sufficient to demonstrate to an inspector of the work they undertake. As defined in EU GMP Annex 11,[14] the third party undertaking this work should have a service agreement and also have the curriculum vitae of its service personnel available and up to date.

Disaster Recovery

Good computing practices require that a documented and tested disaster recovery plan must be available for all major computerized systems. It rarely is. Failure to have a disaster recovery plan places the data and information stored by major systems at risk, the ultimate losers being the workers in the laboratory and the organization.

Disaster recovery is usually forgotten, or not considered, as "it will never happen to me." The recovery plan should have several shades of disaster documented. After loss of a disk drive for any reason from simple equipment failure through to the complete loss of the computer room or building due to fire or natural disaster, how will data be restored from tape or backup store and then updated with data not on backup?

Once the plans have been formulated, they should be tested and documented to see if they work. Failure to test the recovery plan will give a false sense of security and compound any disaster.

Revalidation Criteria

Any change to a CDS should trigger consideration if revalidation of the system is required. Note the use of the word "consider." There is usually a knee-jerk reaction that any change means that the whole system should be revalidated. One should take a more objective evaluation of the change and its impact before deciding if full revalidation is necessary.

Firstly, if revalidation is necessary, to what extent is it required to test a software unit, module, or the whole system? There may even be instances where no revalidation would be necessary after a change. However, the decision must be documented together with the rationale for it.

Therefore, a procedure is required to evaluate the impact of any change to a system and act accordingly. One way to evaluate a change is to review the impact that it would make to data accuracy, security, and integrity. This will give an indication of the impact of the change on the system and the areas of the application affected. This allows the revalidation effort to target the change being made.

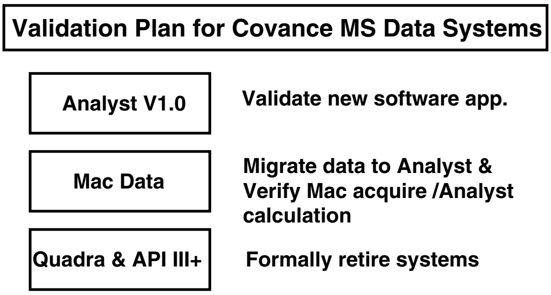

Validation Plan for Covance MS Data Systems
Analyst V1.0
Validate new software app.
Mac Data
Migrate data to Analyst & Verify Mac acquire /Analyst calculation
Quadra & API III+
Formally retire systems

FIGURE 20.7 Overview of the Whole Mass Spectrometry Validation, Data Migration, and System Retirement Project.

SYSTEM RETIREMENT

System retirement occurs at the end of the life cycle of any computerized system. However, there are little or no directly stated regulatory requirements for formal system retirement or general advice on how to undertake the task. System retirement is typically considered when a CDS including file formats, operating systems, and hardware platforms becomes obsolete.

Retirement typically consists of three strands of work that can be collectively managed under a single validation plan as shown in Figure 20.7. These strands of work are described below in relation to an actual project:

1. Prospective validation of the new application software (Analyst version 1.0) and qualification of new instruments associated with them
2. Validation of the migration of electronic records generated using MassChrom software on the Macintosh systems to the new Analyst NT environment, as well as data acquisition on some Macintosh platforms with interpretation using Analyst software
3. Formal retirement of obsolete mass spectrometry and Macintosh computer hardware

Overview of the Bioanalytical Mass Spectrometry Systems

The mass spectrometry equipment, current software options, and computing environment within Bioanalytical Services are presented below and summarized in Table 20.5 and Figure 20.8.

Mass Spectrometry Equipment

There are three main models of mass spectrometer currently operating in the Bioanalytical Services Department: API models III+, 365, and 3000. Of these, the API III+ is obsolete since the Macintosh PC used to run the software is no longer in production. Therefore, the three systems using the API III+ mass spectrometer will be formally retired and only the API 365 and 3000 models will be used thereafter.

Data Acquisition and Processing Software Applications

The MassChrom mass spectrometer software currently used in the Department is a combination of data acquisition software (three versions of RAD and sample control) and data processing software (two versions) that operates on the Macintosh plus the Analyst software designed for the Windows NT environment. The RAD and MacQuan software running on the Macintosh Quadra will be retired under the work described in this chapter.

TABLE 20.5
Data Processing Options Available in Bioanalytical Services Department

Mass Spectrometry Instrumentation	Computing Hardware	Operating System	Data Acquisition Software	MS Quantification Software
API III+	Mac Quadra	Mac OS	RAD 2.6	MacQuan 1.4
API III+	Mac Quadra	Mac OS	RAD 2.6	TurboQuan 1.0
API 365	Power Mac	Mac OS	Sample Control 1.3	MacQuan 1.4
API 365	Power Mac	Mac OS	Sample Control 1.4	MacQuan 1.4
API 365	Power Mac	Mac OS	Sample Control 1.4	TurboQuan 1.0
API 3000	Dell PC	Windows NT	Analyst v1.0	Analyst v1.0

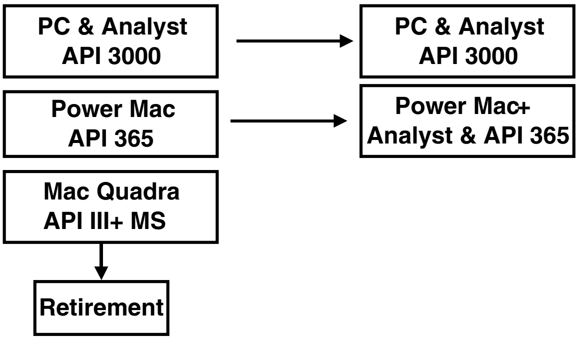


FIGURE 20.8 Overview of Mass Spectrometry Equipment and Data Systems.

A mixed environment will be operated for a transition period where data are acquired by sample control running on a Macintosh but all data processing and quantification are run on the Analyst. In the future, after retirement of all Macintosh computers, there will be an environment that is only Analyst running on Windows NT.

Computing Environments

The existing environment was Macintosh with mass spectrometry being downloaded to a server after it had been acquired. Introduction of the Analyst has started a migration to an NT operating environment that will continue after the completion of the data migration outlined here.

Differences between the New and Legacy CDS Systems

It is vitally important to understand the differences between the two environments before progressing further with any data migration. Covered here are the major differences between the two systems and their impact on the data migration. Essentially, the problem is that we have incompatible

- Hardware
- Operating system
- Application software
- Data file formats
- Application design philosophies

These differences will be discussed below. However, the bottom line is that data file conversion is essential for the data migration to succeed.

Computing Platform Differences

The Macintosh and Intel hardware computing platform and operating system software are essentially incompatible. An emulator is needed to run Windows software on a Macintosh, but there is no corresponding emulator for the Macintosh in a Windows environment that will run the software and be supported by the supplier.

Raw Data File Format Differences

The file formats for the chromatograms produced by the same instrument in the two environments are completely different. The Macintosh uses a different file format whereas the Analyst uses WIFF (Waveform Interchange File Format) file format that can have either single or multiple WIFF files. For the work described here, the use of only multiple WIFF files was evaluated.

Meta Data File Format Differences

The MassChrom software requires three files to set up and acquire data: the Method, State, and Experiment files. The method and experiment files are used to set up and acquire mass spectrometer data and the experiment and state files used to monitor the performance of the mass spectrometer itself.

In contrast, there are just two such files used within the Analyst: data acquisition method (DAM) and instrument file format (INS) files. The mapping of the MassChrom and Analyst files is not one to one: parameters in the experiment file are split between the INS and DAM files on the Analyst application.

Design Philosophy of the Macintosh and NT Software Applications

Although the software running on the two platforms can control the same mass spectrometry instruments, their designs are very different. The MassChrom software was designed in the early 1990s for operators with mass spectrometry training; the terminology and instrument setup within the applications are specialist for trained mass spectrometrists.

Over time, the instrument has been used more widely by chromatographers, and the Analyst software is a response to this as the operation of the application is simpler and uses chromatographic terms more than mass spectrometry ones. This difference in design philosophy is a complicating factor for the data migration as terms have to be mapped between the applications, which we will describe later in this chapter.

Generic Data Migration and System Retirement Process

A generic seven-step process, shown in Figure 20.9, describes system retirement and migration of data. Each stage will be described in overview and is a summary of the work described by McDowall.[26]

Step 1: Inventory of the System

Identify the scope and boundaries of the system and the departments that use the system. Part of this is necessary because the system may be spread across buildings and even networks. The latter is an issue as it can complicate the initial work since data spread over different networks will have to be collated to find out the data volumes and projects/studies involved.

Step 2: Carrying Out a Risk Assessment

How critical is the system? This determines the level of regulatory risk and data criticality and is used to determine the detail required in the remainder of the process.

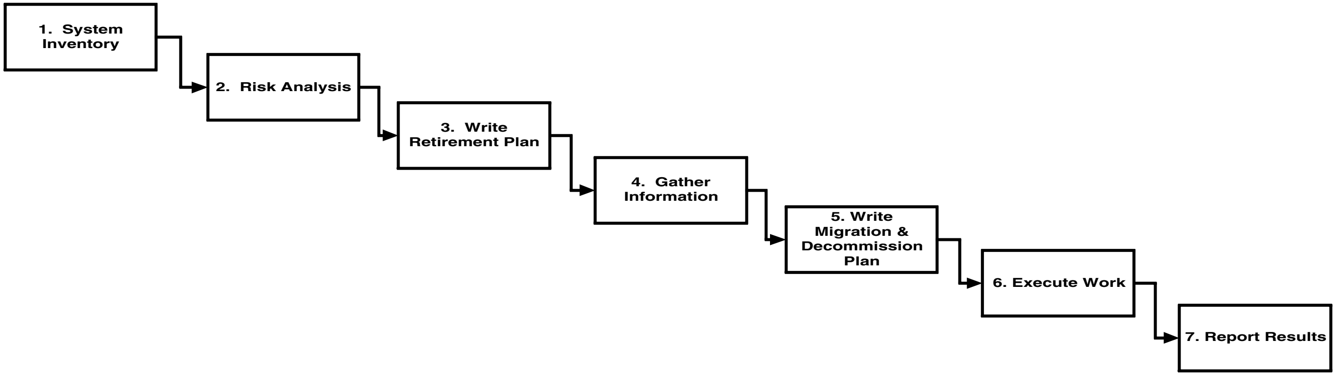


FIGURE 20.9 Generic Process for Data Migration and System Retirement.

Step 3: Writing the Retirement Plan

Using the data generated from Step 1, the plan covers:

- Scope and boundaries of the chromatography data systems
- Roles and responsibilities
- Outline project plan
- Process of system retirement
- Process of data migration

Step 4: Detailed Information Gathering

Collect details of the computer hardware including any specialized devices, the software, and the documentation associated with the system, as well as the data. The data need to be identified in detail, for example, how many tapes are involved (if long-term storage is on tape) and what data relating to which samples are on a specific tape.

Step 5: System Decommissioning and Data Migration Plan

This document is a detailed presentation of the approach being undertaken on the system and describes the roles and responsibilities of people involved in the work, the systems, the data to migrate, the test scripts needed, and what each test script will contain to document the process.

Step 6: Executing Work and Document Activities

Following the tasks described in the decommissioning plan, the data retirement will start first to be followed by the system retirement. Write any scripts needed to check and document the correctness of the data transfer; this is a critical stage in generating confidence in the process. Once the data have been successfully migrated and archived, the hardware can be withdrawn from service and either disposed or reused as appropriate. Again, this will be documented as the process continues.

Step 7: Writing Retirement and Migration Report

This is simply a summary of the work that was undertaken with a description of any deviations from the plan and a discussion of their impact. The data migration together with any validation tests applied will be described, and management will sign off the report.

Data Migration Strategy

The option for data migration is to assess if it is technically feasible to migrate data. The supplier of the mass spectrometry software systems (Applied Biosystems/MDS Sciex) provides conversion programs to allow a user to migrate electronic records from the Macintosh to the Analyst system. Conversion is necessary as the file formats are completely different between the Macintosh and Windows NT environments.

Supplier-Supplied Data Conversion Utilities

Three API File Converter programs were supplied for the conversion of the Macintosh format data and meta data files by Applied Biosystems, the software supplier. These are

- File Translator — Data file conversion program that takes Macintosh formatted data files and converts them to single or multiple Analyst format files (WIFF).

- InstFileGenerator — Instrument file conversion program that combines Macintosh state and calibration files and generates an Analyst instrument file (INS file).
- ExptFile Converter — Experiment file conversion program that combines a Macintosh state file and a Macintosh experiment file and generates an Analyst data acquisition method file (DAM).

It is technically feasible to convert the data and migrate them into the Windows NT environment. The question now becomes "Are all data converted or are files converted on an as needed basis"? The data volume involved is in the range of 100 to 200 GB of data.

Limitation of the Data Conversion Utilities

These utilities have a number of limitations that were not apparent during the early stages of this work:

- They only work on a PowerMac, therefore the objective of retiring all Macintosh computers cannot be realized as at least one is required to run the data conversion utilities.
- The utilities cannot convert RAD version 2.6 files. Only the chromatograms can be converted but the experiment, method, and state files cannot and the data contained therein must be manually input into the Analyst. Therefore, in the case of data collected under RAD version 2.6, the requirements of 21 CFR Part 11 for ready replay of data cannot be met.
- A further limitation of the utilities became apparent during the data migration in that the original baselines were not transferred and new baselines were redrawn with the new system.

Data Migration Options

There are essentially two options for the migration of the data from the MassChrom environment:

- Convert all data into the new data format now.
- Convert selected data on an "as-needed" basis.

The second option was chosen for a number of reasons including the time and cost of conversion. However, two main issues arise from this approach:

- The laboratory is totally reliant on the suppliers' conversion utilities and their continued maintenance of them over time.
- The conversion utilities must be tested to confirm that they continue to operate as expected after every software upgrade.

Evolution of the Data Migration Design

It is important to understand that a data migration project requires a full understanding of the problem. Therefore, this section of this chapter is intended to provide a measure of the evolution of the data migration project, as the understanding of the extent of the issues involved increases.

Initially, a single test script under the Analyst validation was envisioned. However, as the complexity of the MassChrom software versions was understood, a data migration and system retirement test plan was required to explain the overall strategy with five test scripts. Further information gathering revealed more complexity, and the number of test scripts rose to 10.

A complicating factor was that each combination of MassChrom software had been validated on its own but comparison of data across all combinations of the software had not been undertaken as this is not normally considered as part of a normal validation study. Therefore, to ensure a comprehensive approach to the data migration, an evaluation of data acquired by all MassChrom software versions was required to ensure that no regulatory questions remained with the data migration. This approach increased the number of test scripts to 16.

Detailed design of the test scripts enabled a better way of testing to be developed and this reduced the number of test scripts down to 12, of which 3 were for retirement of the obsolete mass spectrometry systems.

Design of the Overall Data Migration and System Retirement

As there was no systematic study of results from all MassChrom software combinations, it was decided to evaluate results from all MassChrom software combinations vs. Analyst. In addition, all future data acquisition and analysis configurations were also evaluated to give a comprehensive approach to the data migration and to find out if there were any problems with the proposed approach.

Standardized Study Design: as the Analyst version 1.0 had been comprehensively validated to include some 21 CFR Part 11 requirements,[27] we decided that this was the standard to which all data migrations would be measured. A series of 32 sample vials was prepared containing standard and blank solutions that represented a standard curve and a series of unknown samples. This standard set of samples was injected into a mass spectrometer controlled by Analyst software, and this set of acquired data was considered the gold standard against which all data migration results were measured.

The standard sample set was then injected into different mass spectrometers controlled by the different software versions, the data analyzed, and then migrated into the Analyst using the supplier's utilities and reprocessed. Therefore we have a situation where the same sample solutions have been acquired and analyzed by the various MassChrom software versions and then migrated into the Analyst and reprocessed and compared against the results of the same samples acquired and processed directly by the Analyst.

In addition, historic study data acquired under MassChrom and archived on tape would be restored to the server, all electronic records then migrated to Analyst, and the results compared.

All test scripts were written, technically reviewed, and then approved by the Quality Assurance Unit before execution.

Data Migration: Key Results

In this section, we present a selective review of the key results obtained from the data migration to illustrate the issues in a data migration project. Four areas will be discussed in light of the migration issues we found, the acceptance criteria that we set, and the results that were obtained after the migration.

Retention Time

Retention time is a fundamental chromatographic parameter and is the time that the chromatographic column retains an analyte. In setting the acceptance criteria, the discussions centered on the conversion of time and we determined that the retention times should be within 1% of the original value, especially as the applications were both from the same software supplier. The acceptance criterion of ±1% was determined on the basis of a 3-min chromatographic run time and likely differences in the peak integration algorithm that may impact the peak apex in the migrated data.

Review of the migrated data showed that there was a large discrepancy between original and migrated results:

- 1.07 (MassChrom)
- 1.12 (Analyst)

Thus, the migration of this parameter appeared to fail against the acceptance criteria. By examining the data more closely, it can be found that the data formats between the two are different: minutes and seconds (MassChrom) and digital minutes (Analyst). Therefore, we are not comparing like with like and the MassChrom values must be converted to digital minutes to make the comparison valid.

Therefore, all MassChrom retention time values must be collated, converted to seconds, and then divided by 60 before comparing with the corresponding Analyst values. After this conversion, the converted retention times were similar to the original results within rounding errors in the second decimal place. In retrospect, the acceptance criteria could have been set within ±0.5%.

Instrument Control Parameters

As mentioned earlier in this chapter, there are design differences between the two software applications. These are manifested in the instrument control parameters which can have a major impact — or none — on the data migration. This area requires a thorough knowledge of the two applications; failure to do this means that the migration will be flawed due to lack of knowledge.

For example, some parameters are the same in both applications and present no problem in the data migration project. An example of this parameter is the scan type such as Multiple Reaction Monitoring (MRM) that is present in both applications. Therefore, the migration is relatively straightforward and the acceptance criteria that are set are an exact match.

However, a parameter can have different terms in the two applications but still refer to the same measurement, and this starts to complicate the migration as the parameters must be mapped. A typical example is the QO voltage (MassChrom) that is equivalent to the Entrance Potential (Analyst) and illustrates the design differences between the two applications. The acceptance criteria in this instance were set to the nearest volt ignoring differences in the decimal values (e.g., 3.0 vs. 3.00); the rationale was that we did not know how numbers were held in either system and that there might be rounding errors involved in the migration.

Adding further complexity to the migration is the situation where a parameter in Analyst has to be derived from two parameters in MassChrom. Thus, the collision cell exit potential value in the Analyst can only be calculated by subtracting the potential for the Rod Offset Potential Q2 from the Inter Quad Lens 3 potential. The acceptance criteria for this were the same as the last example (the nearest volt ignoring differences in decimal values).

Again, this reiterates the need to fully understand the two applications before beginning a data migration. The acceptance criteria for all the instrument parameters monitored in the migration were documented in the appropriate test scripts that were reviewed and approved before the migration.

Integration Algorithms and Calculated Results

When migrating data from one application to another there are a number of results that can be compared. In the example of mass spectrometry, these include:

- Analyte peak heights or areas
- Drug: internal standard ratios
- Calibration curve parameters
- Calculated results from unknown samples
- Back calculated standards

TABLE 20.6
Comparison Peak Areas from MassChrom with the Same Data Converted and Calculated by Analyst

Analyte Standard Concentration	MassChrom Peak Area	Analyst Peak Area
10 ng/ml	4,366	4,544
20 ng/ml	7,851	8,383
50 ng/ml	22,867	23,160
100 ng/ml	45,204	47,667
500 ng/ml	205,054	205,822
1000 ng/ml	399,296	401,330

TABLE 20.7
Calibration Curve Parameters Calculated by MassChrom and Analyst

Calibration Parameter	MassChrom	Analyst
Slope	0.00365	0.00362
Intercept	0.00127	–0.00036
Regression Coefficient	0.99726	0.9960

As the integration algorithms were different between the two applications, an early decision in the migration was taken to avoid using the peak area calculations as a comparator between the two systems:[26]

> *What we need to consider here is, when the data files are in the new data system are similar results … obtained? Expect to see some differences between the two systems. The main issue is whether it matters from a scientific perspective … For instance, if the final calculated result means that a sample that was previously acceptable is now out of specification, the impact of this needs to be assessed …*

This situation was confirmed from the first set of converted data shown in Table 20.6.

Note that the data at first glance are very comparable; however, on closer inspection, the Analyst data were consistently higher. Upon further investigation into the issue, it was discovered that the electronic records were migrated without the original baselines set in the Macintosh environment. However, if the migrated data are auto-processed (baselines were automatically placed using preset criteria) with manually input data from the original MassChrom methods, then similar analyte results are obtained.[26]

Calibration curve parameters for original and converted data are shown in Table 20.7; the values are equivalent. However, the criteria chosen for acceptance of the data migration were based on the calculated results. As the analysis is based upon a comparative method of analysis (chromatography), the results were deemed the best way of evaluating if the conversion was successful. The key question is, would the same decision be taken on the data? Therefore, a regression line of the MassChrom vs. the Analyst across all concentrations should have a correlation coefficient close to 1 if the results were the same by both methods. These data are shown in Figure 20.10.

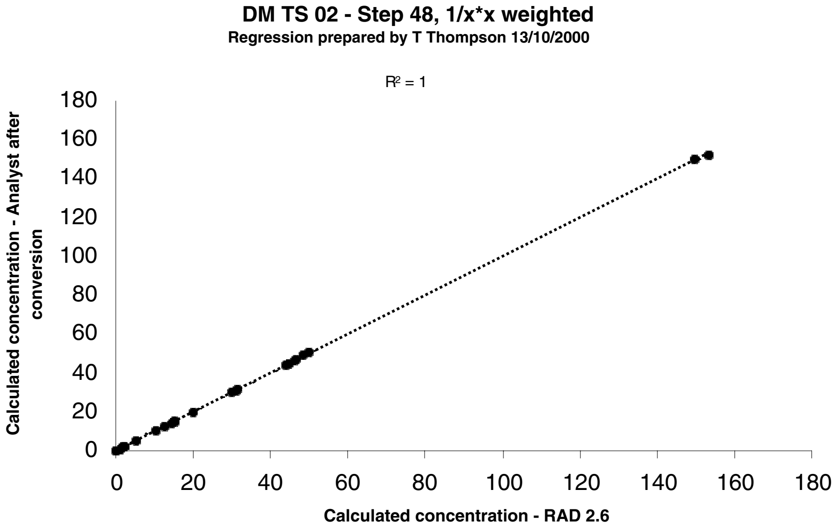


FIGURE 20.10 Regression Analysis of Macintosh and Analyst Data Showing Equivalent Results Obtained from the Data Migration.

History Logs

MassChrom does not have an audit trail associated with the data, but it does have a history log associated with each data file that notes data and time of creation and changes made to the data. The entries created in the Macintosh environment were migrated to the Analyst environment exactly and were updated following change of a baseline or similar events.

Data Migration from Archive

The final segment of the data migration was to take an archived study, restore the data into the Macintosh environment, reprocess them, and then migrate them into Analyst environment for further processing. The two sets of calculated results were compared as above and the results were equivalent.

Data Migration Summary

Data migration from one platform and environment to another was accomplished using the utilities supplied by the supplier. For most cases the tools were successful, however the inability to migrate the previously fitted baselines is a major flaw that prevents the ready replay of data. If data are auto-processed, equivalent results are obtained. A key for success is the technical understanding of both environments so that parameters can be mapped between the two.

CDS System Retirement

Under the data migration and system retirement test plan outlined above, three test scripts were written for the formal retirement of the obsolete mass spectrometry systems. As these systems were essentially the same configuration, the test scripts were identical and just varied with the name and identification of an individual system. The process flow is shown in Figure 20.11; the involvement of management support in the process is a key factor.

The essence of each retirement test script was a pro forma checklist for the systematic collection and confirmation of activities involved in retirement of an instrument. Sections within each test script for the retirement of a system included:

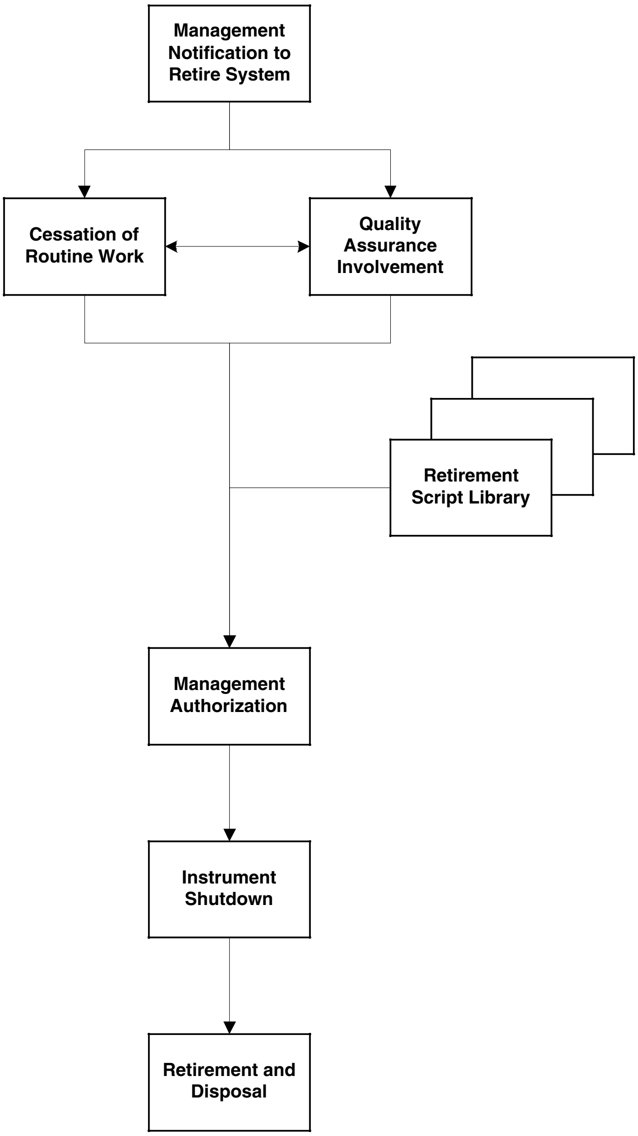

Management Notification to Retire System
Cessation of Routine Work
Quality Assurance Involvement
Retirement Script Library
Management Authorization
Instrument Shutdown
Retirement and Disposal

FIGURE 20.11 Process Flow for System Retirement.

- Component inventory: All components of the system including the computer, network connections, software, and MS instruments are listed in the test script (this is supplied from the system inventory and information gathering stages of the process outlined in Figure 20.9).
- Data: It was confirmed that all data have been backed up and then copied across to a server, and have not been corrupted. This is followed by deletion of the data on the hard drive.
- Computer: The computer was disconnected from the network and the IT department was informed that the socket (IP address) could be reallocated if required. The hard drive of the Macintosh was reformatted before the computer was removed from site to ensure that no confidential data remained.

- Mass Spectrometer: There were several stages to this where it was confirmed that the instrument was biologically and radiologically decontaminated before allowing it to be removed from the site.
- Finance: The fixed asset numbers and identities of the components retired were passed to the Finance Department to update the asset register and to show the item was decommissioned.

Each section in the retirement test script has the expected results and documented evidence, as well as acceptance criteria; the script was completed by management review of the overall retirement.

Data Migration and Retirement Summary

When considering a data migration and system retirement project, the following approaches are suggested:

- Think first and understand the complexity of the whole system and the technical problems associated with it. This is important and, while it will slow the overall project initially, will enable the actual work to proceed more smoothly than would be the case if this step were omitted.
- The problem is unlikely to be solved at the first attempt, therefore adopt an evolutionary approach to the issues. This is illustrated in this chapter where the number of scripts rose from 1 to a final 12.
- Do not rush into actions. Draw up a data migration plan and then do nothing for at least a week so that the plan can be reviewed and refined: is it feasible and what is the regulatory risk?
- Be practical and flexible; it is not unusual to find unexpected issues when least expected. The better prepared you are, the less likely these issues will be major and affect the data migration adversely.
- Large volumes of data will be produced when validating the data migration process; plan well in advance how to capture and handle these data. These data will be both paper and electronic files, manage both well and have file-naming conventions.

Education of the software supplier, if this is a commercial system, may need to be factored into the migration.

REFERENCES

1. McDowall, R.D. (2000), Laboratory Data Systems, Chapter 13 in *Analytical Chemistry in a GMP Environment* (Eds. J.M. Miller and J.B. Crowther), Wiley Interscience, New York.
2. McDowall, R.D. (1999), Chromatography Data Systems III: Prospective Validation of a CDS, *LC-GC Europe*, 12, pp. 568–578.
3. Dyson, N. (1998), *Chromatographic Integration Methods*, Second Edition, RSC Chromatography Monographs Series (Ed. R.M. Smith), Royal Society of Chemistry, Cambridge, U.K.
4. Burgess, C., Jones, D.G., and McDowall, R.D. (1997), All You Wanted to Know about A/D Converters — But Were Afraid to Ask, *LC-GC International*, 10, pp. 791–795.
5. PDA (1995), Validation of Computer-Related Systems, *PDA Journal of Pharmaceutical Science and Technology*, Technical Report No. 18, 49(1), Supplement S1-S17, January/February.
6. FDA (1999), Inspection 483, Observations, Gaines Chemical Company, December.
7. FDA (1999), Warning Letter, Glenwood LLC, May.
8. FDA (1999), Warning Letter, Genesia Scicor, July.
9. FDA (2001), Inspection 483 Observations, Noramco Inc., May.

10. Kornbo, C. and McDowall, R.D. (2002), Scientific Computing and Instrumentation, January.
11. ISPE (2001), *Good Automated Manufacturing Practice Guidelines,* Version 4, International Society for Pharmaceutical Engineering, Tampa, FL, December.
12. IEEE Guide for Developing Software Requirements Specifications, Standard 1233-1998.
13. McDowall, R.D. (1998), *Scientific Data Management*, March 1998, pp. 7–12.
14. Commission of the European Communities, *Good Manufacturing Practice for Medicinal Products in the European Community, Annex 11 — Computerised Systems*, Brussels, Belgium.
15. McDowall, R.D. (1998), *Scientific Data Management*, pp. 8–14, June.
16. McDowall, R.D. (1998), *Scientific Data Management*, pp. 7–11, September.
17. FDA (2002), *Current Good Manufacturing Practice Regulations for Finished Pharmaceuticals* (21 CFR 211), revisions as of April 1.
18. FDA (2000), Warning Letter, Spolana, October.
19. IEEE Standard 829-1983, Software Test Documentation, Institute of Electronic and Electrical Engineers (1983).
20. Fewster, M. and Graham, D. (1999), *Software Test Automation: Effective Use of Test Execution Tools*, Addison-Wesley, London.
21. Wikensted, B., Johansson, P., and McDowall, R.D. (1999), Retrospective Validation of a Chromatography Data System, *LC-GC International*, 11, pp. 88–96.
22. Browne, D., Thompson, T., Mole, D., and McDowall, R.D. (2001), The Prospective Validation of an MS Data System Used for Quantitive GLP Studies, *LC-GC International*, 14, pp. 687–694.
23. Hambloch, H. (1994), Data Center Management and Good Practices, Chapter 7 in *Good Computer Validation Practices; Common Sense Implementation* (Eds. T. Stokes, R.C. Branning, K.G. Chapman, H. Hambloch, and A.J. Trill), Interpharm Press, Buffalo Grove, IL, pp. 113–140.
24. OECD (1995), *Consensus Document on Principles of Good Laboratory Practice Applied to Computerised Systems*, Organisation for Economic Co-operation and Development, Paris.
25. Lepore, P.D. (1992), Chemometrics and Intelligent Laboratory Systems, *Laboratory Information Management*, 17, pp. 283–286.
26. McDowall, R.D. (2000), Chromatography Data Systems V: Data Migration and System Retirement, *LC-CG Europe*, 13, pp. 35–38.
27. FDA (1997), 21 CFR 11, *Electronic Records, Electronic Signatures*, Final Rule, *Federal Register* 62, pp. 13430–13466.

21 Case Study 3: Laboratory Information Management Systems (LIMS)

Christopher Evans, GlaxoSmithKline
Ron Savage, GlaxoSmithKline

CONTENTS

Laboratory Information Management Systems (LIMS) are widely used in the laboratories of pharmaceuticals and related industries. LIMS is typically based on client server technology supported by a Relational Database Management System (RDBMS) as a storage repository (see Figure 21.1). They can be used to manage and process large amounts of electronic analysis data locally within a laboratory or company-wide between sites.

LIMS is now a mature technology, and many packages are commercially available. Therefore, this chapter will concentrate on the selection, configuration, and implementation of a LIMS in a pharmaceutical manufacturing environment, rather than the development of in-house software for laboratory management.

The introduction of LIMS into the laboratory is encouraged by Quality Assurance (QA)/Quality Control (QC) laboratory managers. It provides an automated, efficient, and regulatory-compliant means of dealing with the electronic data produced in the laboratory. The flow of data into the LIMS system is shown in Figure 21.2. The implementation of LIMS should also reduce the possibility of errors due to problems introduced by personnel performing repetitive scheduled tasks. There have been increasing demands from the regulatory authorities for data integrity, data security, and validation of LIMS in line with the current regulations and guidelines,[1–4] with LIMS becoming a focus of interest for these pharmaceutical regulators.

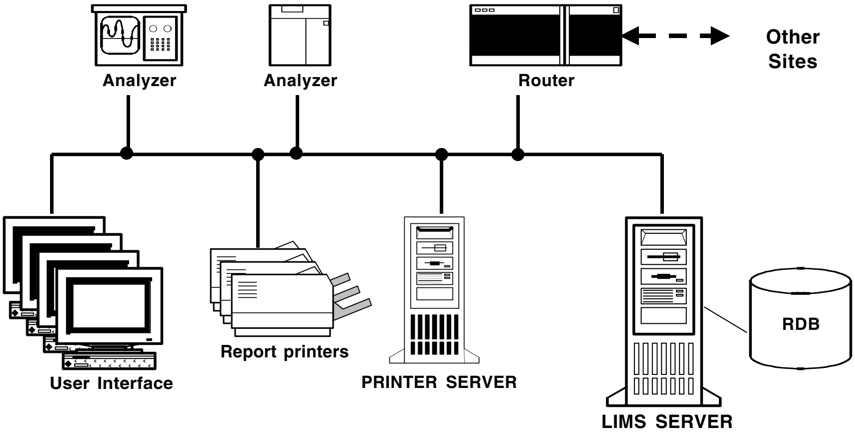


FIGURE 21.1 LIMS Physical Layout.

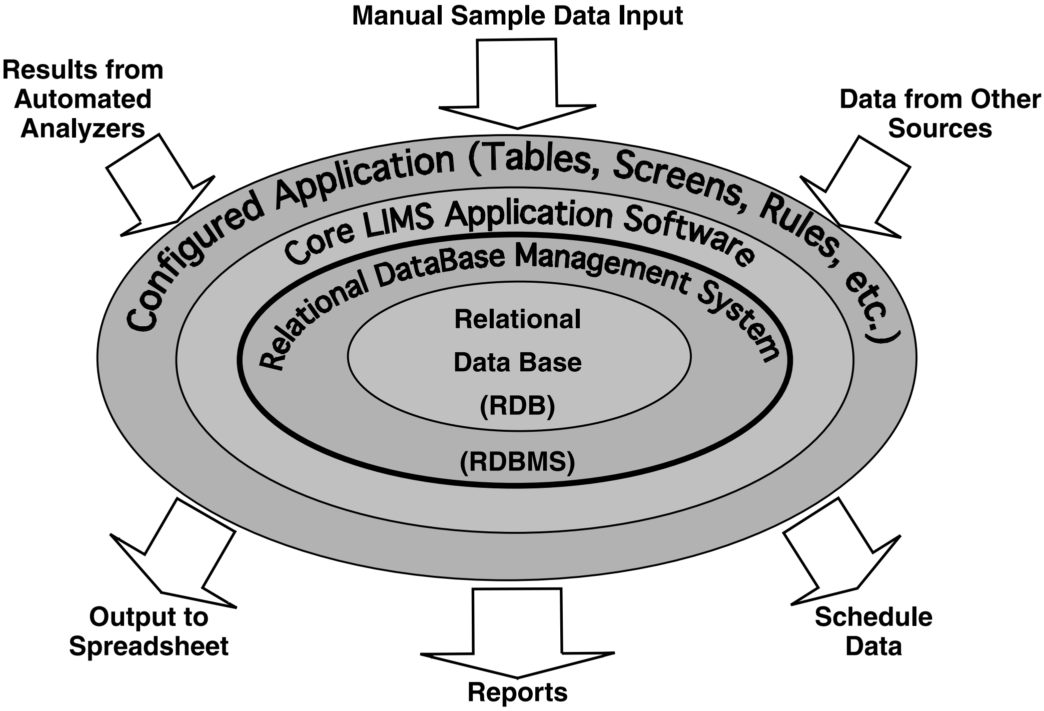

Manual Sample Data Input
Results from Automated Analyzers
Data from Other Sources
Configured Application (Tables, Screens, Rules, etc.)
Core LIMS Application Software
Relational DataBase Management System
Relational Data Base (RDB)
(RDBMS)
Output to Spreadsheet
Reports
Schedule Data

FIGURE 21.2 LIMS Database and Data Flow.

The modern-day LIMS package may be configured to meet the needs of most laboratory analysis and data handling activities. This configuration is made up of software modules designed to provide standard features (e.g., sample log-in, test management, work allocation, results entry, data processing, and reporting). As this is the case, there is now little need to develop bespoke/custom software to meet the normal laboratory needs. Areas of bespoke/custom software development are often limited to programming the reporting package to extract and print information from the database to meet specific customer needs, and developing interfaces between the LIMS and other business systems.

The success of fully integrating the LIMS within the laboratory relies in part on robust, reliable interfaces between the core element of LIMS and a wide range of analytical instrument and system interfaces (e.g., simple instruments such as those producing a single data point, complex instruments such as those producing spectra or result files, computer systems such as Chromatography Data Systems, and Manufacturing Resource Planning Systems MRPII). This will involve the interfacing of different computer platforms and instruments, and presents inherent difficulties in providing a compliant solution as required by the pharmaceutical industry.

As LIMS are basically configurable software packages, the quality of the design, coding, and documentation by the supplier is critical to obtaining an acceptably validated LIMS. The LIMS suppliers have developed an appreciation for the unique requirements of the pharmaceutical industry, and the last few years have seen an improvement in the system development life cycles used by some suppliers. Therefore, a thoroughly documented evaluation of the supplier's practices forms a fundamental part of the validation of LIMS.

LIMS are used to collect, store, and report data, which can be used to provide the final verification that a pharmaceutical product may be released to market or may provide data to be included in regulatory submissions. The validation of LIMS is of critical importance to the compliance profile of the company; failure to validate effectively could have a direct impact on the most important of the company's customers — the patient.

This chapter reviews the areas that are critical to the successful validation of LIMS and suggests an approach which represents good practice for validation of this type of application in the pharmaceutical environment.

Fundamental to the successful implementation and validation of LIMS is to understand the roles and interactions between the three main groups contributing to the project. These groups are:

- *Supplier:* The group that sells the LIMS application. As the developer of LIMS, it is expected that the supplier has in-depth technical knowledge of the product and has experience in its implementation system.
- *Integrator:* The group engaged by the customer to develop interfaces and customer specific code. The integrator may also provide the implementation expertise. Use of an integrator is optional, dependent on the services provided by the supplier and the level of expertise of the customer.
- *Customer:* The group requiring the implementation of the validated LIMS. Customers will have varying levels of expertise, experience, and resource availability to contribute to the implementation and validation.

LIMS FUNCTIONALITY AND TECHNOLOGY

FUNCTIONALITY

A typical LIMS would have the following functionality related to the management of the sample life cycle:

- Log-in of samples and receipt tracking
- Assignment of tests to samples
- Production of analyst worksheets and schedules
- Real-time data input
- Interfacing with analytical equipment and Chromatography Data Systems (CDS)
- Performance of customer definable calculations
- Monitoring of Out-Of-Specification (OOS) results using customer programmed limits
- Review and reporting of analytical results
- Comparison of analytical results with specification
- Maintenance of sample status
- Basic statistical analysis of analytical information
- Audit trail of events linked to results

In addition a typical LIMS would have the following functionality-related management of the laboratory:

- Test method and specification management
- Management of analytical equipment calibration schedules
- Management of the stability program
- Export of data for statistical analysis
- Configurable reporting of laboratory performance metrics

A useful description of LIMS functionality can be found in ASTM Standard E1578.[5]

Most modern LIMS are configurable applications built around sets of standard modules. Early in the validation effort it is important to determine that all of the application modules have been developed to the supplier's quality standards, have not been revised to fit a LIMS application for a specific customer, and do have multiple implementations for which references are available. If the modules are not part of the supplier's standard product, then confirm that they were developed using the supplier's quality standards. If no code review or design review have been performed, the customer will need to make a judgment on the risk this imposes (taking into account the results of the GxP Assessment) and, if necessary, request that the supplier perform and document a code review as part of the project.

In order to ensure that the software comprising the validated LIMS is controlled, it is good practice to avoid installing other software (e.g., word processors, drawing packages, e-mail) on the LIMS client PCs. However, there may be the need for the use of some standard software packages (e.g., spreadsheet packages). The client PC configuration, including the LIMS client, and the additional packages, should be used during the validation testing to ensure that there are no conflicts between the applications. These packages may be updated from time to time (usually by the IT department), in which case this must be taken into account when assessing the needs for ongoing validation maintenance.

Technology

A complicating factor in the validation of LIMS is the wide variety in the scale of LIMS implementations. A LIMS can vary from a PC-based application in a single laboratory to a client/server-based system running across multiple sites on a company-wide area network with shared access to servers and multiple interfaces to other business systems.

Development of browser-based interfaces to replace the more traditional client software is a recent innovation and should provide the basis for a simplification of the validation activities relating to the configuration of the client PCs.

DEFINING BUSINESS REQUIREMENTS FOR LIMS

Definition and documentation of the business requirements for LIMS is the first step in the implementation process. These requirements are not system specific and define, at a high level, what LIMS will be expected to do for the business. The business requirements are used in defining the system requirements in the supplier selection process and in the financial justification for the system.

The business case will normally be based on the ability of the LIMS to:

- Increase the efficiency of the laboratory while reducing costs and possibly resources
- Provide accurate and reliable data to support product release, development activities, or submissions to the regulatory authorities, and
- Decrease the compliance risk through automation of manual business processes

The implementation of LIMS should not be performed without fully assessing its impact in the wider context of how it will be integrated into the laboratory business processes and how it will impact on existing laboratory staff's ways of working.

It must be clearly understood, and covered in the business case, that the implementation and initial validation of LIMS is only one component of the cost of the system through its life. Installation is just the beginning of an ongoing commitment to the maintenance of LIMS in a validated state. This includes the commitment to the ongoing cost of maintenance of the system, infrastructure, and business processes.

The Regulatory Influence

Today regulatory authorities accept the principle of using a computer system for laboratory management and for controlling release of product to market, and it has now become the norm for laboratories to use LIMS. Regulatory inspections of laboratories often audit the use of LIMS. Typical areas of regulatory interest include:

TABLE 21.1
Regulations Applicable to LIMS

U.S. Code of Federal Regulations Title 21, Part 211, Clause 68, and Title 21, Part 11	U.S. Code of Federal Regulations Title 21, Part 58 GLP for Nonclinical Laboratory Studies	European Union Directive 91/356/EEC Annex 11
Personnel	Personnel	Personnel
Building/Facility	Facility Management	Equipment Siting
Equipment Design	Equipment Design	Validation Life Cycle
Equipment Maintenance	Equipment Maintenance	Contracted Out Services
Standard Operating Procedures	Standard Operating Procedures	Business Continuity (Contingency) Plan
		Detailed Description
Record Retention	Record Retention	Record Retention
Calibration		Software Quality
Change Control		Change Control
Testing	Testing	Testing
Backup Files		Error Recording
Data Storage		Data Storage
Hard Copy Records		
Ongoing Evaluation		
Security		Security
Electronic Records		Electronic Records
Electronic Signatures		(See EU Directive 1999/93/EC)

- Documentation and extent of qualification and testing
- Change control, including software, configuration, and documentation
- Validation approach, quality assurance, and auditing
- Compliance with Electronic Records and Electronic Signatures regulations
- Documentation of specification and design
- Traceability between specifications and tests
- Failure to validate reports
- Ineffective Standard Operating Procedures, including security practices and backup and restore
- Use of unvalidated spreadsheets for data collection or extraction from LIMS

Application of the current regulations and how the authorities interpret them is not always consistent. It is therefore important to develop and document a well-planned strategy to ensure that all aspects of the various regulations are addressed and the resulting installation is easy to maintain. There is general guidance within the regulations covering computer systems, but mainly it is not specifically aimed at LIMS. Nevertheless, it must be clearly understood that however general the regulations and guidelines, each regulator will have his/her own specific interpretation of them. Table 21.1 indicates the coverage of the applicable parts of the regulations.

THE GOOD PRACTICE APPROACH TO LIMS VALIDATION

Good Practice for the use of computer systems in laboratory applications is now well established and is based on many years of experience. LIMS is simply another use of the power and flexibility of client/server or browser-accessed computerized systems. Therefore, it may be validated in a similar manner to other systems utilized within the pharmaceutical industry. Specifically, LIMS is

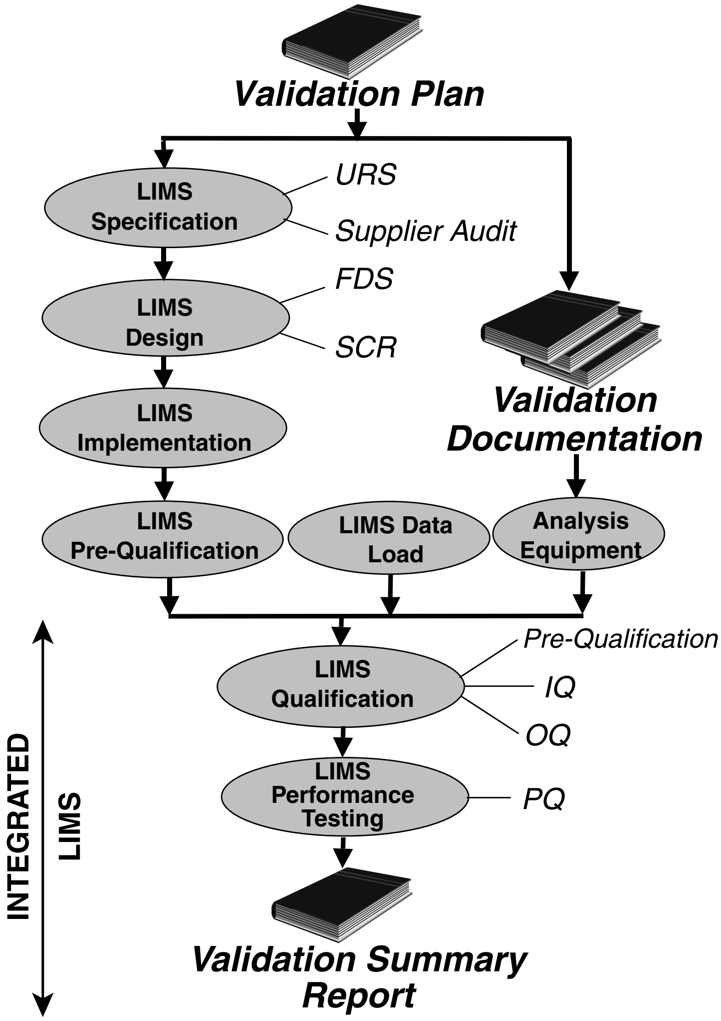


FIGURE 21.3 LIMS Validation Documentation.

deemed to be a form of automated system and it is therefore recognized that the development and implementation of LIMS should be conducted in line with a defined life cycle (see Figure 21.3) such as the Quality methodologies identified as Good Automated Manufacturing Practice (GAMP).[1]

Prior to the start of the validation project, time and resources (and therefore money) must be expended to ensure that the requirements for the validation of LIMS are assessed from a regulatory and business perspective. This assessment should be executed as early as possible, preferably prior to purchasing LIMS. Many projects have had problems due to lack of appropriate funding, and consequently validation difficulties, due to the project personnel not understanding sufficiently early what needs to be done in the project life cycle.

It is the responsibility of the pharmaceutical manufacturers to ensure that they are using "quality" suppliers for the LIMS package and suitably qualified system integrators for implementing the package. This is easy to state but difficult to do as there are many suppliers and system integrators in the market who will advise that their solutions and practices are the best and have passed many regulatory inspections. The pharmaceutical manufacturer must therefore take appropriate steps such as performing supplier evaluations (which will normally require an audit of the LIMS package supplier and the integrator) before any decision regarding which LIMS to implement is made.

Where the pharmaceutical manufacturer has a requirement not met by the supplier's application, there may be a requirement for development of customized software. If the LIMS supplier is to be used as the developer, the development life cycle and ongoing support arrangements should be agreed. The LIMS supplier may wish to include the customization in a future release of the LIMS to increase the functionality of the package. Care should be taken to ensure sufficient testing of the customized code before implementation as it may not have been used previously in a production environment.

VALIDATION LIFE-CYCLE APPROACH

The validation life cycle is typically split into eight main qualification phases as shown in Table 21.2 and are all executed under the control of a Validation Plan. It must be clearly demonstrated that the Quality Assurance (QA) function of the pharmaceutical manufacturer is endorsing the validation approach for the implemented LIMS as documented in the Validation Plan. In its simplest form this will require the QA representative to authorize the Validation Plan and the subsequent key documents produced to support validation (e.g., Qualification documentation, Reports, etc.). Table 21.2 shows the linkage between project activities and the Qualification process, which will be under the control of the Validation Plan.

Requirements Definition

The main purpose of this phase is to ensure that requirements are defined and documented to allow:

- The justification to senior management that LIMS is a cost-effective solution. These "Business Requirements" will be the basis for applying for funding for purchase and implementation of the application. The owner of LIMS must understand the benefits and drawbacks of implementing LIMS prior to purchase; the requirements definition is the *only* time in the project that it can be achieved at a reasonable cost.
- The documentation of the requirements of the users, maintainers, and supporters of the system. These "User Requirements" will form the basis of the configuration and testing of the application.
- The documentation of the infrastructure requirements. These, together with the User Requirements and Business Requirements, will be used in selecting the supplier and application.

Supplier and Application Selection

The LIMS application should be chosen from an analysis of the application capabilities against the requirements and conformance to the GxP regulations for the market to which the pharmaceutical products will be sent. This initial choice of application will lead to a preferred supplier. For the supplier's system to be appropriate for use in a regulated laboratory, the supplier must have used good software development practices during the development and testing of the system. Failure to use good software development practices can lead to a requirement for the purchaser to perform significantly more testing to achieve a validated LIMS implementation. Such testing will require a knowledge of the underlying structure of the LIMS database, and this information is often not released by the application supplier.

Assessment of the supplier's software development practices can usually be achieved by auditing. Experience shows that any actions from audit observations should be agreed with the supplier and included in the purchase and support contracts for the system.

TABLE 21.2
Validation Phases and Project Activities

Validation Phases	Project Activities
Requirements Definition	• Business requirements and business case development • User Requirements Development (Initial) • Project Planning (and Supplier Quality Planning) • GxP assessment
Validation Planning	• Validation Plan production • User Requirement Specification development (Version 1) • Supplier assessment and system/integrator selection • Electronic records and electronic signatures assessment
Install Development Environment	• Install and document development hardware and software
Design and Configure LIMS Package	• Functional Design Specification production • Detailed Design Specification production for hardware and LIMS Package • Configure LIMS package • Configuration review • Design Review
Design and Code Bespoke/Custom Software	• Functional Design Specification production • Program specifications • Develop bespoke/custom software • Source Code Review • Unit testing • Design Review
Pre-Qualification Activities	• Installation of hardware for the production system (computer system hardware, servers, and client PCs) • Installation of software (operating system, standard software modules and bespoke/custom software) • Implementation of infrastructure (e.g., networks and communication links) • Develop test data sets
Installation Qualification	• Verify correct installation of hardware and record as initial configuration • Verify installation of software and record as initial configuration • Verify correct installation of network infrastructure and record as initial configuration • Verify import of test data • Verify availability of supporting supplier documentation • Verify equipment environment is suitable in terms of temperature, humidity, and electrical/magnetic interference
Operational Qualification	• Functional Testing • Security • Backup and restore • Business Continuity (Contingency) Plan
Promote to Production	• Load and verify static data load • Implement support system (e.g., Help Desk, Incident Management) • Issue Interim Validation Report releasing LIMS for use in the production environment
Performance Qualification	• Ongoing verification of functionality • Performance monitoring via support organization
Validation Reporting — Authorization for Use	• Validation Report
Validation Maintenance — Operational Compliance	• Ongoing support and integrity checks • Performance monitoring via support organization

Validation Planning

The Validation Plan documents the validation scope of the LIMS. The Validation Plan must clearly define the boundaries of the validation activities, including the responsibilities of each contributor, and will therefore identify what is within the boundary of LIMS (e.g., interfaces, reporting tools, etc.). It is vitally important to ensure at this stage that there is full validation coverage for LIMS and all associated systems. The Validation Plan will define a number of specific areas on which the approach to validation will be based:

Responsibilities of Personnel/Organizations: The success of any project of this type will rely on personnel having the right experience in terms of LIMS and GxP. These personnel will have the responsibility for authorization of validation documentation and executing the validation activities within the scope of this plan. The responsibilities of the supplier, integrator, and customer must also be understood and documented.

Scope of Validation: The boundaries of the validation project must be defined to ensure that there is full coverage. For example, will the analytical equipment or Chromatography Data System interfaces be validated as part of the project, will Supplier Evaluations be required, etc. It is very important at this stage to determine what is within the scope of the LIMS Validation Plan and what will be validated under other associated Validation Plans. The validation of the implementation of processes and information management within the laboratory should be managed as a cohesive whole to ensure that all parts of the LIMS are developed and validated to the appropriate standards. This may be achieved by the use of a Validation Master Plan (VMP) for all the laboratory processes and information management. The Validation Plan for the LIMS and any associated plans for other interfaced systems would be referenced in and be under the control of this VMP.

Standards/Procedures: In order to ensure consistency, standard and formal processes should be used in the project. These processes may be documented in guidelines, for example GAMP[1] or, more correctly, in company standards/procedures. Company standards and procedures will typically cover production of validation documentation and project governance (e.g., documentation management, change control, configuration management, incident management, issue/risk management, and business continuity planning).

Validation Approach: The project methodology to be followed for validation of LIMS must be clearly defined in the Validation Plan to ensure that the most efficient approach is taken. This document should provide the project team and the LIMS Supplier/Integrator with sufficient detail to allow the appropriate documentary evidence to be produced to support the finally handed over compliant system. It is important that the Validation Plan (or at least the relevant information) be issued to the LIMS Supplier/Integrator to ensure that no surprises regarding the validation expectations occur at a later stage in the project. It is acceptable for the supplier, integrator, and customer to use different validation methodologies, provided that the customer:

- Has understood the interface points and gaps between the methodologies
- Is satisfied that the validation methodologies of the supplier and integrator meet the customer's quality standards

Where a LIMS implementation project is split over a number of different locations (e.g., different laboratories or different sites) but is designed to provide an integrated solution, it may be appropriate to produce individual Validation Plans and associated documentation for each of the LIMS location. Where an installation is implemented and supported on a company corporate basis, it may be more appropriate for the VMP and the resulting validation documentation for the core LIMS to be managed from a central location. In such cases Validation Plans and associated validation documentation supporting the local implementation should be maintained on site.

The laboratory processes and information management VMP mentioned above should act as an umbrella document for the whole project, identifying the scope of each individual Validation

Plan. Using this method of breaking down the documentation will allow a much more focused approach to be taken to the production of documentation; it also simplifies the problems normally found in obtaining validation document authorization across sites and geographical regions. A further advantage of using individual Validation Plans is that the individual LIMS instance may be put into operation as soon as testing and installation are complete, without having to wait for the final validation report covering the whole project. Also, when periodic reviews are performed on LIMS, this may be performed on the local instance of LIMS without needing to review all other instances.

As the Validation Plan will be produced at such an early stage, it will inevitably only encompass the initial understanding of availability of personnel and project scope; therefore it will need to be further developed during the project. The project team and LIMS Supplier/Integrator must appreciate that this is a live document that will be developed during the project and that it will continue to be live until LIMS is decommissioned.

Validation Determination

It is part of the early decision-making process to assess which parts of the LIMS are to be subject to full validation and which are to be covered by good IT or engineering testing practice. The function of LIMS within the company computing architecture may mean that it is utilized for both product quality related and non-product-quality-related activities. The validation determination for LIMS ensures that the validation effort is focused toward those system elements with direct GxP* impact. The justification for validating only certain parts of LIMS must be based on an assessment of the GxP criticality of the data in the LIMS database, the use of the data and the sources of the data. In the ideal world LIMS would be dedicated to GxP activities. However, the economics of the system may mean that it is not practical to separate GxP and non-GxP activities into separate systems.

Due to the critical nature of the data stored in the LIMS database, data integrity must be assured. Current technology allows for disk mirroring to ensure that any database additions and changes (and therefore changes to electronic records) are copied to additional, and possibly remote, locations. The function and use of each mirror for backup, for disaster recovery, and for system startup after shutdown must be understood. Each of these scenarios must be tested as part of the validation.

Typical GxP-critical data elements include:

- Batch/Lot Number
- Item Number
- Shelf Life
- Tester Identification
- Material Specification
- Sample Number
- Sample Date
- Sample Time
- Sample Status
- Test Methodology
- Retest Days
- Test Results

User Requirements Specification

The purpose of the User Requirements Specification (URS) is to collate the customer requirements for how the LIMS application is to operate in both the laboratory and company environment. There may of course be areas in the URS that are wishes rather than essentials and also some requirements that are GxP critical. Following review, the first version of the URS may be used for obtaining quotations and proposals from the proposed LIMS Supplier/Integrators, and as a result this initial version will act as the foundation stone for the project. Time spent in ensuring that the URS is complete, structured, and clearly understandable will deliver large savings of time and effort (and therefore costs) at later stages of the project, and will also reduce the risk of project failure.

* GxP – The combination of Good Manufacturing Practice [GMP], Good Laboratory Practice [GLP], and Good Clinical Practice [GCP].

Unfortunately, it is human nature to wish to spend as short a time as possible on this phase of the project as it does not seem to produce anything but documentation, when the customers want to see something more tangible.

The feasibility of any specific LIMS product meeting the requirements of a laboratory (both technical and maintainability) must be fully assessed prior to purchase. This is part of the LIMS Supplier Assessment process. The assessment process must take into account the supplier's proposed technical solution, ongoing maintenance support (including future development), as well as any company strategy for laboratory management. The decision on which LIMS supplier to use is of major importance and should be based on a detailed understanding of both the supplier and the proposed LIMS product. It is therefore essential at this early stage in the project to build a relationship with the LIMS Supplier/Integrator to use the knowledge of his/her technical personnel in the development of the URS.

The approach of bringing the supplier on board at this stage will assist in the structuring of the URS; the supplier will know what he or she needs to see in the URS to provide a meaningful proposal for a technical solution and how the design documentation is structured. It is not common for such close alliance to be established in these types of projects, but the potential for successful and efficient project delivery makes putting out some effort in this direction worthwhile.

The structuring of the URS to identify uniquely the numerous customer requirements will allow the pharmaceutical manufacturer to demonstrate traceability of each of the requirements (GxP and non-GxP) through the design and testing process, which is now a regulatory expectation.

The agreed URS should clearly identify the following:

- System size and capacity (e.g., numbers of items, Client PCs, User Interfaces, peripherals, data storage capacity, etc.)
- Integration of LIMS with other systems (e.g., MRP II, Chromatography, use of LAN/WAN)
- The number and types of pieces of analytical equipment to be interfaced to LIMS. There will also be a requirement to define the means of communication with these pieces of equipment; this communication link may be typically via RS232 or network interface
- System performance and availability targets
- Documentation requirements (e.g., user manuals, technical manuals, software listings, database structure diagrams, etc.)
- Requirements for data presentation, records, and reports
- Requirements for data manipulation (e.g., calculations)
- Requirements for the use of spreadsheets for the collation or manipulation of data
- Details of external reporting and statistical analysis packages
- Details of supplier change control and patch documentation

The URS is the responsibility of the pharmaceutical manufacturer as the customer. However, it is highly unlikely that they will have sufficient LIMS experts who can be dedicated to the project. It is strongly recommended that future LIMS users provide their input to ensure that there is buy-in for the design and functionality of the solution. This applies in particular to the Client PC user interfaces (e.g., display graphics, user entry screens) and reporting mechanisms.

Where expertise on design is required in the form of LIMS Supplier/Integrators or consultants, these personnel should be included within the project team, but they must work closely with the users. Using the right people at this time will enable the requirements of the future LIMS users to be converted into a functional design document that will form the basis for a successful configuration of the application. When the URS has been agreed upon, which may take several iterations, it will act as the source document for the Design Review process. Any change to the URS will have its impact felt throughout the rest of the project, and the later any changes are introduced into the

URS (e.g., after the design/configuration/testing has begun) the more significant the effect will be on the project cost and time scales.

Supplier/Integrator Evaluation

Due to the expanding nature of the laboratory information management market there are a growing number of suppliers with offerings in this area. Mainstream LIMS suppliers are backed by supplementary application suppliers offering connectivity software, workflow solutions, archiving solutions, reporting packages, implementation expertise, etc. The LIMS market is also sufficiently large and diverse to support segmentation into specialist applications (e.g., manufacturing vs. research, highly configurable vs. limited configuration packages, multilab vs. single lab solutions).

It is essential to conduct a Supplier Evaluation of the proposed LIMS Supplier and Integrator (which normally includes a Supplier and System Integrator audit) as part of the supplier selection process. This evaluation must be conducted prior to placing an order for LIMS as it may be the case that there are some major concerns regarding the ability of the supplier/integrator to deliver LIMS that meet the customer's quality standards.

Supplier/Integrator Audit

The purpose of a Supplier/Integrator audit is to allow the pharmaceutical manufacturer to review documented evidence of the application of the supplier Quality Management System (QMS) throughout the development of the LIMS package. The Supplier Audit will also confirm that the supplier is capable of delivering the correct standard of software engineering and documentation for LIMS.

It has been shown many times that ISO 9001: 2000 accreditation does not necessarily mean the supplier is capable of developing software and hardware products to meet customer quality requirements. This can be because the LIMS Supplier/Integrator is accredited for the implementation and integration of LIMS rather than the development of software. In these cases the only way of gaining confidence in the approach taken in the software development, testing, and change control processes is to visit the LIMS Supplier/Integrator's premises. Where a supplier is accredited to ISO 9001: 2000 TickIT, it will give the pharmaceutical manufacturer greater confidence that a defined life cycle will be followed as it is specifically aligned with the production and ongoing development of software.

When an audit is conducted, the Supplier Audit Report will be the documentary evidence approved by the pharmaceutical manufacturer, which will identify any issues raised during the audit and provide recommendations for any corrective actions. It is expected that these corrective actions will be implemented as part of the approval of the LIMS Supplier/Integrator for the project and will form part of the commercial contract between the two organizations.

The audit is a critical step in the validation of LIMS as it allows the pharmaceutical manufacturer to evaluate key testing and validation deliverables from the supplier. The pharmaceutical manufacturer is accountable to the regulatory authorities for any deficiencies in the chosen Supplier/Integrator's capability. The pharmaceutical manufacturer is expected to bridge any gap in standards by assigning his or her own staff and employing consultants to assist the supplier in the validation of LIMS.

Design Specifications

Although the pharmaceutical manufacturer has determined the requirements and recorded them in the URS, they are normally generic and not technically specific to any particular LIMS application or supplier. The purpose of the design specifications is to translate requirements into a specific technical solution which will fulfil the requirements of the customer. The structure of the design documentation detailing the physical, functional, and performance criteria for LIMS should be directly traceable back to the URS to allow the supplier to confirm that all requirements have been addressed. This may be achieved by the use of a cross-reference (traceability) matrix which identifies each of the user requirements, references the location in the design documentation where the requirement has been addressed, and also holds the justification for any requirements excluded or not to be fully implemented. The design documentation will usually consist of a Functional Design Specification (FDS) to provide a high level overview of the proposed LIMS with more detailed specifications for the hardware, software, and configuration. The design documentation should typically specify the following:

- System Overview Description with diagrams (a concise nontechnical description of the functionality and operation of the LIMS covering all interconnected systems), which for complex systems should be a separate document providing a description of the systems that may be used to describe the system to a regulator
- System Architecture with details of hardware with diagrams
- Data Architecture with details of the sources, uses, and ownership of each static data item
- System landscape showing the interactions of the various instances of the application
- Interfaces with diagrams (including networks, systems, and analytical equipment)
- List of software modules with details of versions, standard software products (e.g., spreadsheet packages), etc.
- System performance requirements (e.g., timing, memory storage, availability, and spare capacity)
- Number of users, user hierarchy, and response time required from LIMS
- Software design documentation
- Communications and network protocols (e.g., RS232, TCP/IP, Ethernet)
- Methodology for data storage, backup, and retrieval
- Server environmental conditions including power supplies
- Client PC user interface software configuration details and terminal emulation requirements
- Peripheral devices (e.g., printers and backup devices)
- Functional descriptions of LIMS operations
- Business Continuity (Contingency) Plans, maintenance, and operating procedures
- Application software and reports
- Security and access control methods

Each of the elements within the FDS should have a unique identifier for traceability through the testing phase of the project. The functions identified should be in sufficient detail to allow meaningful tests to be produced. The results of these will be recorded in the qualification test documentation.

Where bespoke/custom software is produced, Software Module Design Specifications are required.

Design Review

Design Review is a formally documented and structured process of confirming that the LIMS design is both fit for purpose in terms of the requirements of the users/business and also meets regulatory expectations. This is all part of the process of building Quality Assurance into the design.

The process begins with a review of the agreed Business and User Requirements against the proposed design. This is to identify any shortfall in the proposed design which may be the result of a lack of understanding of the users' needs by the LIMS Supplier, or perhaps the definition of what is required is missing from the documented requirements. The outcome of this review should be formally documented and will act as a key milestone in agreeing that the proposed design meets the expectation of the business. Following such agreement the detailed design activities for LIMS may commence.

The information used in this initial Design Review is the URS, the LIMS Supplier/Integrator's proposed design specification, the purchaser's standards, the pharmaceutical regulations, and any local regulations. The intention of the ongoing Design Review process is to monitor the development of the design throughout the project and as such does not end until LIMS is in operational use. Each change to the design following the initial Design Review should be given full consideration of its potential impact on the quality of the final LIMS installation. Elements that make up the reviews of the detailed design will cover the execution of Configuration Reviews, Source Code Reviews (SCRs) for customer-created applications (e.g., user customized reports and spreadsheets) which utilize data from the LIMS, GxP Assessments, etc. It is important to note that the Validation Plan must clearly document the methodology to be followed and the documentation to be produced supporting the Design Review activities.

During the Design Review process a key requirement is to evaluate the system design against the requirements for ongoing use and maintenance. For example, if LIMS utilizes analytical equipment interfaces which must be isolated for maintenance, LIMS must be designed to allow for maintenance activities without there being any effect on the operation of the rest of the system and any impact on the use of LIMS.

The Design Review process should begin as soon as the first version of the URS has been produced or after the LIMS Supplier has been selected, whichever is earlier, and then effectively continues until the implementation of LIMS in terms of hardware, software, database configuration, and beyond. The Design Review process should then become a standard part of any ongoing modification to the LIMS through change control. The objective of the Design Review process is to ensure that a quality system is installed and maintained, thus giving a system in which the pharmaceutical manufacturer has a high level of confidence. Quality can only be built into the design, configuration, and implementation of LIMS; it cannot be tested in afterward. Therefore, if the project does not take the opportunity to apply a structured and formal Design Review process, it may result in an expensive and time-consuming exercise to redesign LIMS; this could easily happen if the installed LIMS does not meet Business and Customer needs.

Rigorous testing has the potential for finding fundamental faults with the design of LIMS, which may mean there is a requirement for redesign at a late stage in the project. The resulting redesign may negate Design Reviews that have already been performed as well as having a knock-on effect on any testing that has already been conducted.

The whole point of applying a life-cycle approach (see Figure 21.4) as defined in GAMP[1] is to provide a structured, controlled, and confidence building approach to the development and validation of LIMS and associated analytical equipment interfaces. The purpose of all activities prior to the implementation of LIMS is to assess and verify rigorously that the design will meet with the requirements of the customers and also the expectations of the regulatory authorities. As a result of this process, in-depth testing can be focused where it is needed, thus reducing the overall amount of testing required.

There is a direct relationship between the quality of the documentation produced during this review process and the quality of the finally installed system. This is because clear, concise, and accurate documentation describing the required implementation will allow the project team to generate test scenarios most closely matching the system use and boundary conditions.

Application Development

In order to provide a controlled, structured environment for the development, testing, validation, and ongoing use of LIMS, it is important to set up server-based environments at an early stage in the LIMS project. These environments are typically the following:

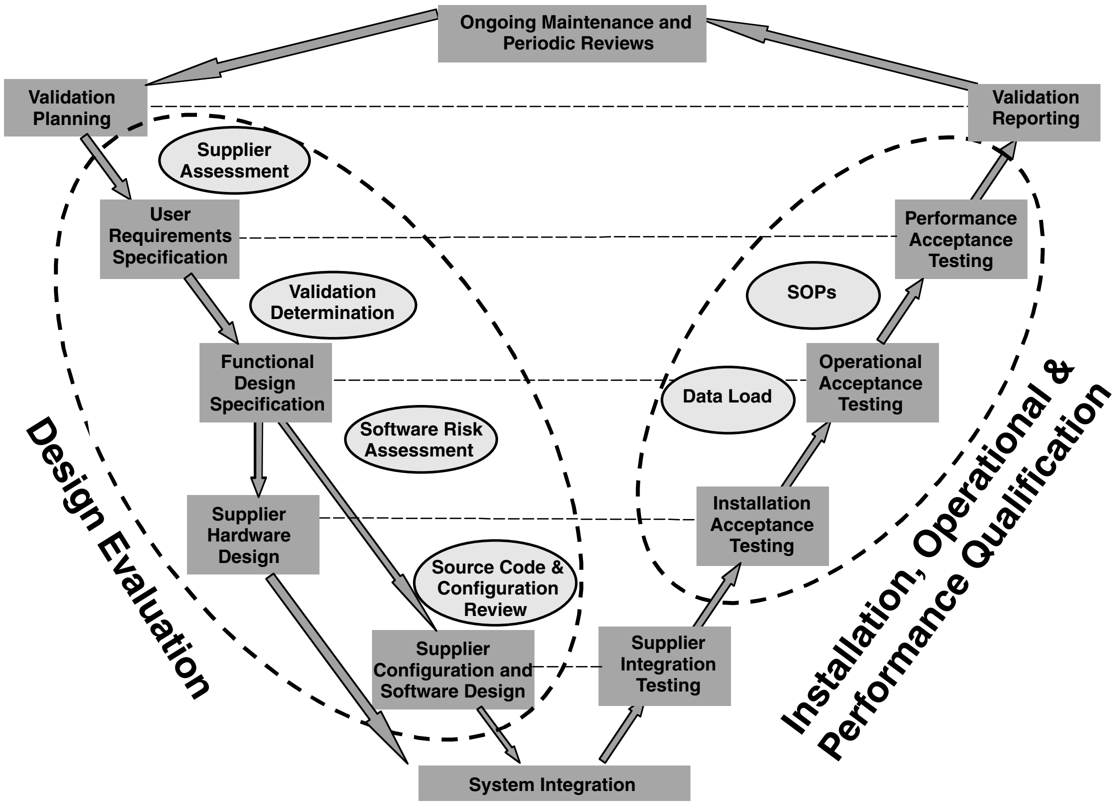

Ongoing Maintenance and Periodic Reviews
Validation Planning
Validation Reporting
Supplier Assessment
User Requirements Specification
Performance Acceptance Testing
Validation Determination
SOPs
Functional Design Specification
Data Load
Operational Acceptance Testing
Software Risk Assessment
Supplier Hardware Design
Installation Acceptance Testing
Source Code & Configuration Review
Supplier Configuration and Software Design
Supplier Integration Testing
System Integration
Design Evaluation
Installation, Operational & Performance Qualification

FIGURE 21.4 LIMS Validation Life Cycle.

Development Environment

This is the area where the project team is able to configure existing standard LIMS software modules or, where required, create new software modules to meet the specific requirements of the design specifications. This area may also be used for piloting of proposed LIMS software module constructions to determine the most suitable application of the LIMS for the customer. This environment is sometimes called a "sand pit" as it is expected that the developer may need to try out different means of implementing the agreed design, not all of which will be successful. These trials are often not formally documented and are used as a means of evaluating the LIMS product. With modern LIMS applications the functionality required by the customer is usually available in the standard modules and therefore most of the developer's activities will revolve around implementation of a configured solution that meets the customers' needs. This environment will have restricted access to enable the development personnel to create and store versions of custom software modules and instances of LIMS. There will need to be extra control on access to the version-controlled standard LIMS software modules in such a way that they cannot be inadvertently modified; this would normally be achieved by maintaining a secure library maintained under a Configuration Management System. Control of the library will normally be by personnel independent of the development team who will be responsible for managing and maintaining the module library on behalf of the LIMS owner. It is also normal practice for the developer to carry out informal testing of modules and proposed configurations of LIMS in this environment to ensure that no obvious errors are present. When the module or instance of LIMS has been accepted by the owner as being ready for formal testing, it will be transported to the Validation/Test Environment.

Validation/Test Environment

This is where the final version the software is tested, through structural testing and functional/stress testing prior to releasing the LIMS instance for use in the production environment. This environment will be strictly controlled and will only be used for validation and qualification activities. The environment, hardware, software, data and configuration should be an accurate representation of the production environment. Testing should be in accordance with good IT practices and formally documented.

At the time of validation testing, training of LIMS users will be required and operational documentation available (e.g., SOPs). All data sets used for the validation testing of LIMS will be retained under configuration management as reference information supporting validation. Following the successful completion of the validation exercise a decision will be made by the project Quality Assurance representative regarding the LIMS software to be released to the production environment.

Production Environment

This is where the validated version of LIMS is made available for the users. This environment must only be accessible to trained users of LIMS. Release of software from the development/test environments to the production environment must be strictly controlled. The Production Environment will be used only for "live" LIMS data.

Configuration Management

The flow of software will in all cases be from the Development to the Test environment, and then on to the Validation, and finally the Production environments. Modifications and development of new software modules occur only in the Development Environment. This discipline must be maintained to ensure that modules already in the Test or Validation environments are not modified. This could cause a mismatch between the reference copies of the modules, held in the configuration management system, and the tested modules. The Configuration Management process and tool are key in controlling the makeup of the different environments and the movement of components between them.

The above environments may be set up either on separate servers or the same server. However, it is normal practice for the development and testing activities to be performed on a server specifically dedicated to original development, testing, and ongoing support. Where separate servers are used, confirmation of equivalence between the validation and production environments [software and hardware] will be part of the validation exercise.

It should be possible to determine what versions of software modules, hardware components, and documentation are used to make up LIMS throughout the design, development, testing, and use of the system. As such, Configuration Management should cover

- Hardware components (e.g., servers, PCs, interfaces to laboratory equipment, etc.)
- Software (e.g., operating system, database, application software modules, data sets, bespoke/custom/standard software modules, etc.)
- Documentation (e.g., configuration records, test records, validation protocols, validation reports, etc.)

The overall role of the Configuration Management system is to allow the management to control all components of LIMS, so that the System Owner can determine what components make up LIMS currently and at any time since it was developed.

In the context of the initial project to implement LIMS, Configuration Management is a means of ensuring that when the system is validated there is a specific record of the makeup of the instance of LIMS. If this information is not available, changes to modules as a result of test failures cannot easily be assessed to determine any impact on testing that has already been performed. For ongoing maintenance there is a similar need to understand the makeup of the LIMS components. This is a process of management/control that is expected to be in place as part of the ongoing support activities for a GxP computerized system.

Source Code Review

Although it is no longer common practice to implement bespoke/custom software solutions when implementing LIMS, it may be necessary for software modules to be developed or modified to meet a specific business need. In such cases the new or modified module should be developed in accordance with a software development life cycle.

As part of the software development life cycle of the custom/bespoke or modified software, it should be subject to SCR. This is another means of building quality into LIMS. When conducting an SCR, the source code should be reviewed against the agreed design and Good Programming Practice (GPP). Particular emphasis should be applied to the software deemed as GxP critical.

The SCR process will consist of assessing the agreed design against printouts of any bespoke/custom software source code in order to assesses compliance with GPP. The aims of the SCR for the software are to:

- Identify logic errors within the code with particular emphasis on critical functionality
- Identify any redundant code (and where appropriate recommend removal)
- Identify any dead code (and where appropriate recommend removal)
- Verify that the software has been written in accordance with the agreed programming standard
- Verify that the code contains sufficiently meaningful comments to ensure that it can be maintained by a competent software engineer
- Verify that critical algorithms and calculations are correct

- Verify the use of version control and change control within the modules, thus verifying that there is traceability of changes
- Confirm that software listings utilized in this process are complete and accurate

The above list is applicable for the core LIMS software modules, interface software, and any other bespoke/custom software utilized by the LIMS. The reviewed source code listings should be marked up with comments and identified issues and then signed as evidence of the review.

Following the completion of the SCRs a review report or a number of review reports will be produced to summarize the review findings. These reports may identify actions to be taken by the LIMS Supplier/Integrator or the personnel responsible for performing the validation of LIMS. These actions may range from identifying specific testing to revisions of the source code due to deviations from GPP, or to correcting the code errors.

Test Strategy

The testing of LIMS is the final intensive activity that is performed prior to release of LIMS into the production environment. A test strategy should be developed to consider what testing is required to mitigate the potential risk of LIMS failure impacting the patient and/or the business and to build a high degree of confidence that LIMS will continually operate as per design. Such activity is crucial in determining the amount and depth of testing that is appropriate to assure this and to meet current regulatory expectations. The test strategy should also examine the testing requirements for the Business Continuity (Contingency) Plan. For the test strategy to be valuable to the project, it must be performed in a structured manner which is formally documented. The pharmaceutical manufacturer should execute the test strategy with assistance from the LIMS Suppliers/Integrators.

The approach to testing should be documented in a test plan which identifies the scope of the testing and the documentation that is to be produced to act as evidence that the implemented LIMS meets the Business/User Requirements and regulatory expectations. Testing will typically cover two main areas:

- Module functionality and stress testing to demonstrate that the required LIMS configuration has been successfully implemented to meet the design expectations
- Integration testing to verify that the integration/interfacing of LIMS to other systems has been successful

All LIMS functions (e.g., manual data entry, automated data entry, and report generation) should be assessed to evaluate the effect that they could have on the data that will support regulatory submissions and release of product to market. This approach must be methodical to ensure that relevant functionality is not overlooked. It is essential, for instance, that Business Continuity Plans are verified as being appropriate and workable. This testing should cover all aspects of the recovery process from the loss of individual components (e.g., an analytical instrument interface) to the full loss of LIMS (e.g., representing a catastrophic failure of the LIMS server). The criticality of the data associated with the management of Electronic Records and Electronic Signatures should also be a focal point. However, it must be accepted that these assessments can be subjective and therefore they rely heavily on the experience of the assessors.

Prequalification Activities

To ensure that all the interconnection of system hardware and loading of software has been successfully completed on the target system prior to formally documenting the final validation, the installation must follow written plans and be formally documented. It is important that the LIMS Supplier/Integrator formally documents this process as part of good IT practice.

Due to the inherent complexity of demonstrating that LIMS is meeting expectations and the logistics of arranging for all the analytical equipment interfaces to be present at this time, it may well be the case that some simulation of data inputs or connections to other systems may be used.

The results of this will provide the first information for specific reference in the IQ protocol, for example version/identification details of the software, identification details of the hardware, and a list of any outstanding issues that were raised during the testing. This link will provide traceability through the testing process and will ensure that any failures and outstanding issues are addressed before the end of qualification.

If such an approach is to be taken, the supplier's installation work must be formally reviewed and approved by competent personnel, and the pharmaceutical manufacturer must also ensure that the completed integration testing documentation is of a quality suitable to support validation. The supplier's work should be witnessed, or at least the results reviewed by the pharmaceutical manufacturer, and a summary report produced.

The summary report will identify any issues raised and the pharmaceutical manufacturer will need to review this information to determine if LIMS is fit for purpose and therefore suitable for moving on to the IQ. Confirmation of acceptance of LIMS will normally be given only if all but minor issues have been resolved.

Installation Qualification

The Installation Qualification (IQ), as its name suggests, is a testing/inspection process that is designed to confirm the compliance of the LIMS installation with the agreed design and equivalence of the system used for testing. This IQ should cover all aspects of the hardware, firmware, software, documentation, environment, and infrastructure for the installed LIMS.

Clearly, before the IQ can commence, the installations should be performed and documented. The IQ will then be conducted to verify the installation of the component parts of the LIMS, which should be installed in their final location, and the completion of all interconnections/interfaces. At the point that the IQ is executed, it is essentially an indication that there is no further need to modify the system (hardware, software, configuration, and data).

Attempts are often made to "rush in" to start the IQ with the consequential problem of failure of tests/inspections (e.g., the incorrect version of software is installed, parts of LIMS are still missing, etc.). This not only has the effect of introducing retests but it also does not give a regulator, who may inspect the system, confidence that LIMS has been designed and installed by a quality aware organization. As it is the intention that the IQ will verify that the installation is correct and complete, no failures should be experienced. As there is ample opportunity for a pre-inspection to be performed by the LIMS installers, it is in fact the case that the validation personnel are simply rechecking and recording. It is essential that the integrated project team are all aware of the contents of the IQ test/inspection protocol to be used in the IQ; this will ensure that there are no surprises for the testers when this execution takes place.

The IQ protocol will cover LIMS hardware and software and all interfacing hardware and software within the scope of LIMS. It will also cover the environment into which LIMS and interface equipment are installed, which may need to be controlled in terms of temperature/humidity, electrical interference, etc. In addition, the provision of services will need to be assessed (e.g., electrical supplies). The typical contents of an IQ protocol are shown in Table 21.3.

The IQ is also the vehicle for confirming that the Design Review process has been successfully completed. It is essential that all installation and infrastructure issues are resolved prior to the completion of IQ. It is therefore useful to provide a check within the IQ protocol, which looks back at the results of the Design Review process and integration testing and assesses the effects of any issues raised regarding the compliance of the installed system.

TABLE 21.3
Example IQ Content

Subject	LIMS Hardware	LIMS Software
Software Versions for Operating Systems, Utility Software, Application Software		✓
Licenses for Core Software and Layered Software Products		✓
Server Operating System Build	✓	
Hardware Platform Details with Unique Identification (e.g., Serial Numbers)	✓	
Labeling of Hardware Platform Equipment (Including Interface Cabinets)	✓	
Diagnostics Self-Test for Analytical Equipment Interface	✓	
Network Compatibility of Peripherals (Printers, PCs, etc.)	✓	✓
Power-Up/Power-Down Tests	✓	✓
Installation of Services (e.g., Power Supplies)	✓	
Installation of Internal Wiring and Marking Up of Cabling for Maintenance, Confirmation That No Disconnected Wiring Present	✓	
Computer Room Environment Testing (Temperature, Humidity, Radio Frequency Interference, Electromagnetic Interference)	✓	
Network Connections to LAN/WAN	✓	✓
Security Access Testing	✓	✓

Installation Qualification Report

Following the execution of the IQ checks there may be exceptions where LIMS has not been installed in full accordance with the design agreed upon during the Design Review process. There are often issues due to hardware, documentation, drawings, environment, interfaces, installation, or even missing components of LIMS. These will result in failures in the IQ checks as LIMS does not comply with the requirements specifically noted in the IQ protocol acceptance criteria. In these cases each failure will be recorded in the Installation Qualification Report and a decision/justification made to

- Fix the noncompliance
- Accept that the corrective action may be deferred
- Accept that the issue will become a permanent feature of LIMS installation

Whatever the reason, the project team will need to assess the situation and determine an appropriate course of action. If the failure does not affect the operational testing (e.g., a specification is incorrect), this could be corrected in parallel with the operational testing. Otherwise the exception must be corrected prior to moving on to the next phase of qualification.

The Installation Qualification Report is therefore a milestone in the project which completes the IQ and records acceptance that any outstanding issues are of a nature that will not affect the integrity of the operational testing.

Operational Qualification

The OQ consists of a series of tests based on LIMS FDS. When the pharmaceutical manufacturer is happy that the installation of LIMS has been satisfactorily completed, the project will move on to the stage where the functionality of the “final system” will be demonstrated. Operational Qualification (OQ) is the vehicle for providing documentary evidence of the demonstrated LIMS functionality for the independent parts of LIMS prior to full integration. There is again a need at the beginning of this phase to review any issues raised during the Design Review process and IQ.

Any issues that will have an effect on the operational testing must be resolved prior to the commencement of OQ.

Interface Testing

It is widely accepted that a validated system must not receive data from an unvalidated system through an interface. The approach to the validation of the interface to LIMS must take into account the need for the validation of all data sources (e.g., analytical equipment, Chromatography Data Systems, etc.) and the interfaces used to obtain these data. It should be the case that the validation of interfaces follows its own validation lifecycle with the importance of the integrity and security of data being of primary concern.

Data Load

Test data will be needed for use during development testing and operational qualification. Test data should cover all of the data types, limit values, etc. That will be expected in LIMS and ideally should be based on a copy of data from a live LIMS. Where LIMS is installed to replace an existing LIMS, or LIMS is implemented to replace manual or semiautomated laboratory information management activities, there will be a need to transfer already existing data from another system or from manual paper-based sources to LIMS. Test data sets should be managed under configuration management.

To support data load, a process for verification and loading of data is required. This covers:

- **Dynamic Data:** information related to individual samples or lots (e.g., test results, sample status). This will be required if the implementation is an upgrade to an existing LIMS. The process should enable the extraction, cleansing, verification, loading, and maintenance of the "dynamic" data from the system being replaced. Dynamic data should be held securely under change control.
- **Static Data:** information which is not related to individual samples or lots (e.g., test methods, specifications, calculations, field formats). This process should enable the collection, cleansing, verification, loading, and maintenance of the "static" data (e.g., specification limits, calculations, test regimes, etc.). Static data sets should be maintained in a configuration control system.

The means by which these data are transferred must be validated or a 100% verification of the data must be performed. Data transfer may not be as simple as copying data from one database to another as there may well be different data fields or different field formats between the two systems. Data cleansing and archiving might also be necessary.

Operational Qualification Protocol

The OQ documents must cover all GxP-relevant functions in sufficient detail to provide the pharmaceutical manufacturer customers with a high level of confidence that LIMS operates in accordance with the agreed design. There should also be challenge testing applied to LIMS to attempt to stress the system and therefore, assuming the tests pass, build further user confidence in the installation.

It is recommended that the structure of the OQs should match that of the FDS to provide a simple mechanism for demonstrating that all of the functions of the FDS have been tested. Each test on OQ should also contain references to the functions they are demonstrating. If the structures of the documents cannot be linked, it will be necessary to provide some form of cross-reference document which will provide this information. The OQ may have one or more functional test scripts

TABLE 21.4
Example OQ Protocol Contents

Subject	Core LIMS	Custom Code
Special Configuration Functions	✓	✓
Testing of Bespoke/Custom Software	✓	✓
Signal Diagnostics from Linked Analytical Equipment Interface	✓	✓
Special Calculations and Algorithms	✓	
Verify Operating Manuals including Challenge Testing	✓	✓
Verify Sample Data and System Parameters, i.e., Check against Source Records to Ensure Accuracy of Data within the RDB; Involves Checking of Data Loaded Manually or via Automated Upload from Legacy Systems	✓	✓
Software Backups and Restoration of Data	✓	✓
Training Records of Users	✓	✓
Routine Maintenance/Calibration Routines	✓	✓
Provision of Service Level Agreements (SLAs)	✓	✓
Data Upload and Migration Checks	✓	✓
Data Integrity Checks (e.g., Range Checks, Validation of Inputs)	✓	✓
Communication Driver Tests	✓	
Database Structure and Population	✓	
Disk Shadowing Demonstration (Where Fitted)	✓	
Archive and Retrieval of Documents and Records	✓	
User Results Input and Displays	✓	
Analytical Report Generation	✓	
Audit Trail Verification	✓	
Demonstrate Features Supporting Use of Electronic Records and Signatures	✓	

for testing each of the functions that are uniquely identified in the FDS. Table 21.4 identifies typical OQ Protocol contents.

At the OQ stage there will be a need for the pharmaceutical manufacturer to ensure that the appropriate management systems in terms of procedures and Business Continuity (Contingency) Plans have been assessed and confirmed as suitable. In some cases it may be that as part of the OQ process draft versions of the procedures and Business Continuity (Contingency) Plans are developed with the revised documentation being issued following the completion of OQ.

Standard Operating Procedures

Standard Operating Procedures (SOPs) must be written to cover operational activities. The SOPs should be written by personnel knowledgeable in the low-level detail of LIMS, and should be detailed enough for the user to work without reference to other personnel or documentation, or memory. By using the OQ as a means of formally testing the SOPs, any issues of detail should be identified. The first version of the SOPs must be authorized and issued prior to the start of Performance Qualification (PQ).

Operational Qualification Report

Following the execution of the OQ tests there may be issues noted that the LIMS does not function in accordance with the design as agreed upon during the Design Review process. As for the Installation Qualification Report the project team will need to review these failures and determine a plan of action or a justification for moving on to the next phase.

Following the successful completion of the LIMS OQ as documented in the OQ Report, the LIMS software is ready for Performance Qualification to commence. At this stage it is normal practice to issue the first version of the Validation Report. This reviews and summarizes all of the qualification activities to this point. Authorization of this report indicates that the LIMS application is ready to be promoted into the production environment and made available to the trained operators.

Performance Qualification

Although the term Performance Qualification is not directly applicable to computer systems, there will be a requirement to monitor the ongoing operation of LIMS in terms of system performance and user interaction. The PQ consists of

- Monitoring the system performance through execution of the life cycle for each sample type, using the user SOPs, in the production environment
- Confirming that the coverage of SOPs is complete
- Monitoring of system incidents and failures
- Monitoring of change requests

The PQ documents that the integrated LIMS system and interfaces perform effectively and reproducibly using live data and user interaction in the production environment. As with the previous phases, the first part of the PQ will be to determine that there are no outstanding issues from the Design Reviews and IQ/OQ Reports which need to be addressed prior to the start of PQ.

The length of time over which a complex LIMS will be subjected to this PQ testing is typically 6 to 12 weeks, but each system will need to be assessed on a case-by-case basis to determine if this is an appropriate period.

It is an expectation of the U.K. MHRA that if LIMS takes over from a manual system, the new LIMS and the manual system should be operated in parallel for a period of time. This period is expected to be long enough to confirm that LIMS is fit for purpose to take over from the manual system. This activity may form part of the PQ activity.

Performance Qualification Report

At the end of the PQ a report will be produced which summarizes the continued operation of the LIMS for the initial period following going live. During the period of the PQ, LIMS is expected to have stabilized in terms of user support requests, system performance, and changes implemented as a result of the qualification process.

LIMS Validation Report

The final Validation Report for the installed and qualified LIMS reviews the results of each of the preceding validation phases. This report will act as a summary of the overall validation status of the entire LIMS. There may have been Validation Reports associated with individual items of analytical equipment and perhaps the core LIMS itself. The final Validation Report updates the initial version of the Validation Report and should cover the following:

- A summary of the results of each of the validation activities
- A summary of outstanding issues
- A summary of outstanding issues associated with the core LIMS
- Time scales for future periodic reviews of LIMS validation status
- Details of the justifications from the pharmaceutical manufacturer for any deviations from the original Validation Plan

- Statement to verify that LIMS is fit for purpose and the key operating limitations/validation boundaries

The validation report will be the document that will be utilized in the first periodic review to confirm that any actions recorded have been successfully addressed.

ONGOING OPERATIONAL COMPLIANCE

Ongoing maintenance of the LIMS validation status requires a suitable infrastructure to be in place. This infrastructure will consist of a LIMS manager/owner and appropriate SOPs. The LIMS manager will be responsible for controlling any changes to the system, interfaces, LAN/WAN architecture, LIMS functionality, and the data held within the database.

RESPONSIBILITIES

The LIMS manager is typically responsible for the daily administration of the entire LIMS (core LIMS database, LIMS servers, peripheral devices — e.g., printers, user PCs, networks, etc.). The manager must respond to user requests and problems in agreed-upon time scales and is in effect providing a service to the laboratory. The duties of the LIMS manager will include:

- Addressing user problems
- Adding and deleting users
- Controlling user privileges
- Managing upgrades to the core LIMS, standard software packages, and operating systems
- Managing Service Level Agreements with the LIMS supplier

As a result of LIMS manager controlling LIMS, the validated status will be monitored and maintained during ongoing operation. Where there is a need to make a change as a result of component failure, upgrade, or LIMS development, a change control system must be followed. Implementing a series of procedures and maintaining access control (both physical and electronic) to the core LIMS and interfaces will assist the maintenance of the validation status over the lifetime of the system.

CHANGE CONTROL

Change control of LIMS hardware, software, and associated documentation (SOPs and Operating Manuals) is necessary to prevent the system from becoming unmaintainable. A good change control system will allow the LIMS manager to determine what changes have been made to LIMS, when they were made, and what effect they had. It is not acceptable that changes be made to the LIMS functionality without the effect of the change being assessed against the current validated status and also current GxP. If the change is necessary and impacts on the validated status, then appropriate revalidation must be performed. This may result in rerunning one or more tests from the IQ, OQ, or PQ or, in the worst case, may result in a revaluation of the fundamental design of LIMS.

The LIMS change control system must record:

- Details of the change
- Authorization for the change
- Assessment of the effects of the change on GxP
- Details of the outcome of the Design Review
- Date when the change was requested and the date when it was implemented

- Testing performed to verify the operation and reconfirm the validated status and details of the test results

Where a change is required to the LIMS hardware due to the failure of a component, there are two possible scenarios. The first is that the failed component is no longer available and a new design must be installed. In this case a Design Review process will be required to assess the effect on the rest of LIMS, followed by the normal testing approach. The second is that the component is a standard offering from the supplier and is therefore a "like for like" replacement. In this case simple testing of the functionality of the replaced component is all that is required.

The change control system will be utilized in the maintenance phase of the LIMS life cycle. However, the level of details of the review and the rigor of the testing should be the same as was used in the original validation process. The testing must therefore be carried out by competent qualified personnel and the records of the testing retained as part of the LIMS validation support documentation.

Upgrades to LIMS

Following the installation of LIMS the core LIMS software will continue to be developed by the LIMS supplier to fix known bugs and implement new features. This means that the LIMS manager will be routinely advised that there is a need to change to the latest version of software. In some cases the LIMS Supplier Service Level Agreement may be linked to the installation of software upgrades. As a consequence of this LIMS, manager will be responsible for the review of the validation status of LIMS, taking into account the effect of the existing validation documentation of the change to the LIMS. Any change to the validation status is likely to involve updating documentation as well as the software and subsequent testing. The approach to change should be as follows:

Software changes: Assess software updates for compatibility with the existing software with particular emphasis on any changes made. In terms of system or standard software products, there is normally a "bug fix" list and details of new, modified, and removed features. Supplier's documentation should include an analysis of the impacts of the patch on their system, and this should be used by the customer to assess the extent of testing required to validate the patch implementation. These documents should be assessed to determine the effects of the change and any appropriate testing performed and incorporated into a validation dossier.

Hardware changes: Assess new hardware for compatibility with the existing hardware. Should any differences be identified, these should be assessed/tested against the existing LIMS design intent.

Documentation changes: The document changes will normally be as a result of modifications to the LIMS hardware, software, or operating methods. The design documentation supporting the maintenance of LIMS should be updated to reflect any changes. It is not acceptable that any documentation that is to be utilized for maintenance purposes be out of date.

Software Bug Fixes and Updates to Core LIMS

Upgrades due to developments, including the installation of bug fixes (patches), must be tightly controlled by the personnel responsible for LIMS. Following the production of the Validation Report, LIMS is deemed to be in a validated state (subject to any issues raised in that report). Changes at the operating system, database system, and application levels must be assessed for impacts on the validated status of LIMS.

Examples exist of systems implemented only a few years ago for which replacement parts and software enhancements are no longer available, let alone the relevant software skills to make one's own bug fixes or developments. Choices must be made whether to upgrade in an evolutionary way, taking into account the cost of buying and implementing the next generation of LIMS, or to upgrade by replacing the present system. Management of the LIMS upgrade path is a key skill in protecting the integrity of the existing LIMS validated status and avoiding maintenance problems in the future.

It is also important that in all the changes described above, the Quality function within the pharmaceutical manufacturer's organization provide sign-off that any change has been performed in a manner that maintains the validated status of LIMS.

LIMS Backup and Restore

Backup and restore applies to the application code, the configuration, and the static and dynamic data. The objective is to be able to recover the system following a crash or other catastrophic event.

LIMS must be backed up on a regular basis to maintain the security of the database. The regulatory inspectors will not accept that data within the database has been lost due to the failure of the server or other incidents (e.g., fire or flood). In order to prevent losses the pharmaceutical manufacturer is responsible for implementing a reliable, robust, and documented backup regime. The frequency of backups must be assessed as part of the GxP assessment process as this will be determined by the frequency at which data will be entered into the database and the redundancy features in the server, and should take into account the risks of data loss. Automation of the backup regime is acceptable provided that it is validated normally as part of the operational testing phase of the project.

A robust and secure process must be implemented for storage, renewal, and eventual destruction of backup copies. This process should be documented as part of the Backup and Restoration SOP. It is important that it can be demonstrated to the regulators that appropriate personnel are managing this.

In order to demonstrate that the data backed up is capable of being restored in the event of a breakdown, and the customer organization must implement a periodic test of the restore procedure. This procedure will demonstrate that data restoration is possible and that the procedures covering this activity are effective. However, this testing must not compromise the integrity of the production data, and it is recommended that the testing take place on an off-line version of the LIMS (e.g., the test or validation instance).

Data archiving: Archiving functionality is required to manage growth of the data within the LIMS database. The policy and process for archiving must be documented. It is important that it could be demonstrated to the regulators that appropriate personnel are managing the transfer of data to the archive. To meet the requirements of the FDA Electronic Records and Signatures regulation 21 CFR Part 11, the archiving process must address the retention periods for GxP-critical data created and stored on LIMS.

LIMS Security

The implementation of an effective security regime is required to comply with the regulators' expectations for control of electronic records. The pharmaceutical manufacturer will be responsible for providing maintenance of the security aspects of LIMS. This is normally accomplished through software protection (e.g., passwords and log-on accounts) but may also take the form of protection through physical restrictions (e.g., locked-up or restricted areas). The management of this function should be in accordance with a formal SOP. The use of passwords and high-level accounts must be strictly controlled to prevent security breaches. Typical examples of control should be:

- Restricted number of high-level users.
- Unique IDs to provide traceability of personnel making changes or signing electronic records.
- No shared user identifiers.
- QA should authorize all users.

- Where electronic signatures are used, they must be the equivalent of handwritten signatures.

Business Continuity (Contingency) Planning

Business Continuity (Contingency) Plans define the controls that minimize the impact of temporary or long-term loss of all or part of LIMS. The extent of planning will be determined by the criticality of LIMS with respect to the GxP operations that it controls or monitors and the data it manages. Business continuity, in the form of standby systems and manual ways of working, should be considered during the development phase for any highly critical computerized systems. However, it is also vital that plans are established that assume the inevitable, i.e., what can go wrong will go wrong. Plans will need to define the requirements for system archive, periodic backup, restoration procedures, and service level agreements. Additionally, plans must address the method of system (which may mean temporary use of a system at a remote location) and data recovery, and defining the manual operations that may need to be applied in the interim until LIMS is reinstated.

Training

Training of LIMS users is a key issue that is likely to be the subject of a regulatory inspection. It is therefore essential that a training program be organized as part of the LIMS implementation and maintained as part of the ongoing maintenance of LIMS. Demonstrated training is required not only for LIMS users but also for the in-house support and development staff. The assessment of training not only applies to the pharmaceutical manufacturer's personnel but also, through audit, to the LIMS Supplier/Integrator's personnel and the validation personnel if they are independent of the LIMS Supplier/Integrator. As part of the Supplier/Integrator evaluation the pharmaceutical manufacturer must assure that competent trained personnel are to be utilized on the project. It is recommended that evidence of this training be provided for reference in the project validation documentation, perhaps in the form of staff résumés or copies of training records.

Periodic Reviews

The validation integrity of LIMS must be periodically reviewed to ensure that ongoing support systems are effective. The review process should be designed to identify trends that may indicate noncompliance with support procedures or weakness in the original validation exercise. The review should further examine the original test data sets to determine their applicability to the current computer system configuration and duty. The review shall determine if there is a need for further validation of the current LIMS installation.

These reviews should typically include an assessment of:

- System performance
- Maintenance records
- Backup records
- Change control records
- Access privileges
- Network integrity
- SOPs
- Fault reports
- Supplier audit follow-up

CONCLUSION

This chapter has reviewed one approach that may be taken to the validation of a typical LIMS that will meet the expectations of the regulatory authorities. If LIMS is to be utilized to support pharmaceutical manufacturing laboratories, the system must be subjected to validation. It is recommended that a life-cycle approach be adopted for validation based on GAMP.[1]

One of the first activities in the validation process is the GxP assessment, which will identify the GxP-critical aspects of LIMS. These aspects must be fully assessed, designed, tested, and documented using a process with a clear audit trail. The GxP assessment will allow the pharmaceutical manufacturer to ensure that effort is concentrated where it is most needed. It is essential that the personnel involved in the validation process are knowledgeable and experienced in the validation of LIMS. This may mean that there is a need to ensure that the LIMS Supplier/Integrator can provide suitable validation personnel or that independent specialists are used. The cost of the system is therefore not the only concern when choosing a new LIMS; if the LIMS Supplier/Integrator is not able to validate the system properly, it will be useless to the pharmaceutical manufacturer. Once a validation project is complete, the system should be under the control of a LIMS manager who will be responsible for ongoing operational compliance. Ongoing operational compliance not only covers the issues of maintaining the validation documentation and managing all aspects of change control, but also refresher training for existing personnel and ensuring that any new personnel are fully trained.

REFERENCES

1. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (ISPE), (www.ispe.org).
2. Medicines Control Agency (MCA) (2002), *Rules and Guidance for Pharmaceutical Manufacturers and Distributors 2002, Annex 11: Computerised Systems*.
3. United States Code of Federal Regulation Title 21, Part 11, *Electronic Signatures: Electronic Records*.
4. United States Code of Federal Regulations Title 21, Part 58, *Good Laboratory Practice for Non-Clinical Laboratory Studies*.
5. American Society for Testing and Measurement — Standard E1578, “Standard Guide for Laboratory Information Management Systems [LIMS].”

22 Case Study 4: Clinical Systems

Chris Clark, Napp Pharmaceuticals
Guy Wingate, GlaxoSmithKline

CONTENTS

Current legislation and regulatory guidance for the management and conduct of clinical trials are undergoing significant changes, and the European and U.S. regulatory bodies are increasingly focusing greater attention on the compliance of the pharmaceutical industry to these regulations.[1–3] Recent observations noted during both regulatory inspections and company vendor audits have indicated that one area of critical noncompliance, and a potential barrier to successful license applications, is that of the development, implementation, management, and controls applied to

TABLE 22.1
Example Clinical Systems

Functional Subgroup			
Data Capture	**Data Processing**	**Production Control**	**Record Management**
Subject Information Systems	Data Management	Clinical Trials Supplies Production	Protocol Management
Interactive Voice Response Systems	Randomization Systems	Analytical Instrumentation	SOP Management
Clinical Trial Data Collection Systems	Clinical Trial Review Tools	Inventory System	Electronic Publishing & Regulatory Submissions
Medical Device Measurement Systems	Statistical Analysis Systems	Labeling System	Electronic Document Management Systems
Optical Character Recognition Systems	Pharmacovigilance Systems	Environmental Monitoring	Training Records
Bar Code Readers	Electronic Data Transfer Systems	Product Tracking/Monitoring	Archive Repository

the use of computerized systems in the GCP environment. Companies invest large amounts of time, resources, and finance into the process of developing, investigating, documenting, and registering new products — a process that can take upward of 10 years to result in a successful launch to the marketplace. There are many stages during this process whereby the new product can fail: for example, by not demonstrating adequate/beneficial therapeutic value, by the presentation of adverse side effects, or by not being capable of being formulated into a delivery system suitable for mass production.

The above listed risks are well understood by those experienced in the new product development process, and they place great reliance upon the collection, manipulation, and presentation of data. It is this very reliance on data that places yet another question of risk in our pathway. The risk raised here is that presented by the failure of our computerized systems involved in these processes to infer adequate integrity and security to this hard-won data. Regulatory submissions based upon data that has been handled or managed by systems that cannot demonstrate adequate controls around integrity and security will, in most likelihood, be rejected by the assessing authority as being unreliable. This avoidable scenario, which inevitably results in loss of revenue through the need to repeat expensive clinical trials, and the consequent delays in getting a product to the marketplace can be prevented by the application of sound computer systems validation practices.

COMPUTERIZED SYSTEMS AND THE CLINICAL TRIALS PROCESS

There are a wide variety of computerized systems involved in the clinical research process, from small stand-alone desktop systems to more complex enterprise-wide systems. Furthermore, they can often be categorized into four functional subgroups. Table 22.1 provides some examples of such widely divergent systems.

GENERAL VALIDATION REQUIREMENTS

Clinical studies may be supported by computer systems in a number of ways from data capture, data processing, production control, and document management. Some systems may be complex, others simple. Some systems may be custom-made, others based on COTS products. Whatever the character of a clinical computer system, the same basic GCP/GLP principles apply. All computer systems that play a part in the conduct or support of clinical studies intended for regulatory submission, therefore, need to be validated. It is vital that such systems manage clinical data reliably and securely.

Validation of clinical research computer systems should demonstrate that the computer system is suitable for its intended purpose.[4] Validation is achieved through a life-cycle approach to computer system development, operation, and maintenance. The various international GCP/GLP requirements also emphasize the importance of data integrity. This covers data input, manipulation, output, and archiving.

This chapter is based on work published by ACDM/PSI.[5] General GCP/GLP computer validation requirements are reviewed. This is followed by a summary of key topics for a selection of common systems found in the clinical environment.

The basic computer compliance requirements for development and installation of clinical systems can be summarized as follows:

- Validation of data processing software prior to use
- Auditing suppliers of software-based systems
- Assessment of the investigator site prior to the start of the trial, including investigator-supplied software-based systems considering such issues as validation activities (planned and completed), level of understanding of GCP requirements for use of computerized systems, statement of level of compliance with 21 CFR Part 11, the presence of remediation plans if required, and calibration and maintenance procedures

The capture, processing, and retention of data should be carefully defined and managed. The ICH GCP Guidelines,[3] which recognize that clinical trial data can take many forms (paper, optical, electronic), indicate that the Sponsor has specific responsibilities regarding the handling of electronic data and the use of remote electronic clinical trial data systems (subsection 5.5.3). Such responsibilities include:

- Validation of the system
- Ensuring that SOPs covering usage are in place
- Maintenance of an audit trail for data changes
- Adequate security systems in place to prevent unauthorized access
- Control of user access rights
- Data backup and recovery procedures
- Maintenance of blinding during data entry and processing

U.K. GLPs suggest systems that organize, tabulate, and subject the data to statistical or other mathematical procedures, or that otherwise manipulate or analyze electronically stored data, and permit the retrieval of original data entries.[4] All network communication links used for data transfer should be considered potential sources of error and controlled appropriately.

OECD regulations have further identified the following operation and maintenance requirements:[6]

- Procedures for operation and use of computerized systems (hardware and software) and the responsibilities of personnel involved
- Procedures for security measures used to detect and prevent unauthorized access and program changes
- Procedures and authorization for program changes and the recording of changes
- Procedures and authorization for changes to equipment (hardware and software), including testing before use if appropriate
- Procedures for periodic testing for correct functioning of the complex system or its component parts and the recording of these tests

- Procedures for the maintenance of computerized systems and any associated equipment
- Backup procedures for all stored data and contingency plans in the event of a breakdown
- Procedures for the archiving and retrieval of all documents, software, and computer data
- Procedures for monitoring and auditing the compliance of operational computer systems

Where system obsolescence forces a need to migrate electronic raw data from one system to another, a process must be validated to ensure integrity.[4] If such migration is not practicable, the raw data must be transferred to another medium (e.g., paper, microfiche), and this verified as an accurate copy (i.e., content and meaning are preserved), prior to any destruction of the original electronic records.[4,7]

RESPONSIBILITY OF GCP/GLP QUALITY UNIT

The Quality Assurance organization has no mandated role in the development of computer systems other than defining QA functional requirements.[8] Once the system has been validated, accepted, and installed, QA will be responsible for monitoring data collection until its reliability is confirmed in accordance with SOP, compliance of user SOPs, training, and security policies.[8] Any performance problems should be communicated to the responsible management personnel in a timely fashion. QA should monitor corrective actions and unscheduled downtime records.

The British Association for Research Quality Assurance (BARQA) has interpreted international GCP/GLP regulations and expects QA personnel to:[9]

- Conduct GCP/GLP awareness training, validation training, and change-control training
- Review and approve validation and change-control procedures
- Review quality plans and key validation documents (i.e., Validation Plan, Requirements, Test Plan, Test Results, Acceptance, Record Retention [Archiving], Change Control)
- Advise projects on software development
- Review changes (individually or as part of periodic review process)
- Conduct system audits (including system development, software, operation, and use)

In addition, QA commonly provides general consultancy and advice on the interpretation of regulatory requirements for computer compliance.

DATA CAPTURE SYSTEMS

Subject Information Systems

Electronic Diary Cards are portable, hand-held systems designed to be programmed according to specific protocol requirements and are used by patients to record directly information on their condition and medication consumption during a particular study. They should be specified and designed so that they are highly prescriptive since they are used in a relatively uncontrolled environment (e.g., subject's home). Specific considerations for the validation of electronic diary cards are:

- Suitability for use by the target patient population
- E-functionality, e.g., time of data capture, checks for logical consistency, data
- Confirmation and auditability, provision of investigator signature
- Supplier auditing
- Usability, robustness, and integrity of both software and hardware
- Tamperproof software, i.e., modification for other purposes should not be possible
- Power backup in the event of expiry or removal of batteries
- Security, controlled by password, including access restrictions and integrity of data

- Transfer of the diary data to the host database, including any data modification, annotation, or processing — occurring before, during, or after the transfer
- Documented training of site personnel and individual patients

Systems are also required to record whether or not all dispensed medication for a clinical trial can be accounted for at the end of the study. For each subject in the trial the amounts of dispensed medication are compared with the amounts consumed and the amounts returned. The returned supplies are then destroyed and certified as such. The amounts dispensed may come from a pharmacy system and the percentage consumption within a dispensing interval could be derived as a measure of subject compliance. Specific points to consider during validation are therefore:

- The incorporation of any derived data algorithms
- Electronic transfer from and to other systems

Interactive Voice Response Systems

An Interactive Voice Response-System (IVRS) is a communications platform based upon the telephone network used to coordinate key clinical trial activities and provide real-time information for study managers. By utilizing the telephone network, the system provides for a direct connection between the clinical trial patient and the central study-specific database. This permits the collection of data in response to preprogrammed prompts from the system, ensuring the recording of key trial events and the provision of information critical to the successful conclusion of the trial. Most IVR systems are individually tailored to each specific study based upon requirements defined by the Sponsor.

Specific considerations for the validation of IVR systems are:

- Formally agreed and documented sponsor requirements
- Formally documented design specifications
- Validation planning for sponsor-specific project
- Supplier auditing
- Traceability between design and testing
- Formal change-control system for data and system
- Validation of the data transfer process to the sponsor database

Medical Device Measurements

Medical devices used to take clinical trial measurements must comply with medical device regulatory requirements. These cover design controls and software validation.[10] Another case study in this book deals with medical device validation (see Chapter 40: Case Study 22).

Optical Character Recognition (OCR)

OCR systems recognize images as alpha-numeric data, as if the data had been entered directly from a keyboard. They do this via recognition engines, operating by template matching, feature extraction, neural networking, or a combination of these approaches.

There is an explicit reliance on operator involvement in the verification of the captured data, whereby the software presents the operator with uninterpretable input image for manual resolution. Validation needs to take account of all dimensions of the system, testing with a sufficiently varied selection of input image. Specific considerations for the validation of OCR systems are:

- Supplier auditing
- Reliability, calibration, and maintenance of scanners

- Correct identification by the system of the type and number of scanned input forms
- Functionality, e.g., substitution, learning capacity, verification
- Reliability of interpretation of the specified field images by the recognition engine
- Handling of indeterminate data
- Training and competency of the operator

Bar Coding Systems

A bar code is a pattern of dark bars separated by spaces. The bar code is read by passing a beam of light over it. Light is absorbed by the bars and reflected by the spaces. The differences in reflection are sensed by the scanning device (e.g., light pen, hand-held scanner, flatbed scanner) and converted into electrical signals corresponding to the widths of the bars and spaces which can then be decoded into the numbers and letters represented by the bar code. There are a number of different bar coding standards.

Specific considerations for the validation of bar coding systems are:

- The system used for creating the bar code labels, e.g., acquisition of number, conversion of number to bar code
- Print quality of the bar code, e.g., specks of ink in the spaces, edge definition of the bars, and print contrast between the bars and spaces
- Presentation of the bar code to the scanner, e.g., creased labels, protective covering
- Robustness and maintenance of the scanning device
- Verification of decoding
- Control of the reuse of preprinted bar codes

Data Processing Systems

Data Management Systems

Clinical data management systems can be used in a wide variety of applications. At the study level, it should be possible to set up the database efficiently to allow easy access to the data. Where the functionality is available, points to consider during the validation include the following:

- Entry screens function as expected (e.g., range checks, look-up tables, auto-encoding using the appropriate dictionary, derived data calculations).
- Entry screen fields correctly relate to database fields and fields are correctly defined in terms of format (e.g., character/numeric, length).
- Entry of confirmed missing values is possible.

Data entry should include identification of individuals using a combination of user-ID and password at the start of the data entry session. Automatic log-off is appropriate for long absences of individuals during operator sessions.

The validation of data capture should include the following:

- Verification. If part of the transfer process, it should result in discrepancies between two manual entries being correctly identified, and their subsequent resolution should result in one correct entry on the database.
- The transfer process should enable single entry of certain data, e.g., electronic laboratory data.
- Data should be loaded into the correct location, i.e., table and field.
- The transfer process should detect duplicate records.
- The user should be notified of nontransferred data.

- Any data identifiers should be correctly assigned.
- The date and time of initial loading of each data item to the database should be recorded by the system, i.e., the audit trail should commence at initial loading.
- Any auto-encoding, if part of the transfer process, should function as expected using the correct dictionary for each coded variable.

Data checking should include edit checking, plausibility checking, range and consistency checking. Any data derived should be validated. At the system level, points to consider include:

- Libraries of standard data checks should be accessible.
- Standard data checks should be adaptable for specific needs.
- Study-specific data checks should be possible.
- Study-specific checks should be correctly incorporated, with standard checks, into the study-specific editing functionality.
- The checks should be executed correctly, i.e., the correct checks should be applied to a data item at the appropriate time.
- Data items accepted following a failed data check should not fail again unless the data change.
- Failed data checks should remain flagged until resolved.
- At the study level the set of specified edit checks should be tested using tailored dummy data to ensure the absence of false positive and false negative failures.

The management and use of the system, and related reference data (e.g., laboratory reference ranges and coding dictionaries), should be controlled by Standard Operating Procedures. Such procedures should include taking data extracts, possibly as predefined reports. Extracts should be validated to demonstrate they correctly identify, combine/merge, and report data requested. At the system level the functionality of a reporting system should ensure that:

- Template programs are available for easy adaptation.
- Study-specific programs can be easily developed.
- Program development takes place in a separate environment from the use of validated programs.
- Documentation of output should include source program, date and time generated, user, page number, and total number of pages.
- All programs, and subroutines or macros called within programs, used to produce formatted output for clinical reports should be validated.

Validation should ensure that the facility for locking/securing the database prevents unauthorized write access. The unlocking of a database should be strictly controlled by an SOP.

Randomization Systems

Randomization systems, which are usually used by statisticians and pharmacy staff, may provide any of the following:

- A list of random numbers
- Code-break envelopes
- Packaging labels for drug supplies
- An electronic file of the patient treatment codes to be incorporated into the study database after it has been locked

Validation of the randomization system should be rigorous as randomization codes and code-breaks, and their security, are key to maintaining the integrity of any clinical trial. The codes and code-breaks, generated prior to the start of the trial for the packaging of medications, will not be linked to the data until the end of the study when the clinical database has been locked, which may be several years after the codes were produced. The following points should be considered during validation:

- The source of the core random number generator and its validation status
- The ability to reproduce the randomization schedule
- Storage of randomization codes and code-breaks, and access control
- Backup and restoration procedures, and their regular testing

Clinical Trial Review Tools

Validation of computer aided review tools, which may be used by in-house or regulatory reviewers to explore the project database on a read-only basis, should address both the system and project-specific aspects.

The underlying code of the generic shell that comprises the tool should be developed according to the software development life cycle. Depending on the degree of sophistication of the system, testing should cover the following areas of functionality:

- Selection of compound
- Selection of trial
- Display of raw data
- Subsetting of data
- "Point and click" cascading menus (i.e., increasing or decreasing the level of detail or subsetting)
- Search facilities
- Display of graphical results
- Linkage between annotation facilities (for sponsor and reviewer) and related data
- Transfer of data between different software systems (e.g., from the SAS package to a spreadsheet)
- Analysis and reporting

Testing for correct project and study setup should demonstrate that the data have been loaded into the system correctly. This will involve checking:

- Completeness, correctness, and consistency of the labels and formatting
- Correct functioning of the screens
- Consistency of the viewed data with the project database
- Consistency of reports and views of data output to the screen with clinical trial report tables, listings, and original

The different ways of viewing data may be too numerous to test exhaustively. Validation requirements, therefore, need to be realistic to ensure an appropriate level of overall confidence.

Statistical Analysis Systems

The statistical software systems used for analysis of clinical trial data can range from custom programs for specific statistical techniques to COTS packages. Such packages (e.g., the SAS system, SPSS, S-Plus) provide the user with a library of statistical procedures (e.g., analysis of variance, regression, generalized linear modelling, nonparametric methods) which can be accessed either by using the native programming language or by selecting the required options from the package's user interface.

It is generally considered that there is no requirement for validation of statistical packages such as the SAS system as entities in their own right. Nevertheless, any custom program written using the package's native programming language should be validated.

The supplier-supplied installation tests should be performed and documented to ensure that the software is functioning correctly within the specific operating environment. In addition, a suite of supplier-supplied programs, test data, and results can be a valuable aid to validation. Repetition of all these tests should be considered each time there is a change to the operating environment.

These include one-off and standard programs and macros developed using either a nonstatistical programming language (e.g., Fortran) or the native programming language of a COTS statistical software package (e.g., SAS programs, SAS macros, SAS/AF applications).

It should be shown that statistical procedures and functions (e.g., SAS PROCS), supplied as part of a COTS product, are used correctly within the context of the program. Software that automates the data analysis process across a number of clinical trials should be validated in the same way as other supplier or custom (bespoke) systems. However, the validation requirements for trial-specific, one-off programs written using COTS package native languages are reduced. Specific issues to consider during validation are:

- Statistical competency of the developer
- Precision and rounding errors
- Handling of missing data values
- Handling of unequal (unbalanced) treatment groups
- Handling of ties in nonparametric analyzes
- Facilities for checking the underlying assumptions of the statistical model
- Facilities for excluding outlying observations from analysis
- Printing of intermediate values during calculations
- Statistical competency and training of the users
- Operating environment and conditions

Pharmacovigilance Systems

Pharmacovigilance systems capture, store, process, maintain, classify, and report adverse event data. Any such systems generating reports for regulatory authorities (e.g., expedited reports, periodic safety updates) and the interfaces into them from a variety of sources, should be validated. Specific considerations when validating these systems are:

- Reconciliation of adverse event data from the clinical trial database, and other sources, with the pharmacovigilance database through electronic interfaces
- Development of programs to generate reports for regulatory authorities, e.g., expedited and periodic reports
- Assurance that all cases known to the system have been appropriately reported in the appropriate time frame
- Electronic transfer to regulatory authorities

Electronic Transfer of Data and Software

Clinical data and software may be transferred electronically, by diskette or direct line, on a routine basis from investigator sites, contract research organizations or central laboratories to the company (and vice versa), between different company locations, between computer systems within a location, and from the company to regulatory agencies.

Specific considerations for the validation of electronic transfers are:

- Internet, intranet, and other communication technologies (e.g., groupware, modem-to-modem, cellular technology)
- Externally owned lines
- Communication medium (e.g., diskette)
- Security (e.g., encryption, passwords, virus protection, "firewalls")

Specifications of the transfer file include:

- File format (e.g., ASCI I, comma delimited)
- Size of file
- Number of records
- Linkage of comments to numeric data
- Recovery following interruption of transmission
- Corruption during transfer
- Consistency of electronic file with source
- Backup and disaster recovery in both the sender and receiver locations

Production Control Systems

These computer systems should be validated to the same standards as expected for manufacturing control systems. Reference can be made to the other case studies in this book, as applicable:

- Clinical trial production (e.g., kilo laboratory production systems)
- Analytical instrumentation
- Inventory systems
- Labeling systems
- Environmental monitoring
- Product tracking/monitoring

Record Management

Protocol Management

Protocol management may include any or all of the following features:

- Controlled protocol authoring
- Electronic storage of protocols or data, either scanned in or created electronically
- Controlled distribution of protocols to, and retrieval by, multiple users
- Review and approval of protocols, e.g., within a workflow component
- Publishing of approved protocols
- Index generation
- Retrieval of indexed documents

The validation issues include:

- The life span and characteristics of the storage medium used, including the frequency and type of testing required
- The security levels of the protocols, including process-specific security such as that used for electronic signatures

- Version control of protocols including audit trail
- Validity period of printed/published protocols, e.g., SOPs
- The qualifications, training, and competency of users
- Indexing functionality

In most cases, there will be a requirement for a protocol to be appropriately approved and signed off. Options include the scanning and storage of the signed document, scanning and storage of the signatures associated with an electronic document, and the use of electronic signatures. It is important to define the "master" version (i.e., as paper or electronic). Signatures should be verified and stored with associated protocols.

SOP Management

Management systems for SOPs should establish and validate:

- Workflow for approval of SOPs
- Electronic records and electronic signatures
- Facility for user requests to change an SOP
- Storage of forms and templates in original software
- Storage of previous, current, and under-revision versions
- Controlled distribution

Specific validation issues include:

- Access security, especially write access to approved SOPs
- Documentation of notification of new/revised SOPs to all appropriate staff
- Version control
- Integrity of the system, especially when replicated across servers
- Control of printed versions of SOPs

Issues to be considered during validation include:

- Testing of all possible routes to ensure that a document does not become suspended within the system
- Testing of parallel tasks to ensure that the result of those tasks is the same regardless of their sequence in real time
- Linkage and preservation of electronic annotations
- Corruption of the master document by annotations
- Printing of document and annotations

Other validation requirements that may be applicable have been discussed under protocol management systems.

Regulatory Submission and Electronic Publishing Systems

Electronic regulatory submissions combine components from specific systems, e.g., computer-aided review tools and electronic document management systems. Electronic publishing systems assemble electronic documents and images into electronic dossiers. The validation requirements of the publishing system, over and above the requirements for each component system should be assessed.

Electronic Document Management Systems

Document management refers to procedures or systems designed to exert an intelligent control over the creation, management, and distribution of documents. Electronic document management systems (EDMS) may include any or all of the following features:

- Controlled document authoring
- Electronic storage of documents or data, either scanned in or created electronically
- Controlled distribution of documents to, and retrieval by, multiple users
- Review and approval of documents, e.g., within a workflow component
- Publishing of approved documents
- Archiving of documents for completed projects

Typically, EDMS may need to address a range of issues including version control, access control, organization and management, workflow, imaging, publishing, document reuse, indexing, and searching.

The document types that may be stored within the system may have a wide variety of file formats and sources, and range from just key documents to the totality of documents generated for a project. For each type of document stored in the EDMS, it is important to define the "original" version (i.e., as paper or electronic).

The validation issues for EDMS include:

- The life span and characteristics of the storage medium used, including the frequency and type of testing required
- The security levels of the documents and the system, including process-specific security such as that used for electronic signatures
- Version control of documents including audit trail
- Continuing readability of documents through technological changes, e.g., the use of Portable Document Format (PDF) file type
- Validity period of printed/published documents, e.g., SOPs
- The qualifications, training, and competency of users
- Indexing functionality

Optical images may be produced by scanning in a paper document or a faxed image into the system. Apart from general configuration and installation requirements, specific validation considerations should include:

- Procedures for calibration and maintenance of scanners
- Definition of the master record, i.e., paper version or electronic image
- Routing of images to appropriate locations
- Interfaces with other systems
- For fax to image, correction of transmission errors
- Readability of retrieved images
- Image quality prior to destruction of the original document
- Search-and-sort capability

Training Record Systems

Regulatory authorities do not generally inspect these systems; instead they inspect individual training records. Such systems should, however, be validated to ensure their reliability and performance. Specific validation issues to be addressed include:[11]

- Testing record retrieval times
- Backup and recovery procedures
- Electronic record/signature controls

Archive Repository

The repository may range from a specific directory on a server, with a work group password protection, to a software package-controlled database repository implementing full database security controls. There may also be a requirement to produce and store multiple renditions of a document within the repository. Specific validation issues include:

- Continuing readability following software upgrades
- Integrity of the document during conversion
- Production, storage, and retrieval of multiple renditions of a document
- Storage of signatures associated with documents

An important part of any electronic archive system is a policy or SOP which will affect the validation effort, including:

- Which documents should be kept in hard copy form and which may be kept only in electronic form
- How long documents are maintained on the system
- How long printed hard copies of documents are kept
- The need for off-site electronic backup when hard copies of documents are destroyed
- The possible uses of the documents, including whether they may be required by a court of law
- Access to the archive
- Storage criteria for electronic media and any special considerations, e.g., refreshing tapes/disks

E-MAIL AND INTERNET TECHNOLOGIES ISSUES

Clinical data and software may be transferred electronically, by diskette or direct line, on a routine basis from investigator sites, contract research organizations, or central laboratories to the company (and vice versa), between different company locations, between computer systems within a location, and from the company to regulatory agencies.

Today the most common method of direct line transfer is likely to be an e-mail message attachment via an Internet link. This has several problematic issues, the most significant being:

- Lack of audit trail data
- Concerns over system administration
- Robustness of e-mail systems
- Security during data transfer

Specific considerations for the validation of electronic transfers were discussed earlier in this case study.

ELECTRONIC RECORDS AND ELECTRONIC SIGNATURES

This case study will not discuss electronic record/signature controls in detail because these are examined in detail elsewhere in this book. Nevertheless, key points are summarized below.

U.S. 21 CFR Part 11 Regulation

Since it became effective, there has been much discussion about the impact of FDA 21 CFR Part 11,[12] in particular in the manufacturing arena, with some progress being made toward achieving full compliance. However, it is generally accepted that such progress has been slow to commence within the clinical trial domain. Even today, several years after the rule became effective, it is more likely that any assessment of a computerized system for compliance with 21 CFR Part 11 will result in a number of issues being identified as falling short of what is required.

In general 21 CFR Part 11 does not introduce anything radically new to the debate about what requirements should be placed upon systems used for GCP processes. Most of the requirements detailed within the rule are essentially good IT systems and electronic records practice, with some additional emphasis placed upon controls for electronic signatures. What should be recognized is that compliance with the rule is not simply one based upon a technological approach. Many compliance issues are related directly to putting in procedural systems to support any technology introduced. Hence there is a not unreasonable expectation by the FDA that many of the procedural systems will have already been introduced within organizations, and where gaps exist, plans for remediation will have been drawn up.

Specific 21 CFR Part 11 considerations regarding the validation of clinical research computerized systems include:

- Changes to data/software
- Audit trail — design and integrity
- Audit trail — paper vs. electronic
- System controls for electronic signatures
- Event logging — date and time stamp synchronization
- Procedural controls for granting access and permissions
- Company security policy and action on detection of fraudulent activity
- Personnel training/understanding of electronic signature authority/responsibility

Other International Regulatory Requirements

Complete copies of records must be available for inspection, review, and copying by regulatory authorities.

Audit trails should be generated to log the creation, modification, and deletion of electronic records. From this information it should be possible to reconstruct the electronic records as they existed at any date and time in the past. It should be possible to associate all changes to data with the person making these changes. Audit trails, therefore, need to be time-stamped with date and time changes that were made. Computer systems should provide for the retention of full audit trails to show all changes to the data without obscuring the original data.[4] Consequently, audit trails need to be protected such that no direct modification of the stored information can be made.

In addition to the requirements defined in 21 CFR Part 11, the legal requirements surrounding the attachment of a signature to an electronic record is controlled under EEC legislation in the form of Directive 1999/93/EC — a Community Framework for Electronic Signatures.[13] The most common form of electronic signature is through the applying of unique combinations of user-ID and password. It is recommended that passwords be changed at established intervals.[4] Biometric signatures can also be used. When considering the application of signatures to electronic records it is worth considering the actual nature and GCP criticality of the action. In many cases, a signature is being applied to indicate the completion of an event or the attainment of a milestone. In many less frequent cases is the signature actually being applied to indicate compliance or achievement of a GCP requirement. The distinction here is to be able to distinguish whether the signature is being used for identification or authentication purposes. This distinction is clarified by these definitions, provided by Julian Ashbourn:[14]

> **Authentication** refers to the verification of a claimed identity. In other words, the user wishes to log on to a network or service and claims to be a certain person.
>
> **Identification** seeks to identify a user from within a population of possible users, according to a characteristic or multiple characteristics, which can be reliably associated with a particular user without an identity being explicitly claimed by the user.

The distinction should be made between identifying signatures that must be incapable of being repudiated in a court of law (e.g., GCP critical) and those that are not critical enough to warrant the nonrepudiation safeguards.

REGULATORY INSPECTION

Inspectors will normally want to identify those computer systems involved with the particular clinical study under investigation. They will be interested in data capture, processing, and retention. User interaction will also be a key topic of interest in regard to how the computer system assists in making decision, collation of study data, and submissions. Examination of validation documentation and methods of testing may be requested for specific functions. Occasionally, demonstrations of specific functionality might be requested. Security access controls and general administration SOPs on the other hand are often discussed. The main focus is likely to be data integrity and computer system operation and maintenance.

As an example, topics covered at a recent GCP inspection by the FDA of a pharmaceutical manufacturer in North America included:

- Study-specific data entry system — validation, supplier audit, change management, and test protocols
- Data entry system security — virus protection, access management, disaster recovery, archive, and retention
- Electronic records/signatures — assessments and follow-up plans
- Laboratory information management systems (LIMS) — data transfer from clinical systems to networked data management applications

A further example involves an FDA investigation related to a specific submission. In this example, the investigators not only made observations related to the data handling systems employed directly by the pharmaceutical company, but also went on to investigate the systems employed by a Contract Research Organization (CRO) employed to collect data related to the study. Their resultant observations made it clear that the FDA expected the sponsor organization to undertake due diligence when it comes to employing such organizations, and failure to exercise such diligence could result in the issuance of a "483" observation. The same investigation also reviewed the procedures employed for the electronic transmission of electronic data between the CRO and the sponsor, resulting in observations related to the failure to adequately safeguard the security and integrity of the data which was subsequently part of the submission.

No matter how well a pharmaceutical manufacturer believes it conducts validation, it will count for nothing unless during an inspection the regulator understands what has been done and can easily find his or her way around supporting documentation. To this extent a key feature in any validation exercise is inspection readiness.

Seven key elements for being inspection-ready are listed below; others can be added appropriate to the way a pharmaceutical manufacturer wishes to manage regulatory inspections:

- Inventory of systems
- System/project overviews
- Validation plans/reports and reviews
- Presentation slides
- Internal briefing papers
- Document map
- Trained personnel

Using terminology that the various regulatory authorities are familiar with will help enormously. Try to avoid use of company-specific jargon. IT staff especially tend to freely use company-specific acronyms and terminology. Time should be taken to explain topics during an inspection and to prepare IT staff on what to expect during an inspection. They are typically not familiar with regulatory inspections, but with more and more clinical systems using databases and client/server technology, there is a much higher likelihood that IT staff will be required to support inspections. Further details can be found in Chapter 16.

REFERENCES

1. FDA (1999), *Computerized Systems Used in Clinical Trials*, Guidance for Industry, Food and Drug Administration, Rockville, MD.
2. EU (1991), Good Clinical Practice for Trials of Medicinal Products.
3. ICH (1996), *Guideline for Good Clinical Practice*, ICH Harmonised Tripartite Guideline, International Conference on Harmonisation of Technical Requirements for Registration of Pharmaceuticals for Human Use.
4. U.K. Department of Health (1995), The Application of GLP to Computer Systems, The Principles of Good Laboratory Practice, United Kingdom Compliance Program, London.
5. ACDM/PSI (1998), "Computer Systems Validation in Clinical Research: A Practical Guide," Version 1.1, December.
6. OECD (1995), Principles of Good Laboratory Practice to Computerized Systems, Organisation for Economic Co-operation and Development, Paris.
7. FDA (2003), Part 11 *Electronic Records, Electronic Signatures — Scope and Application*, Guidance for Industry (www.fda.gov).
8. DIA (1988), Computerized Data Systems for Nonclinical Safety Assessment: Current Concepts and Quality Assurance, Red Apple Report, Drug Information Association, Maple Green, September.
9. BARQA (1997), *Regulatory Compliance and Computer Systems*, Conference Proceedings, January 7–8, Cambridge, U.K.
10. U.S. Food and Drug Administration (2002), *General Principles of Software Validation; Final Guidance for Industry and FDA Staff.*
11. Gallup, D., Beauchemin, K., Gillis, M., Altopiedi, D., and Manor, J. (2003), Selecting a Training Documentation/Record-Keeping System, *PDA Journal of Pharmaceutical Science and Technology*, 57(1), January/February.
12. FDA (1997), *Electronic Signatures and Electronic Records*, Code of Federal Regulation Title 21: Part 11, Food and Drug Administration, Rockville, MD.
13. Directive 1999/93/EC of the European Parliament and the Council of December 13, 1999, on a Community Framework for Electronic Signatures.
14. Ashbourn, J. (2000), Biometric Definitions (www.ntlworld.com/avanti/authentication.htm).

23 Case Study 5: Control Instrumentation

Tony de Claire, Mi Services Group
Peter Coady, P.J. Coady & Associates

CONTENTS

Instrumentation is the critical link between the manufacturing process and the control system. Instruments are the eyes (i.e., transmitters, sensors) and limbs (i.e., actuators, positioners) of a process control system and enable it to perform the actions that were once performed by operators and laboratory technicians. If an instrument should malfunction, data integrity and the predefined control actions will be affected. Indeed unsatisfactory instrumentation can cause significant operational problems. It is essential that an instrument is carefully chosen to be fit for purpose (i.e., correct type, size, materials, accuracy, repeatability, reliability, documentation, etc.) to enable confidence to be gained in its ability to perform its intended function.[2,3]

This case study embraces the design and validation of both standard and intelligent instrument applications, and briefly discusses special instrument systems (shutdown systems and analyzer packages). The number of instruments that can be involved with the dynamic operation of a pharmaceutical manufacturing process can be large (2000 to 3000 items). In most cases, they are remotely installed from the control room environment and operate unsupervised except during routine maintenance work.

INSTRUMENT APPLICATION DESIGN

The design of instrument applications is an interactive process centered around an instrument schedule and process data. The instrument application design process is shown diagrammatically in Appendix 23B and its chronological position, in relation to an overall project development life cycle, is presented in Appendix 23A.

The instrument schedule is generated from an approved set of Process and Instrument Diagrams (P&IDs) and is used to identify all instruments associated with a manufacturing process. The instruments in the schedule are grouped in tag number (loop) order along with an identification of all associated documentation on which they appear.

Process data are provided by the end user and include critical process data that are generated from, or with reference to, the end user's pharmaceutical product manufacturing specifications.

A HAZOP study (sometimes known as a Hazard Study or Hazard Analysis) is normally carried out on the P&IDs during the design phase to determine where potentially hazardous conditions could occur during the operation of the process and the circumstances that lead to them. The results of the HAZOP study are used to generate safe working practices and the selection of suitable safety devices including rupture disks, safety relief valves, dedicated safety shutdown systems (solid state or PLC- [Programmable Logic Controller] based systems), and hard-wired (relay-based) trip systems.

The types of documentation produced during the instrument application design process are listed below, and the content of each document is described in more detail in Appendix 23E.

- Instrument schedule
- Instrument specification/data sheets
- Cable block diagrams
- Cable schedule
- Pneumatic tubing schedules
- Termination drawings
- Miscellaneous label schedules
- Field panel specification and drawings
- Field junction box drawings
- Electrical hookup (loop or wiring diagrams) drawings
- Pneumatic hookup drawings
- Process hookup drawings
- Instrument layout (location) and cable/tubing routing drawings
- Earthing schedules and drawings
- Miscellaneous drawings (control room/instrument room layouts)
- Instrument installation specification
- Package plant instrument specification
- Electrical and instrumentation interface panel
- Any special instrument specifications and wiring diagrams

Due to the interactions between the various types of instrument design documentation, and the sharing of input information, many of the documents can be produced in parallel. Each document should contain the required content (see Appendix 23E), presented precisely and completely, to enable that part of the plant to be constructed and maintained in a proper and auditable manner.

SUPPLIER SELECTION AND AUDIT

Instrument suppliers are generally selected based on company or site "standards." Where there is no stated preference or knowledge of a particular instrument, selection can be made using a technical evaluation and a tender process. Companies that supply special instrument systems (e.g., shutdown systems, analyzers), companies that provide design/validation consultancy, and subcontractors (engineering design contractors, site installation contractors, panel manufacturers) should be prequalified to determine their suitability from both an engineering and a commercial perspective to receive the tender enquiry documents. Specialist instrument suppliers and consultants would generally be subject to a Supplier Audit prior to any order being placed. Similarly, suppliers wishing to be included on the preferred vendor list for the site will also be subject to a Supplier Audit.

Audits should be conducted by suitably trained/qualified personnel against the applicable ISO 9000[4] series standard, with special reference being made to software quality guidelines (e.g., ISO 9000-3,[5] TickIT[6]) for instrument systems involving software. The software audit may cover the development of both application software and core (operating system level) software, depending on the type of system to be supplied. Follow-up audits should be considered.

PREDELIVERY TESTING AND CALIBRATION

A detailed account of the calibration life-cycle processes can be found in the *GAMP Good Practice Guide on Calibration Management*.[11]

Calibration and Test Equipment Requirements

The manufacturer should possess test equipment to enable all manufacturing tests and inspections to be performed. All test equipment used by the manufacturer must have a standard of accuracy better than the stated accuracy for the instrument(s) to be tested. All applicable test equipment must have a valid calibration certificate issued by a calibration laboratory that is certified to either a national or international standard (e.g., NAMAS [National Measurement Accreditation Service]) for calibrating the specific types of instruments concerned.

Factory Testing

The manufacturer should have written test procedures and should test all equipment supplied against these procedures prior to delivery. The manufacturer's testing should comprise the physical checking and operational and functional testing of instrumentation (e.g., valves, transmitters) and equipment (e.g., panels, analyzers) to be supplied. All tests must be fully documented, the results recorded, and the appropriate test sheets signed off by an authorized person.

Factory Calibration

Factory calibration should be carried out against the instrument specification/data sheets supplied. The calibration activities should address the following areas, as applicable:

- Process operating ranges
- Required accuracy
- Repeatability
- Hysteresis effects
- Switch set points
- Condition of switching (e.g., on a rising or falling measured variable)
- Switch action (e.g., open or close on fault condition)

In most cases, the manufacturer’s own calibration procedures should be acceptable as they form part of the manufacturer’s own performance guarantees and/or quality certification. If special calibration is required, then companies with custom-built test facilities must be used (e.g., a magnetic flowmeter that requires a specific certification will require calibration in a flow rig that has been certified by an approved certifying authority).

DESIGN REVIEW

Design review (also called Design Qualification: DQ) is the name given to the technical and quality audit of the instrument application design engineering, construction documentation package, vendor documentation, factory inspection report forms, and calibration data. The purpose of the design review is to verify through defined procedures and support documentation that the individual items of instrument application design have been designed and approved so that they meet the needs of the customer and the contractor’s project and quality plan. The findings of the design review and the documentation inspected should be formally recorded, and a design review report should be produced.

MANUFACTURING DOCUMENTATION REQUIREMENTS

The manufacturer should provide copies of the following documents prior to the delivery of the equipment. It should be specified that documents are written in the local national language wherever possible.

Factory Calibration Certificates

The calibration certificate should include the following information:

- Serial number
- Instrument specification/data sheet number
- Tag number
- Model number
- Certificate numbers of the test equipment used for the calibration
- Validity period of the calibration certificate
- Calibration data
- Limits of uncertainty (for critical instrumentation)

Further information on calibration certificates can be found in the *GAMP Good Practice Guide on Calibration Management*.[11]

Equipment Test Records

Copies of the test records for panels and associated equipment should be provided. Equipment test records can include the following:

- Copies of the signed and approved test record sheets
- The manufacturer’s own quality system compliance
- Electro-magnetic compatibility Declaration of Conformity certificates (self-certification) for equipment containing European Union (EU) CE-marked equipment (e.g., panels)

Hazardous Area Approval Certificates

Copies of hazardous area approval certificates for all applicable equipment should be provided. The approval certificates are issued by national or international approvals bodies, including:

- British Approvals Service for Electrical Equipment in Flammable Atmospheres (BASEEFA) and Physikalische-Technische Bundesanstalt (PTB), Germany, which, along with other European approvals bodies, provides certification to the European Committee for Electrotechnical Standardization (CENELEC) standards
- Factory Mutual Research Corporation (FM) and Underwriters Laboratories Inc. (UL), United States
- Canadian Standards Association (CSA)
- Standards Association of Australia (SAA)

Material Certificates

Material certificates are normally only required where the materials of construction were specified on the instrument specification/data sheets to comply with process or environmental requirements (e.g., valve bodies and trims).

Construction Documentation Package

The construction documentation package comprises the instrument installation specification, drawings, and documentation listed in Appendix 23E, "Instrument Installation Specification." During this phase and thereafter, it is important that the drawing register for the manufacturing process is maintained on-site and that only the latest revisions of drawings are used.

The instrumentation construction documentation package may also reference supplementary information contained in the following documentation:

- Process design documentation (e.g., P&IDs, ELDs [Engineering Line Diagrams], ULDs [Utility Line Diagrams])
- Piping design documentation (e.g., piping isometric drawings)
- Process unit design documentation (e.g., vessel connections)
- Mechanical design documentation (e.g., package plant vendor installation drawings)
- Computerized control system documentation (e.g., termination and interconnecting cabling drawings)

EQUIPMENT DELIVERY, INSPECTION, PROTECTION, AND STORAGE

On arrival at site, instrumentation and equipment should be checked against the delivery note, checked for damage, and then either preinstallation tested and installed or put into a suitable store until required. Any discrepancy or damage should be recorded and reported to the supplier through the contractual channels established for the project. Rejected items should be stored and controlled separately from accepted items.

Instrumentation and equipment (e.g., panels, junction boxes) that cannot be installed on delivery must be housed in a properly constructed and conditioned store and protected from dust and moisture. Completion of control rooms should be programmed to permit the installation of panels immediately on receipt to minimise handling. If the control room heating system is not in operation, temporary heaters must be installed to ensure that the panels and instrumentation are kept within acceptable temperature and humidity limits.

Throughout the construction period, instruments that are not provided with housings must be adequately protected by covering with heavy duty plastic bags of an approved type or by applying more robust protection where necessary. The protection of instruments and the provision of covers is the responsibility of the installation contractor.

SITE PREQUALIFICATION

GENERAL

All instruments should, wherever possible, be subject to a preinstallation test; this test should commence as soon as practicable after the receipt of the instrument on-site. The object of preinstallation testing is to ensure that each instrument has been supplied in accordance with its specification, is functionally correct, and is in working order. Where the preinstallation test is not specified or where circumstances prohibit carrying out the prescribed test, the installation contractor must propose a suitable test method for approval by the customer.

The tests should be performed as described below and with due consideration given to the manufacturer's recommended test methods. Adjustments must be carried out in accordance with the manufacturer's instructions. Any deviation from this must be approved by the customer and supplier prior to testing. All tests must be fully documented, the results recorded, and the appropriate test sheets signed off by an authorized person.

Instrument testing should preferably be carried out in a calibration workshop. However, instruments that form part of an integrated system or control panel may be tested in the control room or instrument room after installation, using portable test gear and/or simulation equipment. All instruments that require calibration must be calibrated in both the upscale and downscale directions and, if necessary, adjusted until their accuracies are within the limits stated by the manufacturer. On completion of the tests, the instrument must be suitably cleaned and protected in accordance with the manufacturer's recommendations.

A detailed account of the calibration life-cycle processes can be found in the *GAMP Good Practice Guide on Calibration Management.*[11]

PREPARATION FOR SITE TESTING

The following checks should be carried out before preinstallation testing commences:

- The instrument must be checked for damage (e.g., damage to doors, linkages). Any such damage must be rectified and approved before any tests are attempted.
- The data plate on the instrument must be checked for agreement with the information contained in the appropriate instrument specification/data sheet.
- A suitable means must be provided for simulating the required process conditions, and test gauges or meters must be made available with a sufficient degree of accuracy for the tests to be performed.
- The instrument to be tested should be mounted in the correct plane on a rigid and vibration-free stand or structure.
- The manufacturer's instruction book must be made available.
- All tests must simulate as closely as possible design process conditions.
- Tests must not be carried out on electronic instruments until an adequate warm-up period has elapsed. Wherever possible, instruments must be energized for at least 24 h prior to testing.
- The instrument to be tested must be properly prepared prior to testing by the removal of any shipping stops and the installation of any miscellaneous components (e.g., charts, mercury, oil).

TEST STATUS INDICATION

On completion of each site test, the stage reached in the testing procedure should be clearly indicated. A typical method would be affixing to each instrument or installation a colored label conforming to the following code:

- Blue: Preinstallation tested
- Yellow: Pressure tested
- Green: Cables tested
- Red: Precommissioned
- White: Test failed (a written message may be added giving the reason for failure)

This identification must be shown on all components in the loop, thereby making all personnel aware of the current status of any instrument and its installation.

Connecting an Energy Supply

The following procedure is common to all instruments that require an energy supply source and that generate a signal output.

Pneumatic Instruments

Connect the air supply and adjust the air supply regulator to the correct setting (e.g., 1.4 bar for a standard transmitter with an operating range of 0.2 to 1 bar, 20 psig for an operating range of 3 to 15 psig). Connect the output to a suitable test gauge via a capacity chamber (approximately 0.5-l capacity).

Electronic Instruments

Connect a suitable power supply. Connect the output to a suitable test meter, preferably a digital voltmeter, installed across a current dropping resistor.

Connecting a Signal Generator for Process Simulation

Connect a signal-generating source with an accurate indicator to the sensing device, together with the means of isolating and regulating its output. The type of signal generator required will depend on the type of signal to be simulated and should conform to recognized instrument calibration standards and the supplier's instructions. When hazardous fluids or gases are involved (e.g., oxygen or ammonium nitrate), suitable safety precautions must be observed and the test method must be agreed on with the customer.

Site Calibration

The calibration procedures adopted on-site must be agreed on with the customer and conform to recognized industry instrument calibration standards and the supplier's instructions. These procedures must be applied to all in-line instrumentation, loop instrumentation, local controllers, analyzers, and so on. Where the control and monitoring instrumentation is integrated with a computerized control system and where factory tests have been carried out, the installation calibration procedure should be agreed on with the customer.

Calibration checks are usually carried out on analyzers by injecting known samples into the sample conditioning systems. This must be determined for each type of analyzer by reference to the manufacturer's handbook or by consultation with the instrument vendor, and it must be agreed on with the customer. Complex analyzer systems usually require specialist personnel from the analyzer manufacturer to assist in precalibration and commissioning and are generally outside the scope of the installation contractor's responsibility and experience.

The calibration results should be recorded (a sample instrument preinstallation calibration sheet has been provided for reference purposes in Appendix 23C) and included with the site test records.

Further information on calibration records can be found in the *GAMP Good Practice Guide on Calibration Management*.[11]

Instrument Mounting and Accessibility

Each instrument to be installed must be inspected to check that its data plate agrees with the specification and that it has been preinstallation tested, if applicable, as described in the preceding sections. The instrument should then be installed in its intended location on brackets, a subpanel, a mounting post, or a pedestal, ensuring that it is leveled, plumbed, and firmly secured. The installation must follow good instrument installation practice and the supplier's instructions, and the instrument should be protected from damage until it is put into service.

Indicating instruments and instruments requiring adjustments should be accessible for observation and servicing from the floor level, walkways, permanent ladders, or platforms. Where possible, actuated valves should be accessible from the floor or permanent platform level.

Instrument Piping and Tubing

The installation and pressure testing of air supply piping, transmission/signal tubing, and process impulse piping must conform to, and be checked against, the instrument installation specification, the construction documentation package drawings, and recognized industry instrument standards.

Cable Installation and Testing

The installation and testing of signal transmission cabling and power cabling must conform to, and be checked against, the instrument installation specification, the construction documentation package drawings, and recognized industry standards. Immediately after cables have been laid and before connection, all electric and electronic instrument wiring must be checked for polarity, continuity, and insulation resistance between the conductors and between the conductors and earth. These tests should be carried out before final loop tests and should comply with industry standards and statutory regulations.

Coaxial cables used for data highways must be tested using sine-wave reflective testing techniques. Circuits involving intrinsically safe (IS) instrumentation must be tested (e.g., loop impedance, inductance, L/R [Inductance/Resistance] ratio) in accordance with the manufacturer's instructions and approved by the customer.

Wiring that connects field instrumentation to a computerized control system should be isolated from the computerized control system (e.g., at the control room/marshaling cabinets) during cable testing in order to safeguard against damage due to incorrect connections. Isolation from the computerized control system may be provided by the use of isolating (e.g., knife edge) field terminals and/or by disconnecting the computerized control system input/output (I/O) wiring termination blocks.

After the tests have been completed, the cables should be identified with a colored label which clearly indicates its test status (see above). Wiring should be reconnected on completion of cable testing and recorded as such.

Loop Testing

The object of loop testing is to ensure that all instrumentation components in a loop are in full operational order when connected together and are in a state ready for process commissioning and validation (Operational Qualification [OQ] and Performance Qualification [PQ]). Loop testing also encompasses the integration of instrumentation with any associated computerized control systems.

The procedure to be adopted in carrying out these tests is detailed below but, in general, the completed loop should be tested as one system and, where necessary, adjustments should be made to ensure that the loop is fully operational as a system and is correctly calibrated. Associated alarms and trips must be checked during loop testing.

The loop test results should be recorded (a sample instrument loop check sheet has been provided for reference purposes in Appendix 23D) and included with the site test records. Checks for mechanical/electrical completeness are recorded using the upper section of the sheet and the dynamic loop test results are recorded on the lower section. The test results sheets will provide the documentary evidence essential for Installation Qualification (IQ). Representatives from the installation contractor and/or the customer will witness the final loop tests and countersign the test sheets. Any tests not witnessed must be accompanied by written confirmation from the customer that witnessing has been waived.

Loop testing of remote control loops is a two-person exercise, with one person located in the field and the other in the control room or instrument room. Each person must be provided with an adequate means of remote communication (e.g., field telephones or two-way radios) as approved by the customer.

Loop testing of instrumentation and any associated computerized control system should encompass the interfaces with electrical equipment (via the electrical and instrumentation interface panel) and its related operation. Loop testing must not be carried out on electronic equipment until an adequate warm-up period has elapsed. Where possible, equipment should be energized for at least 24 h prior to testing.

On completion of loop testing, it is recommended that all control devices/functions are left set with the correct control action and with a 100% proportional band setting. Derivative and integral functions must be set at their minimum time values.

Loop Testing Procedure

The following test procedure should be carried out in order to test the correct operation of field instrumentation and equipment installed in a control loop, and to provide the necessary documentary evidence (test records) to satisfy the requirements of the IQ protocols:

- Inspect the loop and set air/electrical supplies where appropriate. Check in particular that control valve air supply pressures are set in accordance with the instrument specification/data sheet.
- For electronic loops, check polarities, measure the loop impedance, and make the necessary compensating adjustments. The compensating adjustments on smart instruments can be made using either a handheld terminal or directly at the instrument. Smart instruments, if supported, can also be adjusted using an instrument configuration page on a computerized control system.
- Transmitter output signals equivalent to 0, 50, and 100% of the instrument range should be generated, either manually (e.g., hot oil bath, dead weight tester) or by applying appropriate signals at the field terminals in order to check the response of all other instruments and control valves in the loop. Instrument zero settings and calibration checks/adjustments should be made as necessary.
- Switch the loop controller to manual operation and, by applying the appropriate output signals, ensure that the control valve(s) stroke correctly. Valve positioner gauges should also be checked during this stage.
- Apply an input signal to the loop controller equivalent to 50% of the instrument range and adjust the output of the manual pneumatic regulator to 50%. Adjust the loop controller setpoint to 50% and, by switching the auto/manual transfer switch, check for “bumpless” transfer. Using the manufacturer’s instructions, adjust where necessary until a satisfactory bumpless transfer is achieved.

- Check all alarm and trip actions by varying the loop controller input signals and adjust as necessary.

After the tests have been completed, the loop controller must be switched to manual operation and then identified with a colored label that clearly indicates its test status (see above).

Testing Computerized Control System Loops

Before this level of testing commences, the instrument loop must be prepared in accordance with the loop testing procedure. All test methods must be agreed with the customer.

The operation of instrumentation must be checked from the field to the control room graphics display unit or local controller I/O registers as applicable, depending on the type of system installed, and vice versa. The operation of control instrumentation (e.g., control valves, actuated on/off valves) should be checked by energizing each control system field output from either the control room display or local controller as applicable, and observing and recording the results.

The operation of monitoring instrumentation (e.g., transmitters, switches) should be checked by either injecting a suitable signal at the field instrument terminals or by installing the instrument in a comparator (e.g., a hot oil bath, dead weight tester). The result received by the control system on the control room display or local controller I/O register, as applicable, should be recorded. Any problems should be reported to the company/companies responsible.

The results of all tests must be recorded on loop test sheets. A sample copy of a loop test sheet is provided in Appendix 23D. As each test is completed, the tested item must be identified with a colored label that clearly indicates its test status (see above).

Testing Safety Interlocks

Safety interlocking and shutdown systems require detailed test procedures based around the design documentation (e.g., cause-and-effect charts, binary logic diagrams) that must be formulated and agreed on in advance.

Site Modifications and As-Built Drawings

Any modifications, exceptions, or additions to the construction documentation must be confirmed in writing by the customer before such work is commenced by the installation contractor. This work must be properly organized, with clear definitions of responsibilities for checking and approval, and must be undertaken following a strict change control procedure that identifies all documentation affected by the change.

The installation contractor should assist in the provision of as-built drawings that may be completed by others (e.g., the engineering design contractor) at the end of the contract. To assist in this exercise, the installation contractor must keep a set of printouts of all documentation that must be marked up as and when changes are agreed.

QUALIFICATION

Installation Qualification

On completion of installation, all documentation associated with the installation, calibration, and testing of the field instrumentation, along with any associated computerized control system documentation, should be collated by the project manager/engineer ready for IQ. Most of this documentation will be in the pre-“as-built” condition at this stage of the project and will, therefore, contain site (red line) markups.

The documentation should typically include the following:

- Construction documentation package (see Appendix 23E, “Instrument Installation Specification”)
- Instrumentation and loop test records and reports
- Instrument calibration certificates[11]
- Change control notices and supporting forms
- Manufacturer’s operation and maintenance manuals

IQ should be carried out using a written protocol[7] that is completed during the inspection of the installation and its documentation. The protocol should typically describe how the documents inspected will be marked to show their IQ acceptance status, as well as the acceptance criteria for the items inspected, and it should have space to record the reference numbers of the documents seen and where they can be located.

Operational Qualification and Performance Qualification

OQ verifies that the control and monitoring instrumentation, as integrated with the process equipment and any associated computerized control system, meets the operational and functional requirements defined in the instrument application design documentation and/or computerized control system User Requirements Specification (URS).[7] PQ verifies that the control and monitoring instrumentation, as integrated with the process equipment and any associated computerized control system, meets the operational and functional requirements defined in the instrument application design documentation and/or computerized control system URS, and produces pharmaceutical product consistently to specification.[7]

OQ and PQ should be carried out using written protocols that are completed during the functional testing of the manufacturing process. The protocols should typically describe how the tests will be carried out, as well as the acceptance criteria for the items tested, and they should have space to record the reference numbers of the documents seen and where they can be located. The principal reference document for the OQ testing of the instruments will be the instrument site commissioning test procedures and results. Some points to consider during OQ and PQ are described below.

Recalibration

There may be a large time gap between the IQ, OQ, and PQ phases for instrumentation associated with particular manufacturing processes due to the site construction program (e.g., unavailability of utilities, panels). As a result, some control and monitoring instrumentation may need to be recalibrated prior to commencing OQ, and possibly again prior to PQ, depending on calibration frequencies. It would be advisable to recalibrate critical instrumentation anyway to ensure its status is known prior to OQ and PQ.

Recalibration must be carried out to agreed upon standard procedures using calibration test equipment that is traceable back to national standards. All calibration tests must be fully documented, the results recorded, and the sheets signed off by an authorized person. Calibrated instruments must be provided with a full calibration certificate that details the test results and their limits of uncertainty. A detailed account of the calibration life-cycle processes can be found in the *GAMP Good Practice Guide on Calibration Management*.[11]

Testing

The customer should be kept informed of all site tests and when they are to occur by the supplier/contractor so that arrangements can be made for the end user to attend for witnessing purposes.

All testing must be fully documented using test record sheets, and witnessed and signed off by the customer and the supplier. Copies of all test result sheets/test records and reports should be reviewed and approved by the supplier/contractor and suitably qualified customer representatives. Any document revisions necessary during OQ and PQ must be progressed through the site document management system and implemented under a formal change control procedure.

Handover

The completion of OQ marks the end of the site construction (installation and commissioning) phase for the project, and the manufacturing process is formally handed over to the customer by the signing of a handover certificate prepared by the engineering design contractor. The handover certificate should be accompanied by an "as-built" issue of the construction documentation package bound in a series of Operation and Maintenance (O&M) manuals, along with all test results, calibration certificates, manufacturer's reference manuals etc., to enable the manufacturing process to be properly maintained for the rest of its operational life. Some of the information contained in the O&M manuals (e.g., calibration certificates) may be removed and placed in a centralized maintenance system for ease of control.

Reporting

Testing occurs at many levels in a project, as described in the preceding sections of this case study, starting with design phase verification activities (design reviews) and ending with the testing associated with IQ, OQ, and PQ on-site. All reviews and tests must be fully documented using appropriate methods (e.g., minutes of design review meetings, test record sheets), and must have the required checking and approval signatures. These documents, along with change control documentation, calibration certificates,[11] etc., comprise the formal records generated to provide the necessary evidence to support validation.

PERIODIC REVIEW

Periodic reviews of the manufacturing process, including the control and monitoring instrumentation and any associated computerized control system, must take place from the time it is handed over to a site until it is replaced and/or decommissioned,[7] in order to verify that it continues to be capable of producing quality product to specification. The purpose of a periodic review, with regard to control and monitoring instrumentation, is to verify that it has been maintained in a validatable condition.

Typical activities associated with the periodic review phase for instrumentation include establishing a recalibration program[8,11] and conducting routine maintenance activities as part of a planned preventive maintenance scheme. All maintenance activities must be carried out under a formal change control procedure and any associated testing must be fully documented using test record sheets.

NOTES ON SPECIAL INSTRUMENTS AND TECHNOLOGIES

A number of special instrumentation systems may be required for a project. This section briefly describes some typical systems used in the industry.

Emergency Shutdown Systems

Emergency shutdown (ESD) safety systems are relatively new to the pharmaceutical industry but have been used for many years in the petrochemical and other industries. They provide an independent, reliable (high integrity) method for protecting plant and personnel from situations that could lead to a dangerous occurrence (e.g., runaway reactions, handling of dangerous substances). A typical application would be on a solvent recovery plant or tank farm for the isolation of solvent feeds to the process. ESD systems are often designed by the instrument discipline as part of trip system design and can be either solid-state or PLC based.

Solid-state systems comprise a number of cards, each of which contain a number of independent logic channels (i.e., AND, OR, NOT function blocks) and special functions (e.g., timer blocks) that are hard-wired to provide the shutdown logic required. The cards are monitored and controlled by dual redundant CPU cards and are interfaced to the field by high integrity/availability relays to prevent problems with contact welding and/or malfunctioning due to lack of use. The systems (cards, CPUs, etc.) are subject to inspection and testing by an independent certification body (e.g., the German TÜV [Technischer Überwachungs-Verein] in Europe) and issued with an approval certificate.

PLC-based systems are similar to solid-state systems. They use the same field interface cards and have the same level of certification but the logic functions are performed by software. They are also easier to interface to other systems (via serial communications) in order to notify the system of a failure. The PLC used is subject to rigorous source code (machine level) inspection and testing by an independent certification body (e.g., TÜV in Europe) to examine the safety integrity of the code, including all possible failure paths. The PLC system (application software and hardware) should be developed and tested using a formal, life-cycle methodology.[7,9]

The shutdown logic required is usually determined during a HAZOP and represented on either cause and effect charts or on binary logic diagrams. Cause-and-effect charts have a spreadsheet (matrix) type of presentation and show for each identified failure condition (e.g., high temperature alarm) the safety status required for all affected equipment. Binary logic diagrams perform a similar function but show the physical connections between each logic block in diagrammatic form. More detailed versions of the binary logic diagrams showing card numbers, addresses, etc. should be provided by the system supplier as part of the documentation package.

Analyzer Packages

Analyzer systems are now becoming more widely used in the pharmaceutical industry at all levels of product development (primary, secondary, and R&D). Some typical systems include mass spectrometers, gas chromatographs, and near infrared (NIR) systems.

Each analyzer should have a URS and should be supplied with a detailed design specification, installation and maintenance documentation, and operator instruction manuals. Some special considerations include the following:

- *The sample loop:* The sample (fast) loop for off-line analyzers must bring a representative sample to the analyzer in the shortest acceptable time so that, accounting for the analysis time itself and the output response time of any corrective systems, the system response time is achieved.
- *Sample conditioning:* The need for sample conditioning (heating, cooling, drying) must be addressed and consideration given to the effect this could have on the system response time.
- *Analyzer results:* The format of the data produced by the analyzer must be specified.
- *Communications:* The ability of the analyzer to send data or signals to other systems must be specified. Raw analysis data may need to be transmitted to another system for further analysis and/or presentation. Further, the analyzer may also monitor for the occurrence of critical situations (e.g., the start of an exothermic reaction, high solvent content, high particulate count), in which case there may be a requirement to communicate (via communications link or hard-wiring) to a shutdown system or circuit.

- *Validation issues:* The analysis data usually form part of the manufacturing batch records. Consideration must be given to a Supplier Audit, and this will include a formal software audit if the analyzer is controlled by a computer system. The design and testing of the analyzer computer system (application software and hardware) will need to follow a formal, life-cycle methodology.[7,9]

Intelligent Instruments

Intelligent instruments are widely used in the industry and include smart transmitters, loop controllers, chart recorders, machine monitoring systems, and fume cupboard controllers. These instruments contain embedded software in the form of nonuser programmable firmware that is configured either by "filling in the blanks" or by entering high level statements. These devices are considered to be "black boxes" in validation terms and are classified as "Category 2 Software" in the GAMP Guide.[9]

Although these devices are not subject to the normal rigor of application software validation on a project, the manufacturers of these devices are still expected to have formal documented methods and records in place for developing and testing the software, for controlling and reporting changes due to bug fixes and upgrades, and for the configuration management of both the software and firmware products. Information on these and other areas may be obtained for the project validation records either by a manufacturer audit or by signed statements from the manufacturer in response to a questionnaire.

Intelligent devices are now becoming associated with the evolving Fieldbus standard[10] for data acquisition, device control, and configuration. Fieldbus™ is a multidrop network standard that allows both digital (serial) and analog (4–20 mA) signals to share the same cable without interference. The main points to consider for validation are data integrity and security within the network (i.e., evidence to ensure data are not corrupted when being transmitted within the network between devices or to a computerized control system). This evidence should include formal documentation and records of the data transfer/communication standard used, and error and diagnostic checks.

In addition to the more commonplace monitoring instruments and measurement devices, technology continues to present the industry with a range of "super" instruments. Equipment such as robotics, spectrophotometers, bio-instruments, vision/imaging systems, color recognition, particle monitoring, and other sophisticated analyzers could be considered as in-line instruments with embedded software. With these devices, more complex calibration and qualification issues will have to be addressed in addition to the manufacturer's design and testing methods described above.

RETROSPECTIVE VALIDATION

This case study has concentrated on the design of new instrument applications subject to prospective validation. The validation of existing instrument applications will necessitate a retrospective validation approach. The main difference between the two approaches is in the use of historical design information to prove that the current installation is properly documented and maintained.

Retrospective validation invariably involves the verification of any existing documentation, and the generation of missing information/documentation. As the plant/process is existing, some of the documentation involved with the original site installation can be ignored, provided it does not contain information needed for operation or maintenance. Maintenance and effective records are the main considerations when deciding on whether a particular document is or is not required.

Considering the construction documentation package listed in Appendix 23E, "Instrument Installation Specification," the minimum level of documentation that should be in place for retrospective validation is as follows:

- Instrument schedule
- Instrument specification/data sheets
- Cable schedules
- Pneumatic tubing schedules
- Field panel drawings (e.g., layout, wiring, termination, and piping drawings)
- Electrical hookup (loop or wiring diagrams) drawings
- Pneumatic hookup drawings
- Earthing drawings/details
- Electrical and instrumentation interface panel wiring diagrams
- Special instrument specifications and wiring diagrams
- Shutdown/safety system logic diagrams
- Junction box layouts

The following certification, operation, and maintenance documentation will be required:

- Instrument calibration certificates[11]
- Instrument hazardous area classification certificates
- Materials certificates or other method of material verification
- EMC Declaration of Conformity certificates (self-certification) for equipment containing EEC CE-marked equipment (e.g., panels)
- Manufacturer's O&M instructions

The following supplementary documentation provided by other disciplines will also be required:

- Process design documentation (e.g., P&IDs, ELDs, ULDs)
- Mechanical design documentation (e.g., package plant vendor drawings)
- Computerized control system documentation (e.g., termination and wiring drawings)

CONCLUSION

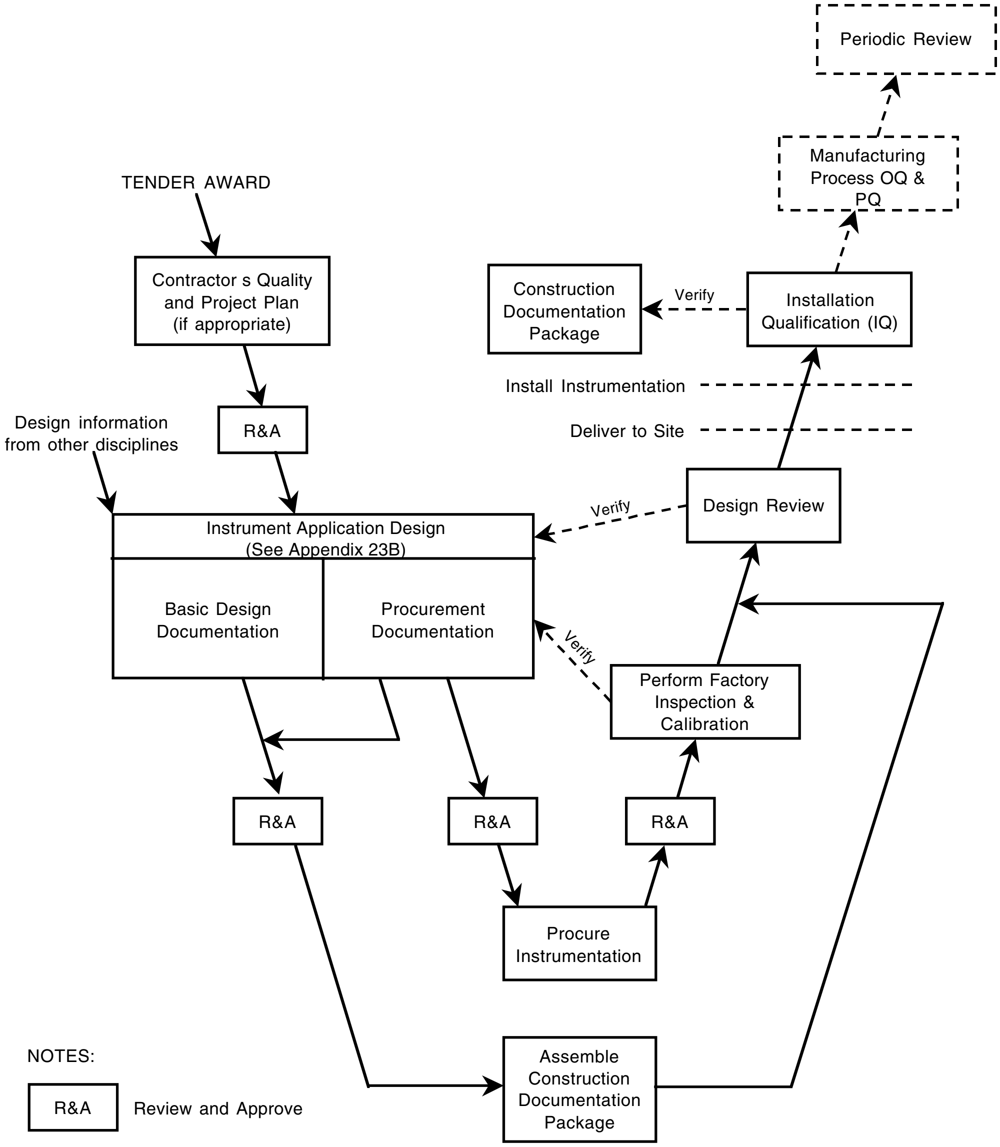
Instrument application design is one of the most extensive activities, in terms of the number of types of instrument and the associated paperwork, and requires a logical, carefully controlled approach. A change to an instrument tag number or the addition/deletion of an instrument can affect up to ten different documents. However, with good document control, it is possible to produce a system of documentation that will allow instrument applications, both simple and complex, to be validated and easily maintained. This case study has described the instrument application design process commonly used in the industry and how this can be used to support validation.

REFERENCES

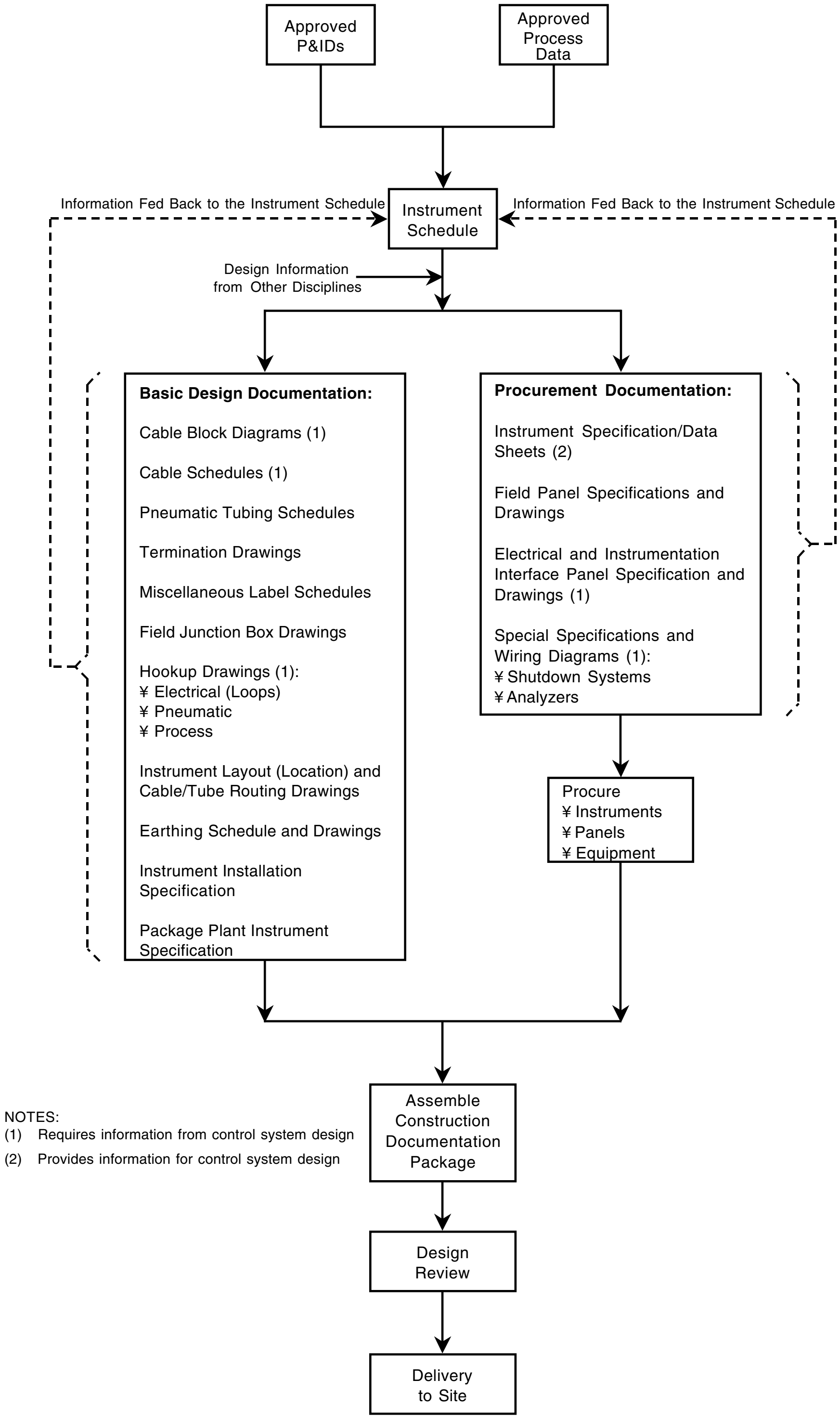
1. U.S. Code of Federal Regulations Title 21, Part 210 (last amended April 2000), *Current Good Manufacturing Practice in Manufacturing, Processing, Packaging, or Holding of Drugs;* Part 211 (last amended April 2000), *Current Good Manufacturing Practice for Finished Pharmaceuticals.*
2. U.S. Code of Federal Regulations, Title 21, 211.68(a).
3. U.S. Code of Federal Regulations, Title 21, 820.72.
4. BS EN ISO9001:2000, Quality Management Systems — Requirements. International Organization for Standardization, Geneva.
5. ISO9000-3:1997, *Quality Management and Quality Assurance Standards* — Part 3: *Guidelines for the Application of ISO9001:1994 to the Development, Supply, and Maintenance of Computer Software.* International Organization for Standardization, Geneva.

6. *The TickIT Guide: Using ISO 9001:2000 for Software Quality Management System Construction, Certification, and Continual Improvement*, Issue 5 (January 2001). DISC TickIT Office, London.
7. Coady, P.J. and de Claire, A.P. (1995), Best Practice Engineering for Validation of Process Control Systems. *Pharmaceutical Engineering* (July/August): 18–30.
8. U.S. Code of Federal Regulations, Title 21, 211.160(b)(4).
9. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (www.ispe.org).
10. ANSI/ISA-50. Fieldbus Standard for Use in Industrial Control Systems. Multiple part standard. ANSI/ISA.
11. ISPE (2000), *GAMP Good Practice Guide: Calibration Management*, GAMP Forum.

APPENDIX 23A
THE INSTRUMENT APPLICATION DEVELOPMENT LIFE CYCLE

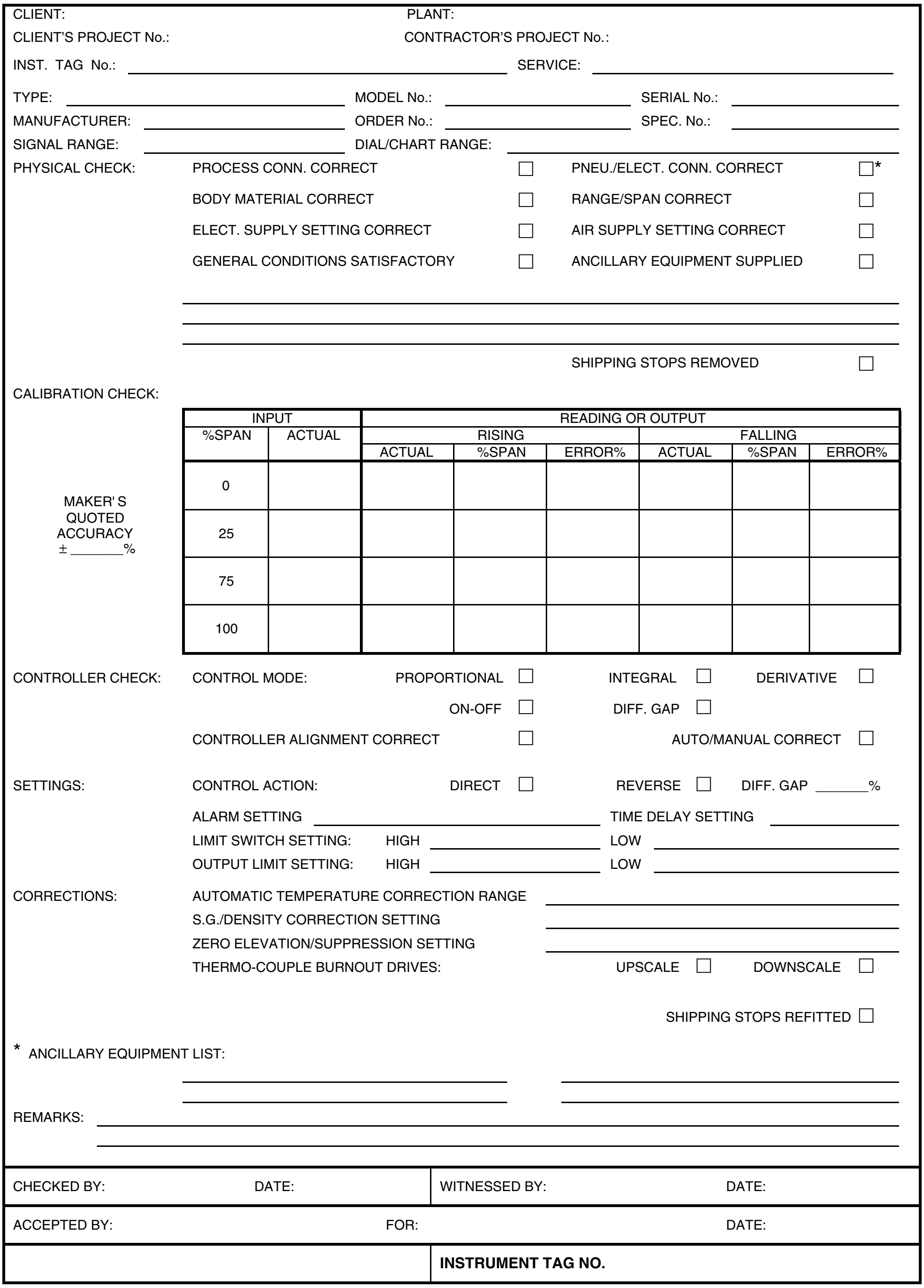

Periodic Review
Manufacturing Process OQ & PQ
TENDER AWARD
Contractor s Quality and Project Plan (if appropriate)
Construction Documentation Package
Verify
Installation Qualification (IQ)
Install Instrumentation
Deliver to Site
R&A
Design information from other disciplines
Design Review
Verify
Instrument Application Design (See Appendix 23B)
Basic Design Documentation
Procurement Documentation
Verify
Perform Factory Inspection & Calibration
R&A
R&A
R&A
Procure Instrumentation
NOTES:
Assemble Construction Documentation Package
R&A
Review and Approve

APPENDIX 23B
OVERVIEW OF THE INSTRUMENT APPLICATION DESIGN PROCESS


Approved P&IDs
Approved Process Data
Information Fed Back to the Instrument Schedule
Instrument Schedule
Information Fed Back to the Instrument Schedule
Design Information from Other Disciplines
Basic Design Documentation:
Cable Block Diagrams (1)
Cable Schedules (1)
Pneumatic Tubing Schedules
Termination Drawings
Miscellaneous Label Schedules
Field Junction Box Drawings
Hookup Drawings (1):
¥ Electrical (Loops)
¥ Pneumatic
¥ Process
Instrument Layout (Location) and Cable/Tube Routing Drawings
Earthing Schedule and Drawings
Instrument Installation Specification
Package Plant Instrument Specification
Procurement Documentation:
Instrument Specification/Data Sheets (2)
Field Panel Specifications and Drawings
Electrical and Instrumentation Interface Panel Specification and Drawings (1)
Special Specifications and Wiring Diagrams (1):
¥ Shutdown Systems
¥ Analyzers
Procure
¥ Instruments
¥ Panels
¥ Equipment
Assemble Construction Documentation Package
Design Review
Delivery to Site

NOTES:
(1) Requires information from control system design
(2) Provides information for control system design

APPENDIX 23C
SAMPLE INSTRUMENT PREINSTALLATION CALIBRATION SHEET



CLIENT: PLANT:

CLIENT'S PROJECT No.: CONTRACTOR'S PROJECT No.:

INST. TAG No.: ____________ SERVICE: ____________

TYPE: ____________ MODEL No.: ____________ SERIAL No.: ____________

MANUFACTURER: ____________ ORDER No.: ____________ SPEC. No.: ____________

SIGNAL RANGE: ____________ DIAL/CHART RANGE: ____________

PHYSICAL CHECK:

- PROCESS CONN. CORRECT □
- BODY MATERIAL CORRECT □
- ELECT. SUPPLY SETTING CORRECT □
- GENERAL CONDITIONS SATISFACTORY □
- PNEU./ELECT. CONN. CORRECT □*
- RANGE/SPAN CORRECT □
- AIR SUPPLY SETTING CORRECT □
- ANCILLARY EQUIPMENT SUPPLIED □

____

____

____

SHIPPING STOPS REMOVED □

CALIBRATION CHECK:

MAKER' S QUOTED ACCURACY ± _______%

INPUT		READING OR OUTPUT					
%SPAN	ACTUAL	RISING			FALLING		
		ACTUAL	%SPAN	ERROR%	ACTUAL	%SPAN	ERROR%
0							
25							
75							
100							

CONTROLLER CHECK: CONTROL MODE: PROPORTIONAL □ INTEGRAL □ DERIVATIVE □

ON-OFF □ DIFF. GAP □

CONTROLLER ALIGNMENT CORRECT □ AUTO/MANUAL CORRECT □

SETTINGS: CONTROL ACTION: DIRECT □ REVERSE □ DIFF. GAP _______%

ALARM SETTING ____________ TIME DELAY SETTING ____________

LIMIT SWITCH SETTING: HIGH ____________ LOW ____________

OUTPUT LIMIT SETTING: HIGH ____________ LOW ____________

CORRECTIONS: AUTOMATIC TEMPERATURE CORRECTION RANGE ____________

S.G./DENSITY CORRECTION SETTING ____________

ZERO ELEVATION/SUPPRESSION SETTING ____________

THERMO-COUPLE BURNOUT DRIVES: UPSCALE □ DOWNSCALE □

SHIPPING STOPS REFITTED □

* ANCILLARY EQUIPMENT LIST:

____________ ____________

____________ ____________

REMARKS: ____________

____

CHECKED BY:	DATE:	WITNESSED BY:	DATE:
ACCEPTED BY:	FOR:		DATE:
		INSTRUMENT TAG NO.	

APPENDIX 23D
SAMPLE INSTRUMENT LOOP CHECK SHEET

CLIENT: PLANT:

CLIENT'S PROJECT No.: CONTRACTOR'S PROJECT No.:

LOOP No.: ______________________ SERVICE: ______________________

LINE OR EQUIPMENT No.: PIPE I.D.:

MECHANICAL/ELECTRICAL CHECKS

MEASURING ELEMENT:		INSTALLATION CORRECT	□	LOCATION CORRECT	□
		ISOLATING VALVES CORRECT	□	MATERIALS CORRECT	□
		TAPPING(S) POSITION CORRECT	□	ORIFICE DIAMETER: ____________	
IMPULSE CONNECTIONS:		CORRECT TO HOOK-UP	□	MATERIALS CORRECT	□
		PRESSURE TESTED	□	TEST PRESSURE: ______________	
		STEAM/ELECT. TRACED	□	LAGGED	□
FIELD INSTRUMENT(S):		INSTALLATION CORRECT	□	AIR SUPPLY CORRECT	□
		WEATHER PROTECTED	□	POWER SUPPLY CORRECT	□
PANEL INSTRUMENT(S)		INSTALLATION CORRECT	□	AIR SUPPLY CORRECT	□
		SCALE/CHART CORRECT	□	POWER SUPPLY CORRECT	□
CONTROL VALVE(S):		INSTALLATION & LOCATION CORRECT	□	SIZE & TYPE CORRECT	□
		STROKE TESTED	□	POSITIONER CHECKED	□
		LIMIT SWITCH(ES) SET	□	I/P TRANSDUCER CHECKED	□
AIR SUPPLIES:		CONNS. CORRECT TO DRAWINGS	□	BLOWN CLEAR & LEAK TESTED	□
TRANSMISSION	PNEU:	LINES INSPECTED, BLOWN CLEAR & LEAK TESTED		□	
	ELECT:	INSULATION CHECKED - CORE TO CORE	□	CORE TO EARTH	□
		CONTINUITY CHECKED	□	LOOP IMPEDANCE CHECKED	□
		EARTH BONDING CHECKED	□	ZENER BARRIERS CORRECT	□
TEMPERATURE LOOPS:		T/C OR R/B CHECKED	□	CABLE TO SPECIFICATION	□
		CONTINUITY CHECKED	□	LOOP IMPEDANCE CHECKED	□
GENERAL:		SUPPORTS CORRECT	□	TAGGING CORRECT	□

CHECKED BY: DATE: WITNESSED BY: DATE:

LOOP TEST:

MEASUREMENT	TRANSMITTER INPUT	TRANSMITTER OUTPUT	LOCAL INST. READING	PANEL INST. READING	
	0				
	50%				
	100%				

CONTROL	CONTROLLER INPUT	TRANSDUCER OUTPUT	VALVE POS'NR OUTPUT	CONTROL VALVE POSITION	
	0				
	50%				
	100%				

REMARKS: ____

____

CHECKED BY: DATE: WITNESSED BY: DATE:

ACCEPTED BY: FOR: DATE:

INSTRUMENT TAG NO.

APPENDIX 23E
INSTRUMENT-RELATED PLANT DOCUMENTATION

This appendix describes the typical contents of instrument application design documentation used by the industry. It is essential that the system of documentation used is suitable for its purpose, properly implemented, and auditable in order to support validation and future maintenance activities.

Document Control

Each document (drawing, schedule, specification, etc.) should contain a title block or front page (as appropriate) that contains the following information:

- Document title
- Document number
- Document revision number or code
- Project name and number
- Site name
- Plant area (if applicable)
- Type of document (e.g., instrument schedule, loop diagram, emergency shutdown system specification, temperature transmitter specification sheet)

Each document should contain the following change control and author information:

- Reason for issuing the document (e.g., "Issued for customer comment," "updated in accordance with change note 123")
- Name of the document originator or modifier, as applicable, and the date of completion
- Name of the document checker and the date checked
- Name of the document approver and the date approved

Each document should contain cross-references to other design and/or reference documents, as applicable. Examples include the following:

- Process design documentation (P&IDs, process data specifications, etc.)
- Instrument design documentation (schedules, drawings, and specifications)
- Customer site or company standards

Instrument Schedule

An instrument schedule lists all instrumentation on the project, grouped by its unique tag (loop) number. For each instrument, the instrument schedule will typically provide the following information:

- Unique tag number
- Service/duty description
- Equipment description/type
- Location (e.g., pipe, process unit, or panel number)
- Manufacturer
- Requisition number
- Process design drawing number (P&ID, ELD, ULD, etc.)
- Specification/data sheet number
- Electrical hookup (loop or wiring diagram) drawing number

- Pneumatic hookup drawing number
- Process hookup drawing number
- Control system I/O address or tag number
- Notes/comments

Instrument Specification/Data Sheets

Instrument specification/data sheets provide the technical specification and design data for each unique type of instrument on the instrument schedule. They are used for purchasing the equipment, for providing design information for other disciplines (e.g., the definition of instrument tag number, signal, and range that are essential for the design of any associated computerized control system), and as the basis for calibration data.

Similar items can be included on the same specification sheet in separate columns, and identical instruments can be listed by tag number under common specification details. There are three main classes of instruments: pipe mounted, process unit (e.g., vessel) mounted, and field/panel mounted.

Each specification sheet should contain the following instrument data:

- Unique tag number
- Instrument type
- Supply voltage/pneumatic supply details (as applicable)
- Electrical/pneumatic connection type (as applicable)
- Signal type (e.g., 4–20 mA, 0.2–1.0 bar, serial, Fieldbus™)
- Type of mounting
- Range of instrument (calibration range) or switch set point values
- Materials of construction of wetted parts
- Control characteristics
- Other requirements (e.g., smart or standard instrument, Fieldbus™)

Each specification sheet should contain the following environment information:

- Process connection details (e.g., chemical seals, capillary lengths, flange rating)
- Ingress protection (IP) rating of the housing (e.g., weatherproof, dust-tight)
- Type of hazardous area protection (e.g., intrinsically safe, explosion proof)
- Requirement to meet EMC regulations

Each specification sheet should contain the following process data:

- Process fluid/material
- Engineering units
- Working range (of all the process variables affecting the measurement)
- Maximum range (of all the process variables affecting the measurement)
- The fail-safe mode
- Duty requirements (applicable to modulating control or actuated on/off valves [e.g., trip service or normal operation])

Each specification sheet should contain the required documentation needs, for instance:

- Factory calibration certificates (full certificate or batch certificate of conformance)
- Testing/calibration equipment stipulations (e.g., traceable to national standards)
- Manufacturer’s O&M Manuals
- Approval certificates for equipment in hazardous areas

- EMC Declaration of Conformity certificates (self-certification) for equipment containing EU CE-marked equipment (e.g., panels)
- Layout drawings showing overall dimensions
- Electrical schematic wiring and/or pneumatic connection diagrams
- Valve sizing calculations
- Number of copies of each document required

Each specification sheet should contain the required calibration[11] and testing needs. Consideration should be given to how any associated computerized control system will be tested to ensure conformance to the URS. Typical information to be considered includes the following:

- Reviews of algorithms and calculations (smart instruments, loop controllers)
- Representative testing across the full operating range, including range boundaries
- Testing of alarms and hard-wired interlocks
- The need for specific test data, conditions, or equipment
- Records of test results
- References to specific calibration and testing procedures

Cable Block Diagrams

Cable block diagrams show in schematic form the panels (e.g., field panels, computer system cabinets, marshaling racks) and large items of equipment (e.g., packaged plant), the cables connecting them, and any intermediate junction boxes through which cables are interconnected. Each diagram should contain the following:

- Blocks uniquely representing instruments, combined systems (e.g., panels, PLCs), and large equipment items (e.g., analyzers)
- Interconnecting lines representing cables, referenced by unique cable numbers
- Clear representation of the location of each block in the plant

Cable Schedule

A cable schedule is a list of all field instrument cables in identification number order and is used for allocating cable numbers and providing installation information. Each schedule should contain the following for each cable:

- Unique cable reference number
- Cable source, location/routing, and gland type
- Cable destination, location/routing, and gland type
- Number of cores/pairs
- Estimated route length and route identification, if possible
- Cable specification/data sheet number
- Identification of special circuits (e.g., IS circuits)
- Any appropriate notes on segregation requirements for installation purposes (e.g., distance from electrical power cables)
- Other information if required (e.g., drumming details, termination details at each end)
- Notes/comments

Pneumatic Tubing Schedules

A pneumatic tubing schedule is a list of all field instrument pneumatic tubing (single and multitube) in identification number order and is used for allocating tubing numbers and providing installation information. Each schedule should contain the following for each tube/multitube:

- Unique tube reference number
- Tube source, location/routing, and bulkhead fitting type
- Tube destination, location/routing, and bulkhead fitting type
- Number of tubes
- Estimated route length and route identification, if possible
- Tube specification/data sheet number
- Notes/comments

Termination Drawings

Termination drawings are used to allocate cables to particular sets of terminals inside panels (e.g., field panels, system cabinets, marshaling racks), junction boxes, and plant equipment. Each drawing should contain the following for each core:

- Unique cable reference number
- Core reference number
- Terminal number at source and source location
- Terminal number at destination and destination location
- Core outer sheath color/number
- Core size
- Any appropriate notes on segregation requirements (e.g., IS circuits, analog signals, security)

Miscellaneous Label Schedules

A miscellaneous label schedule is a schedule of all labels used on the project (e.g., instrument labels, junction box labels) and their specifications. Each schedule should contain the following information for each label:

- Equipment reference
- Dimensions of label
- Dimensions of character height
- Engraving details (including line separators)
- The numbers of each type of label required
- Label material type (e.g., Traffolyte, stainless steel)
- Label colors (e.g., Traffolyte sandwich colors)

Field Panel Specification and Drawings

Each specification should contain the following panel data:

- Panel reference
- Details of mounting or fixing
- Materials of construction and surface finish
- IP rating (e.g., weatherproof, dust-tight)
- Type of hazardous area protection (e.g., increased safety, air purged)
- Requirement to meet EMC regulations

Each specification should refer to all associated drawings, for example:

- Front of panel layouts, including label information
- Interior layouts, including label information
- Internal wiring/pneumatic tubing drawings

Each specification should refer to appropriate standards, for instance:

- General panel/junction box specifications
- Cable/tubing standards

Each specification should refer to test requirements and certificates, covering:

- Electrical power and/or pneumatic checks
- Functional checks

Field Junction Box Drawings

Field junction box drawings can range in complexity from a simple cable termination box or cabinet, to a local control station containing electrical/pneumatic equipment. Complex junction boxes may require information similar to that described above for instrument panels (e.g., specifications, drawings, test certificates), depending on the level of complexity involved.

Junction boxes used for cable termination/routing require the following drawing details:

- Junction box reference
- Details of mounting or fixing
- Materials of construction and surface finish
- IP rating (e.g., weatherproof, dust-tight)
- Type of hazardous area protection (e.g., increased safety, air purged)
- Interior layouts, including any label information
- Internal wiring details (cable and core identification, cable entry and glanding, etc.)

Electrical Hookup (Loop or Wiring Diagrams) Drawings

Instrument loop diagrams are the key design documents for instrumentation systems and are also the main troubleshooting tool for fault finding and maintenance activities on-site. Single loop diagrams, showing all items in the loop, tend to be the industry standard. Each loop diagram should contain the following functional information:

- All instrument items in the loop
- All control items in the loop, including loop controllers and computerized control system I/O card address and termination details; the I/O details would need to be provided by the computerized control system designers
- The cables connecting each applicable item in the loop along with their identification numbers and, for multicore cables, pair number/identification
- Termination details (terminal block identification and terminal numbers)
- Electro-pneumatic equipment/circuit information (e.g., control valves and other electro-pneumatic valve circuits) is usually shown on the electrical loop diagram

As applicable, each loop diagram should contain the following additional information:

- Hazardous area classification zone, temperature classification, and gas group
- Instrument type and tag number
- The diagram should show the power flow from its source, often in a control room, through the various locations of any intermediary junction boxes, barriers, or switches, and out to the load and back; the different locations through which each cable passes (e.g., control

room, interface room, field panel, junction box) should be represented clearly so that the full extent of the different environments is conveyed precisely

Pneumatic Hookup Drawings

Pneumatic hookups are the pneumatic versions of instrument loop diagrams and serve a similar purpose. They are produced for control loops that are purely pneumatic. For loops with a small pneumatic element (e.g., control valves and other electro-pneumatic valve circuits) the pneumatic circuit is usually shown on the electrical loop diagram. Each drawing should contain the following:

- All filter/regulation sets, piping, valves, etc. relating to the pneumatic hookup
- Details of all components and their connections/fixings
- Air supply pressure before and after controlling devices and at strategic points in the circuit
- Installation notes

Process Hookup Drawings

Process hookups are drawings showing the physical method for connecting and mounting instruments connected directly to process piping and equipment. The drawings are used for costing purposes by installation contractors and for the subsequent installation of instrumentation. Each drawing should contain the following:

- The instrument and its method of connection to the process piping/equipment
- Instrument mounting arrangement
- All piping and fittings required to install the instrument (material takeoff)
- Installation notes

Instrument Layout (Location) and Cable/Tubing Routing Drawings

Layout and routing drawings are floor plans of the building/process area showing the locations of process equipment and instrument panels. Each drawing should contain the following:

- The “spot” location of each instrument and its tag number
- The location of all instrument panels and junction boxes and their identification numbers
- Main instrument tray/ladder rack routes for cables and tubing and all tray/trunking branches (the identification numbers of the cables/tubes on each route and branch should be shown for installation purposes)
- The size and specification of each instrument tray/ladder rack and its height above finished floor level
- Elevations, as necessary, to help prevent potential clashes with other items of tray work/ladder racking and with other services (e.g., pipework and HVAC [heating, ventilation and air conditioning] ducting)
- Separation distances from other cables and other services, on both plan and elevations, where compliance is necessary for safety reasons (e.g., intrinsic safety), for functional reasons (e.g., analog signal integrity), or for electrical interference reasons
- Details of any transit frames or holes required through walls or floors to permit the installation of cables and tubing

Earthing Schedules and Drawings

Earthing schedules and drawings detail the preferred methods for earthing various types of equipment and its associated earth testing information. The documents should address the following (as applicable):

- Static earthing
- IS earthing
- Computer earthing

The earthing drawings should show how equipment should be connected for each of the above situations. Each item to be earthed should have an entry in the appropriate earthing schedule that details the following information:

- Plant item to be earthed and plant area location in which the item is to be found
- Earth bond size and location of earth bar to which the item is connected
- Earth test results (resistance in ohms) and date tested

Miscellaneous Drawings (Control Room/Instrument Room Layouts)

There is usually a need to provide additional drawings to clarify certain aspects of the design. A typical example is the provision of layout drawings for instrument/interface rooms and for control rooms to show the location of key items of equipment and their relationship to other equipment that may already be installed.

Instrument Installation Specification

An installation specification should be prepared for issue to prospective installation contractors for tender. The specification forms the technical section of the construction documentation package that should be appended to the specification. Each specification should contain, or provide detailed reference to, the following information:

- Regulations and codes of practice
- Company/work rules
- Competence levels required of the installer
- Safety standards required of the installer
- Good practice guidelines for the installation of instrumentation
- Good practice guidelines for the installation of cabling
- Good practice guidelines for the installation of instrument piping/tubing
- Good practice guidelines for calibration management[11]

Each specification should incorporate a brief scope of work and refer to the construction documentation package that should be appended. The construction documentation package will typically comprise the following drawings and schedules:

- Instrument schedule
- Instrument specification/data sheets
- Cable schedules
- Pneumatic tubing schedules
- Termination drawings
- Miscellaneous label schedules
- Field panel specifications and drawings
- Field junction box drawings
- Electrical hookup (loop or wiring diagrams) drawings
- Pneumatic hookup drawings
- Process hookup drawings
- Instrument layout (location) and cable/tubing routing drawings

- Earthing schedules and drawings
- Miscellaneous drawings (control room/instrument room layouts)
- Electrical and instrumentation interface panel drawings
- Any special instrument wiring diagrams

Each specification should refer to the following test requirements and certificates:

- Loop test sheets
- Site calibration check sheets[11]
- Earth loop tests
- Insulation tests
- Completion certificates

Package Plant Instrument Specification

The package plant instrument specification details the instrumentation requirements and standards to be applied to any equipment packages. Typical equipment packages include Water for Injection (WFI) systems, chiller packages, gas scrubber packages, tablet presses, packaging machines, freeze dryers, autoclaves, and so on. Each specification should contain, or provide detailed reference to, the following information:

- Regulations and codes of practice
- Company/work rules
- Safety standards required of the supplier
- Good engineering practice guidelines for the installation of instrument apparatus
- Good engineering practice guidelines for the installation of cabling
- Good engineering practice guidelines for the installation of pneumatic piping/tubing
- Good practice guidelines for calibration management[11]

Each specification should contain details of the local power supplies available and refer to the necessary drawings and schedules, for example:

- Supply voltage and air pressure
- Cable block diagrams describing the interface with the rest of the plant
- Cable schedules detailing interconnections with the rest of the plant

Each specification should refer to test requirements and certificates, for example:

- Instrument hazardous area certificates
- Calibration certificates[11]
- Requirement to meet EMC regulations

Electrical and Instrumentation Interface Panel

The electrical and instrument interface (interposing relay) panel is the method used to interface low voltage control systems (e.g., PLCs, Distributed Control Systems [DCSs], shutdown systems, loop controllers, etc.) to electrical switch circuits for operating and monitoring items of electrical equipment, including equipment packages. The documentation requirements for the panel comprise a panel specification, panel layout drawings, and wiring schematics (see above).

Special Instrument Systems

A number of other instrumentation and control equipment may be required for a particular project. These could include:

- ESD systems
- Analyzer packages
- Laboratory instrumentation (mass spectrometers, gas chromatographs, etc.)

Each item would generally require a detailed specification and installation documentation. The standards and documentation detailed in the appropriate sections above should be applied to this equipment. In addition, if the equipment involves the use of a computerized system, the system (application software and hardware) should be developed and tested using a formal, life-cycle methodology.[7,9]

24 Case Study 6: Programmable Logic Controllers

Rob Stephenson, Pfizer
Stephen C. Giles, Pfizer

CONTENTS

This case study discusses the validation of Programmable Logic Controllers (PLCs) and the particular requirements within a bulk pharmaceutical chemical environment. Emphasis is on bespoke systems, but the requirements of prebuilt PLC package systems are discussed in the section on embedded PLCs.

PLCs are microcomputers that have input/output (I/O) connections that enable them to communicate with external devices, usually in an industrial environment. Communication includes monitoring a state, detecting a change of state, activation, or deactivation. Devices include actuators, switches, thermocouples, and other instruments. PLCs can be programmed to process the incoming signal and, if required, respond with an appropriate output signal to enable control.

PLCs originated in the car industry in the 1970s where programmable logic devices were used to do repetitive tasks. Within the pharmaceutical industry, their use has evolved over the past 20 years as their processing power has increased. Complex nonrepetitive tasks are now routinely performed, and as a consequence PLCs are now used to control the majority of medium-sized manufacturing facilities. The worldwide market for PLCs has been estimated to be on the order of $3 billion.

The use of PLCs provides an inherent flexibility for the automation of a production plant. They can be used as an independent control system either embedded into or remotely linked to one or

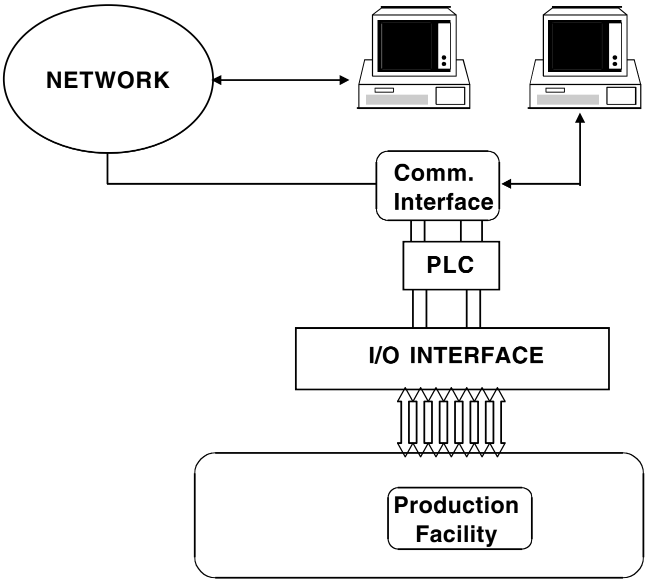


FIGURE 24.1 Example PLC Application Layout.

more items of production equipment. Expanded automation can be achieved by networking PLCs together or using them as slave stations to a central Supervisory Control and Data Acquisition (SCADA) system or Distributed Control System (DCS). Figure 24.1 shows an example layout.

VALIDATION OF AUTOMATED SYSTEMS

Validation of automation systems is now expected by the regulatory authorities governing the production of pharmaceuticals. It features extensively in articles in the literature and guidelines produced by both industry[1,2] and regulatory bodies.[3–6] The main reasons for validating PLC-based systems are as follows:

- ***Getting an application to work the first time:*** This is the aim of most automation projects, and the validation process assists in fulfilling this objective. Failure will have significant cost penalties in lost production output.
- ***Obtaining the full potential of a PLC system:*** PLCs can be regarded as black boxes to those who are not familiar with their functionality and capabilities. Thus, they may not be used to their full potential. A team approach to validation can help all involved, particularly if a user is part of the team, leading to a better understanding of how the PLC-based system controls their plant.
- ***System maintenance:*** Validation results in a well-documented system, which is a prerequisite for efficient planned maintenance schedules and for effective troubleshooting.
- ***Change management:*** PLC systems tend to be dynamic in that they are constantly being refined. Application software is regularly modified to cope with new processes, user requests, or changes to the hardware. Validation provides the documentation necessary for the process automation engineer to have required information readily at hand in order to make the requested changes.
- ***Pharmaceutical regulatory compliance:*** The benefits of applying Good Engineering Practice to an automated system are identical to those above. The additional effort required to validate a new system according to guidelines such as GAMP[2] is likely to be only a small additional element of the project cost. For this reason, allocation of

resources to prospectively validate new PLC systems or a major modification to an existing PLC is generally easy to justify. It is much more difficult to justify the required resources to retrospectively validate a PLC-based system that has, for many years, been controlling plant and equipment manufacturing product that meets its quality specifications, and is not subject to ongoing development. Regulatory compliance is really the only reason if the PLC-based system is in a well-managed plant.

- ***Other compliance:*** The control of a production facility using PLCs also has ramifications for compliance with other statutory regulations such as health and safety legislation and environmental regulations. Addressing all of these requirements in one validation study can help to focus effort and resources at the appropriate time. In addition, one set of documentation, properly written, approved, and controlled, can minimize duplicative work. Some companies prefer to keep separate compliance documents for each regulatory body. This is a matter of choice.

SELECTION OF A PLC

A lot of effort goes into selecting a PLC platform, since once selected, in-house experience is accumulated in utilizing its capability and functionality. Once a particular system is in use, it takes a lot of justification to change. In practice, the main criteria that influence selection of a PLC-based system are as follows:

- Pharmaceutical company-preferred suppliers
- Conformance to IEC 61131-3: structured design, standards for reusable software, and interchange of software between different PLC products
- Hardware/software capabilities
- Supplier characteristics: durability, future proofing, and cost
- In-house and contract programming knowledge availability
- Supplier has quality management systems in place and in use
- Cost

Supplier Audits can be used to obtain some of the knowledge required to assess the above criteria.

The selection of the PLC platform is critical to project success. The capabilities of the system will affect:

- The degree of automation
- Equipment Performance
- Efficiency Benefits
- User Benefits
- Maintainability
- Improvement potential/flexibility
- Scaleability

Customers are looking for confidence in the ability of a product to meet its business needs. Obtaining this confidence involves evidence gained from the Supplier Audit with other data which may be more subjective.

VALIDATION OF A PLC-BASED SYSTEM

The methodology used to validate a PLC-based system is very similar to that used to validate other automated systems that electronically communicate with external devices. Validation is generally

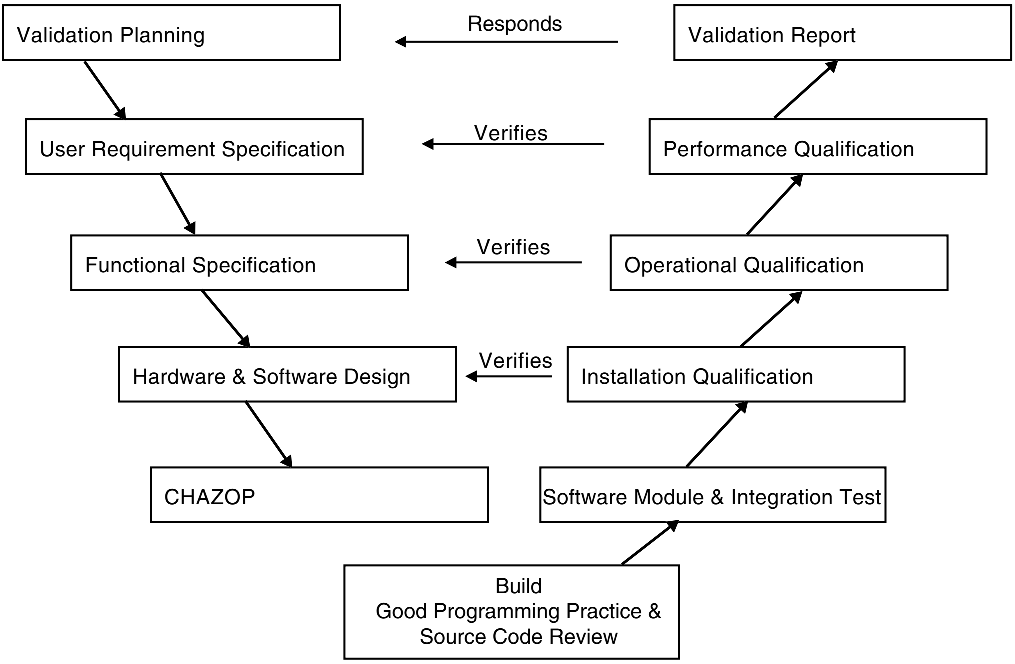

Validation Planning
Responds
Validation Report
User Requirement Specification
Verifies
Performance Qualification
Functional Specification
Verifies
Operational Qualification
Hardware & Software Design
Verifies
Installation Qualification
CHAZOP
Software Module & Integration Test
Build
Good Programming Practice &
Source Code Review

FIGURE 24.2 Validation Life Cycle.

viewed as a life cycle. There are many different types, but the current preference in the pharmaceutical industry is for the V-Model.[1] It is important to recognize that a PLC must be validated in conjunction with its Human Machine Interface (HMI). If a PLC is being programmed to be connected to an existing networked system or to an existing SCADA or DCS HMI, then, unless some changes are required to the HMI, the PLC can be validated in isolation using the standard V-Model (Figure 24.2). Only the interaction between the PLC and the HMI need be validated. If, however, the application involves configuration of the HMI and programming of the PLC, then a variation of the V-Model is appropriate (Figure 24.3). This is the most likely situation that will be faced.

The difference between a DCS, where the HMI is usually an integral part of the system, and a PLC, which usually contains a controller and an HMI, is that separate software and hardware design specifications exist, and there will be separate build and Module Testing. Additionally, Integration Tests must take place to ensure that the individual systems communicate and interact in the correct manner. Any assessment of the potential hazards of such a system, such as a Computer HAZard and OPerability (CHAZOP) study, should cover the total system.

THE VALIDATION PROCESS

The validation process can be divided into several phases. These phases have distinct requirements but should not be considered in isolation:

- Planning
- Specification
- CHAZOP
- Build
- Testing
- Operation and Maintaining the Validated State

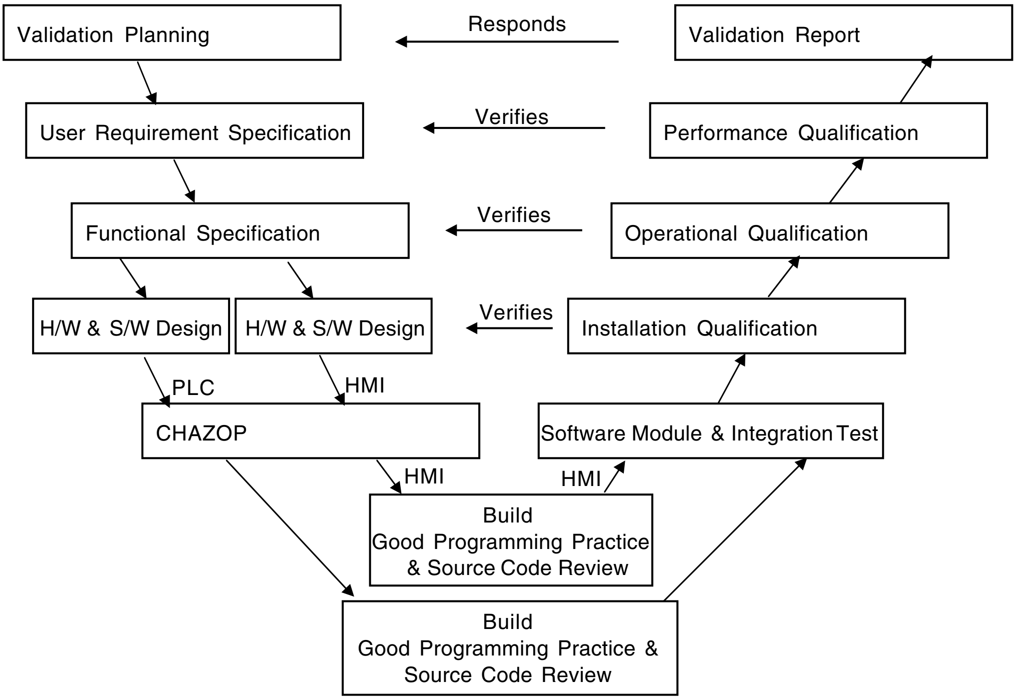

Validation Planning
Responds
Validation Report
User Requirement Specification
Verifies
Performance Qualification
Functional Specification
Verifies
Operational Qualification
H/W & S/W Design
H/W & S/W Design
Verifies
Installation Qualification
PLC
HMI
CHAZOP
Software Module & Integration Test
HMI
HMI
Build
Good Programming Practice
& Source Code Review
Build
Good Programming Practice &
Source Code Review

FIGURE 24.3 Modified Validation Life Cycle for PLCs and Their Associated HMIs.

Any validation effort requires a team approach for efficient execution. The key skill areas required are as follows:

- Production Engineering — knowledge of the plant and equipment
- Automation Engineering — knowledge of the PLC and the HMI
- Validation Engineer — knowledge of the validation process
- Quality Assurance — knowledge of Good Manufacturing Practice (GMP) and Good Documentation Practice

The representatives of these areas on the team have to work together and have to be committed to validation, otherwise conflicts and delays will inevitably follow.

Planning

Determining and documenting how a PLC-based system is going to be validated, in the form of a Validation Plan, is an integral element of the validation life cycle. A Validation Plan defines the strategy for establishing an appropriate level of documentation to demonstrate that a PLC-based system functions in a manner consistent with its specification and does not in any way impair product safety, quality, and efficacy. A Validation Plan should be produced for all validation projects. The components of a Validation Plan should include the following:

- Authority for the validation (e.g., Validation Policy, Validation Master Plan)
- Scope and purpose of the validation
- Overview of the process to be controlled
- Description of the environment in which the PLC-based system is to function
- GMP implications
- Safety Implications — IEC 61508

- Background to the PLC installation
- Organization and responsibilities
- Validation approach and the procedures to be used
- Results of any supplier selection process
- Ongoing operational requirements for GMP
- Ongoing operational requirements for IEC 61508
- A milestone plan
- An historical summary of the PLC-based system — if modifying or retrospectively validating an existing system

It is the responsibility of the Business Owner for the process (i.e., the plant manager) to ensure that a Validation Plan is in place. However, its preparation can be delegated. Typically a Validation Plan for a PLC-based system would be prepared by a validation specialist and approved by the process automation engineer and the plant manager.

The main purposes of the Validation Plan are as follows:

- Get management to commit suitable resources to the validation effort.
- Make those involved in the validation process aware that their skills are required.
- Define what documents need to be prepared and who is going to prepare them.
- Demonstrate that validation is not an exercise at the end, but is integral to achieving good practice throughout the project and system life cycle.

The Validation Plan is produced at about the same time as the URS. The information in the URS is summarized in the Validation Plan with emphasis on GMP and Safety critical issues.

Specification

A critical step in any validation life cycle is establishing the User Requirements Specification (URS). This is the document against which the system performance is ultimately assessed. It is therefore worth spending time and effort getting this document right. In our experience, users generally write incomplete URSs due to a lack of detailed knowledge of PLCs. This inevitably causes delays and problems getting validation right, as it is difficult to determine what is being validated against. However, users do know how they want their systems to operate, the features necessary for an efficient process, GMP, safety, and environmental constraints. We have instituted an Outline Specification (OS) in which they describe the critical parts of the system from a user perspective. It is an early warning that a PLC-based system project is in the offing so that realistic plans can be made in busy work schedules. The OS is then developed into a URS by an engineer who is more familiar with the intricacies of PLCs. Care must be taken at this stage that the URS does not become an SDS. Our experience is that each project is different, people change, and the content of various documents must be assessed during the life cycle of each project. The URS must highlight mandatory features and those that are desirable. Example sections of a URS are presented in Table 24.1.

This information is then used by the supplier/programmer to develop a Functional Specification (FS). The FS describes the operational and performance criteria to be provided by the PLC-based system. It consists of the same section layout as the URS, giving an overview of the PLC system hardware and software required to fulfill the URS, and notes any noncompliances. Programming PLCs is becoming easier, and we have found spreadsheets invaluable in consolidating the I/O requirements requested in a URS. These spreadsheets can be built up and form part of the FS and detailed design, then used to generate the test sheets for Installation Qualification (IQ) and Operational Qualification (OQ). Figure 24.4 illustrates this approach. Other application tools that support this process are also emerging.

TABLE 24.1
Some of the Elements of a URS

Section	Contents	Examples
Description of the Automation Project	Process description	Critical variables
		GMP implications
		Process limits
	PLC description	Contribution to production
		Degree of automation required
		Production considerations
		Engineering considerations
	Constraints	Timing
		Equipment availability
		Engineering constraints
	Operational environment	Zoning (e.g., Zone #1, Zone #2, and SAFE)
		Cleanliness
		Potential electromagnetic sources
Interfaces	Human machine interface	Number and type of terminals
		Required displays in diagrammatic form
		Access security requirements and conformance with standards
	Process interface	I/O capacity
		Installed and future I/O spare capacity
		Communications interfaces
Control Requirements	Process control	Sequence control
		Continuous control
		Batch management
System Attributes	Alarms	Principle
		Method of viewing
		Method of acceptance
	Events	Change in status
		Trending of data
	Interlocks	Hardware
		Software
Other Requirements	Performance/response time	
	User training	
	Maintenance	
	Redundancy	
	Intrinsic and operational safety	
	Expansion philosophy	
	Security and integrity	

The FS is developed into a hardware design specification (HDS) or a software design specification (SDS) or both. The SDS provides the facility to break each functional requirement into appropriate subfunctions, increasing in detail at each subsequent level, until eventually the design reaches a state where it can be translated into software code. The SDS should mimic the sections of the FS but be specific to software construction. Some examples are as follows:

- Design methodology (e.g., sequential function charts)
- Module design: inputs, functionality, and outputs

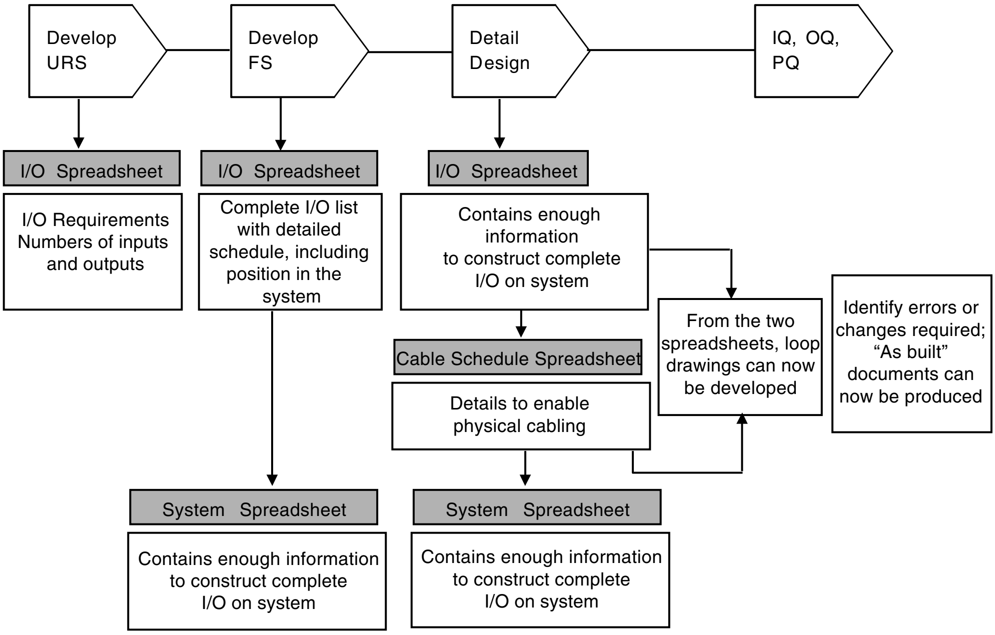

Develop URS
Develop FS
Detail Design
IQ, OQ, PQ
I/O Spreadsheet
I/O Requirements Numbers of inputs and outputs
I/O Spreadsheet
Complete I/O list with detailed schedule, including position in the system
I/O Spreadsheet
Contains enough information to construct complete I/O on system
Cable Schedule Spreadsheet
Details to enable physical cabling
From the two spreadsheets, loop drawings can now be developed
Identify errors or changes required; "As built" documents can now be produced
System Spreadsheet
Contains enough information to construct complete I/O on system
System Spreadsheet
Contains enough information to construct complete I/O on system

FIGURE 24.4 Spreadsheet Development.

- Communications interface (data transfer protocols)
- Process interface (consideration of the signal from the instruments)

Hardware design defines the PLC platform that hosts the software, including the relationship with the HMI. It is often an overlooked part of PLC-based system validation. Areas that need to be addressed include the following:

- PLC hardware
- Interfaces: human, process (analog I/O, digital I/O, totalizers, interposing relays), and communications interface (hardware protocol, buffer capabilities, cable specifications, line drivers)
- Services (earthing, filtering, loading, surge protection)
- Maintenance procedures (spares, diagnostic checking, calibration)

The instrument supply industry is moving toward smart devices, with corresponding software to provide configuration, testing, and monitoring. Communication trends are starting to move away from the 4–20 mA analog signal toward digital signals utilizing one or more of the many bus technologies that are available on the market today and the IEC 61158 and 61784 standards that apply a framework for their application. The applicability of these developments to the project or staying with the tried and tested methods needs to be considered in the HDS.

Validation is no different from many processes, inasmuch that an inadequate specification will lead to extended project times, higher costs, and increased risk of failure. Our experience merely confirms the above. Get the specification phase right and the rest of the validation life cycle will be relatively smooth sailing.

Computer Hazard and Operability Study (CHAZOP)

Some companies would not include CHAZOP as part of the computer system validation. However, the CHAZOP process can provide a formal mechanism for reviewing and assessing a system design.

A CHAZOP provides a structured and documented process for evaluating the risks (failure modes) of a PLC-based system that the standard Hazard Study (HAZOP) has identified as having an impact on plant safety. It should also explore remedial actions. It is not strictly a test, but a design review intended to examine how control systems deviate from designed function (e.g., produce incorrect output for a given input) and the effect that this would have. It is the first formal step in assessing that the proposed automated system is fit for purpose. The process itself is similar to a HAZOP, in that it uses keywords to examine whether the safeguards currently in place/proposed will adequately prevent the deviation or mitigate its consequences. To complete a CHAZOP requires a team of people providing knowledge of the PLC, HMI, the process, the equipment being controlled, and an independent leader familiar with the study process.

CHAZOP should be completed after design, but before programming or build has progressed too far so as to minimize abortive work. Our experience is no better than most as this approach is immature. We have learned that a CHAZOP must follow a HAZOP (where a HAZOP is required) and not vice versa, otherwise it becomes difficult to focus the CHAZOP on the automation system without exploring the consequences on the associated equipment. The best way to learn about the CHAZOP process is to do it in a controlled, planned manner.

Build

Once the prior phases have been completed, the supplier can get on with the task of building and programming. Putting sufficient time and effort into producing good specifications considerably eases programming. Testing begins during this phase.

Testing

The testing of PLC software and hardware is in a number of different phases during the life cycle of the project and is designed to achieve different things.

Source Code Reviews are used to ensure that controlled development and a consistent programming style have been used. It also checks GMP critical features. Source Code Reviews need a standard against which to check, and programming practices for all programs should be documented. An audit of the supplier's compliance with its own quality management procedures should have been carried out to ensure that appropriate quality is being built into the system.

Programming details will vary according to the PLC manufacturer, but the overall style should be consistent. Areas that should be addressed are as follows:

- Directory structure
- Software construction
- File header and commenting
- Version control
- File naming conventions
- Coding practice (e.g., naming conventions for ladder logic, special functions, subroutines, and loops)
- Functions, subroutines, and loops

Source Code Reviews of PLC software are not easy tasks as programs are constructed using low level code such as ladder logic. It is not generally the practice to review all code, as this would take considerable expertise, time, and associated cost. There is no guarantee that all errors would be identified. Effort should be directed to critical areas of the code and high level coding conventions to assure that software has been constructed in a structured manner and contains clear concise comments. The quality of these sections of code should act as a guide to the overall quality of the software.

The next phase of testing begins to investigate whether the implemented design conforms to the requirements of the design specifications and Functional Specification. This process starts with

Module and Integration Testing conducted by the supplier. Completion of this is otherwise known as Factory Acceptance Testing or off-line testing. Module Testing of functional routines verifies the components of the design in isolation. Integration Testing assembles the various functional routines and tests the system as a whole. These tests provide the best opportunity to challenge the system in ways that are unlikely to be faced in operational use, and, as such, they should be carefully designed to test:

- Databases used to develop the design have been implemented correctly
- Devices
- Loops
- Fault trees
- Sequences
- Interface with HMI
- Invalid inputs

What is required, as previously stated, is to validate the PLC, the HMI, and the connection. We can use the HMI to help us do Integration Testing. The sequence is as follows:

- I/O on PLC
- Devices on PLC
- I/O via database on HMI
- Devices on HMI
- Sequences

The careful design of tests can ensure that unlikely conditions can be assessed and their impact on the system evaluated. Errors in system construction may also be identified. Comprehensive testing builds confidence into a system and can help reduce subsequent OQ testing (Site Acceptance Testing or on-line testing). This can often produce savings in time, particularly in plant downtime if modifications are being made to an existing system.

It is not feasible to test every aspect of the programs. Tests must be directed toward critical features. Coordination between the customer and supplier can help to minimize superfluous tests. The trend toward incorporating the programming standard of IEC 61131–3 means that validation becomes easier for each new system for which a PLC is used. There are only so many devices a PLC will support, most of which will be common to each system. We have found that the use of generic descriptions of how to test a device and automatic generation of specific test sheets from a database saves a considerable amount of time when testing new systems. There are some software tools available, such as simulation software, that can aid testing. However, this software must be validated before use.

Since Module and Integration Testing is done in isolation from the process and equipment, confirmatory tests are required, and these form part of the OQ. Before this can be completed, proof is required that hardware and software are installed correctly. This is done at IQ, which ensures that all the PLC-based system components are present and installed correctly in accordance with design specifications. Some representative IQ tests are included in Table 24.2.

OQ ensures that each function of a PLC-based system performs in accordance with its FS throughout representative or anticipated ranges. The boundaries of the system should have been tested in the Module and Integration Tests, but a proportion of these tests must be repeated in OQ now that real devices are attached. Vagaries in the devices can have effects on the operation of the software and these need to be evaluated. OQ tests include the following:

- Hardware/software startup and operation
- Operation of devices

TABLE 24.2
Example Installation Qualification Tests

Item	Tests
PLC	Confirm all parts are present and correct
	Confirm power supply voltages and quality are adequate
	Confirm earthing meets manufacturer's requirements
	Confirm correct versions of software have been supplied and installed
	Confirm hard-wired interlocks have been correctly installed
I/O Cards	Identify serial numbers and model numbers of cards
	Identify calibration certificates
I/Q Tests	Confirm signal continuity between field termination and PLC
	Confirm signal continuity between HMI and PLC
Documentation	Check that all required documentation is present and useable

- System timing, in particular any handshaking that may be done with other systems
- Software failure and restart routine
- Critical fault trees
- Security features, such as password protection
- Control sequences
- Ensuring GMP-critical parameters have been implemented correctly
- Field I/O is correctly calibrated
- HMI interacts with the PLC in the desired way
- Backup and restoration

We have used a database not only to populate PLC and SCADA values, but also to generate forms used in the testing. These forms, referenced by each protocol, can help to minimize documentation, but, more importantly show at a glance the continuity of completed tests.

Other GMP issues that need to be addressed during IQ and OQ include the following:

- Training
- Standard Operation Procedures (SOPs)
- Calibration Schedules, both GMP and IEC 61508
- Maintenance Schedules

The final phase of validation is Performance Qualification (PQ), which ensures that the PLC-based system performs its functional requirement as specified in the URS within the manufacturing environment. In reality this means using the PLC-based system to control the process while product is being manufactured.

Typical PQ tests include the following:

- Batch startup
- Monitoring of GMP crucial parameters
- Monitoring of alarms and messages
- Confirming security access procedures
- Confirming HMI is fully functional
- Confirming batch recording is correct

A final Validation Report is required in order to summarize the results of testing and assign a validated status to the system in its operating environment. This is usually performed by a validation specialist who may have been a witness to some of the tests but is impartial. This provides a knowledgeable but independent summary of the validation exercise. A validation certificate is sometimes used to demonstrate that a PLC-based system has been successfully validated and when the periodic review or validation assessment activity must be undertaken.

Operation and Maintaining the Validated State

The validation effort does not stop when the Validation Report is completed and the PLC is on-line. A number of support procedures must be implemented to maintain the PLC-based system in a validated state.

Change Control is the most important and ensures that future modifications to the PLC-based system can be managed in a controlled way without impacting GMP, safety, or environmental considerations, and in order to maintain the system in a validated state. Change control starts after IQ, superseding the project change control process which will have been in place during the development stages of the project.

Configuration Management is closely linked to change control and should be followed to ensure that a current and accurate statement of the PLC-based system components is maintained as changes occur.

Contingency Planning defines the plans and procedures that are required to ensure that system failure (hardware or software) can be recovered with minimal (quantifiable) impact on the validated state and without risk to safety, the environment, or product quality. Automation strategies incur the penalty that most, if not all, the “eggs are in one basket.” If a PLC goes down there can be serious consequences for a number of different controlled or monitored areas. If this is the case, then the URS should have specified some sort of backup system. Even if the plant can cope if the PLC goes down for a short period of time, procedures identifying equipment that are still operable, those that are in a safe condition, and areas with the highest risk (on whatever grounds this is assessed) are essential. The immediate availability of recovery procedures and access to a secure backup copy of the current version of the software is necessary in the event of an unplanned system failure.

Maintenance Planning can help to minimize the need for contingency procedures by having planned service outages to maximize the time between failures of the PLC-based system. A possible strategy is to carry a spare PLC and one each of the I/O cards. This is not an expensive option as the spare PLC can double up as a development machine.

Fault-Logging should be in place to allow the user to record any observations, problems, or suggestions for improvement. These should be reviewed along with change control records, a selection of production records, and training records in a regular review of the PLC application.

Periodic Reviews are an important element of a continuous improvement program. A topical issue at present is being able to demonstrate that source code has not changed. The use of PLC software tools that enable detailed comparison between software is a useful check during the Periodic Review, and an inspection of Change Control records, Contingency Plans, Maintenance Plans, logbooks, and security measures demonstrate that the automated system is under control.

VALIDATION OF LEGACY PLC-BASED SYSTEMS

The use of PLCs in some automated pharmaceutical plants may predate the current regulatory requirements relating to computerized system validation. There is, therefore, often a requirement

TABLE 24.3
Comparison of the Documentation Requirements for Prospective and Retrospective Validation

Document	Prospective Validation	Retrospective Validation
Validation Plan	Yes	Yes
User Requirements Specification	Yes	URS and FS may be combined document
Functional Specification	Yes	
Supplier Audit	Yes	Yes (if further development anticipated)
Software Design Specification	Yes	No
Hardware Design Specification	Yes	No
CHAZOP	Yes	No
Source Code Review	Yes	Yes
Installation Qualification	Yes	Yes
Operational Qualification	Yes	Yes
Performance Qualification	Yes	Yes
Validation Report	Yes	Yes

to validate existing systems. Changes in the use of a system or changes to the regulations themselves may also bring an existing system into the scope of a formal validation effort. Validation of legacy systems is often quite difficult as much of the information needed is difficult to locate, if it exists at all. It differs from prospective validation in ways that reflect how information is put together in a documentation package and in the amount of effort put into retrospective testing. The User Requirement Specification (URS) and Functional Specification (FS) may be combined into one document. A Software Design Specification (SDS) and a Hardware Design Specification (HDS) are not necessary, but the "in use" configuration of the system should be documented. A Source Code Review is desirable to assure the quality and maintainability of the code. Installation Qualification (IQ) should be confirmed against the current configuration document and Operational Qualification tests (OQ) may be repeated (or performed for the first time). Suggested documentation requirements for prospective and legacy system validation are summarized in Table 24.3. Legacy system validation requires significant resources, a risk assessment should therefore be performed in order to determine the scope of the validation effort and its priority in any overall validation program.

PLCs AS PART OF A SYSTEM

The validation of a PLC cannot be viewed as a stand-alone operation. PLCs are there for a purpose — to provide automotive functionality. A project Validation Master Plan that details what other validation (equipment qualification, validation of other automated systems) is required. An overall project plan timetable must be in place in order to ensure, e.g., that validation of the PLC does not adversely delay the qualification of important equipment. Where the PLC is to communicate with another PLC embedded into an item of equipment, the link as well as the validated, as well as the individual PLCs, but the link cannot be validated until the software processing the information from the particular I/O point is validated. The definition of the boundaries of each validation prevents unnecessary duplication. For PLCs, it is best to use the I/O cards as the boundary. A final check from HMI to the device can then be performed to ensure the system works in its entirety.

EMBEDDED PLCs

Embedded PLCs (where a PLC is delivered with the equipment used to control the process) tend to be standard packages. In these cases there is no intention to make the PLC field programmable and often the program is provided as firmware without comprehensive program documentation. It is now nearly impossible to dictate to proprietary equipment manufacturers about the type of PLC to be used, so you are looking for a system built to a good quality management standard. Supplier Audits are used to determine whether the implemented system is appropriate for your validation needs and what, if any, corrective action is required. The focus of testing shifts from Module and Integration Testing to OQ. Although the purchaser may have access to the supplier's test plans and results, test design and justification must be included in the OQ protocol to assure that the PLC functions are required in your manufacturing environment. Equipment suppliers may not be accredited PLC suppliers. The approach to validation will not change, but there may be increased emphasis on particular elements such as functional testing, and less on others such as Supplier Audits. This is very much a customer decision based on the customer's own experience.

The amount of testing depends on three factors:

- Conclusions from the Supplier Audit
- Criticality of the PLC in controlling the equipment
- Criticality of the equipment in your process
- Unnecessary tests cost time and money

Some PLCs can carry on functioning if the HMI is unavailable. More sophisticated PLCs are dependent on the HMI. It is important to identify this situation and validate accordingly. A current approach in industry today is to purchase a PLC and bolt on a SCADA system. The SCADA system tends to be written by a third party and must be configured for the application. Validation is not totally integrated and must be carefully considered during the validation life cycle.

ELECTRONIC RECORD/ELECTRONIC SIGNATURE (ERES) CONSIDERATIONS

The applicability of the U.S. 21 CFR Part 11 regulation[4] and other international GxP guidance on electronic records and electronic signatures[6] to PLCs and PLC-based systems is still the subject of discussion and interpretation within the industry. It is the purpose of PLC-based process control systems to receive, modify, and transmit electronic signals and data. Where systems do not retain any GxP-related records on a durable media (e.g., hard disk, tape, or other form of permanent storage) Part 11 is not applicable. However, many systems, although they do not retain raw data (temperatures, pressures, flow rates, etc.), may store important process data (recipes, control parameters, etc.) or configuration data (I/O scaling factors, etc.), the integrity and security of which may be important for the assurance of product quality, efficacy, or patient safety. Other electronic records such as PLC configuration and code may be retained in the event of loss of electrical power from the system and could also be subject to regulation.

Electronic records include data-storing memory cards used in embedded control systems such as steam sterilizers. These cards are used to transfer data (critical parameters, cycle times, etc.) from the control system to a main database application which is usually networked. A memory card is about the size of a credit card. Manual installation, removal, and transfer of the memory cards must be conducted in accordance with defined procedures. Reuse of cards must be carefully considered; data must not be deleted until confirmation is received that it has been successfully downloaded to the database. The local operating environment and the possibility of EMI/ESD damage to data integrity of the memory card must also be evaluated.

Records held in memory, e.g., temperature profile from an Autoclave, and retained for a period (say, 10 batches) to enable reprint, should not be discounted as an electronic record on the grounds that it is not committed to durable media. Although the record may not have yet been saved to durable media, it may still be vulnerable to unauthorized alternation. FDA durable media concept was intended to exempt the keyboard buffer from audit trails. The example cited raises data integrity issues.

The most appropriate course of action at the present time would be to ensure that PLC-based systems are included in the scope of any electronic record/signature initiative. Systems should be assessed against regulatory requirements to establish if they contain GMP-related data. A detailed ERES assessment and an associated GxP focused risk analysis such as that published by the GAMP Forum is suggested.[7] Technical and procedural controls are typically available to help bring systems into compliance.

In the longer term, pharmaceutical firms must work with the regulatory bodies and PLC control system and equipment suppliers in order to achieve a consistent industry-wide approach to defining good practice on this important issue. This process has already started with the issue of the Good Electronic Records Management guideline.[8]

CONCLUSION

In an ideal world, validation would be an integral part of the development of every PLC application that has an effect on product quality. Over the past several years, awareness of validation has increased in both supplier and purchaser organizations. Validation is still being paid for either in direct monetary terms for increased documentation or indirectly by lengthening project times. Validation should be no more than documenting what happens now from planning to delivery, but the word *documentation* sends shivers down the spine of many a supplier and purchaser.

In the manufacturing environment the benefits of validation tend to go unnoticed as, by definition, the PLC-based system is working according to its design, contributing to consistent product output, meeting its quality parameters, and remaining relatively trouble free.The role of the Quality Assurance department in ensuring that the documentation produced is appropriate, required, and not excessive cannot be understated. Emphasis on the positive benefits of validation as "good engineering practice" rather than on regulatory compliance (which is often used like a stick rather than a carrot) helps to encourage effective validation. The individual pharmaceutical firms must not lose sight of the fact that they are ultimately responsible for validation of their systems.

A structured approach to validation will ensure that quality will be built into a PLC system. For the automation of a production plant, we believe that the selection of the appropriate PLC platform and the use of in-house expertise to develop the control application gives us the confidence that the software has been constructed and tested in the appropriate way and that changing production demands can be addressed efficiently.

ACKNOWLEDGMENTS

This case study is based on original work by Richard Mitchell and John Gough of Pfizer Limited. The authors gratefully acknowledge the contributions of Paul Benson (Pfizer), Keith Brown (Pfizer), Hugh Langston (Pfizer), and Eric Wells (Pfizer) in the preparation of this revision.

REFERENCES

1. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (www.ispe.org).

2. GAMP Forum (2003), *Validation of Process Control Systems, Good Practice Guide*, International Society of Pharmaceutical Engineering (www.ispe.org).
3. Rules and Guidance for Pharmaceutical Manufacturers and Distributors (2002), The Stationery Office, London.
4. FDA (1997), *Electronic Records; Electronic Signatures*, U.S. Code of Federal Regulations, Title 21: Part 11, Food and Drug Administration, Rockville, MD.
5. FDA (2003), *Part 11 Electronic Records, Electronic Signatures — Scope and Application*, Guidance for Industry (www.fda.gov).
6. PIC/S (2003), *Good Practices for Computerised Systems in Regulated "GxP" Environments* (PI 011-01), Pharmaceutical Inspection Convention, Geneva, September.
7. ISPE/PDA (2001), Complying with 21 CFR Part 11, *Electronic Records and Electronic Signatures, Good Practice and Compliance for Electronic Records and Signatures* (www.ispe.org).
8. ISPE/PDA (2002), *Good Electronic Record Management, Good Practice and Compliance for Electronic Records and Signatures* (www.ispe.org).

25 Case Study 7: Industrial Personal Computers

Owen Salvage, ABB Life Sciences
Joan Evans, ABB Life Sciences

CONTENTS

Information Technology, including computer systems and network infrastructure, is the primary way information is distributed within an organization. Distributing information to where it is needed for decision-making is one of the most effective means of achieving efficiencies within manufacturing operations. Information capture at the point of data generation reduces errors in transcription and delays in data availability.

The use of Industrial Personal Computers (IPCs) is an important means by which information and electronic functions are presented to personnel working in the more hostile environments in the pharmaceutical manufacturing environment. These environments include:

- Clean room facilities
- Packaging facilities
- Warehouses, dispensing and materials handling facilities
- Bulk pharmaceutical manufacturing plant

IPC technology is becoming faster, more reliable, and more widespread, and is to be found not just in the manufacturing arena but also attached to a wide range of analysis and control equipment.

IPCs must contend with a range of conditions not normally encountered with the more common usage of PC technology, i.e., in the office, computer room, or process control room environments. IPCs must deal with conditions such as:

- Sterilization procedures in a clean room
- High temperatures, humidity, and long running times
- Dust build-up and chemical attack
- Vibration and mechanical shock
- Power disruptions and Electron-Magnetic Interference (EMI)

IPC technology can be used in a number of ways, for example, to present information on the shop floor enabling decisions to be made at the point where materials are dispensed and to record the event at the point where goods are received or instructions are to be presented for guidance during drug manufacture.

In other circumstances IPCs can be used to control intelligent instrumentation, analysis equipment, or small-scale production units. The technology can be found as a ruggedized PC, a Windows terminal, panel-mounted, rack-mounted, or hand-held portable device, or as a portable terminal with radio frequency communications. The term IPC here is treated as covering all these types of equipment.

The opportunities for this form of computer technology are increasing within the changing regulatory environment, encouraging more innovative uses of technology found in other industries and the use of Process Analytical Technologies (PAT), which offer potential supply chain efficiencies.

REGULATORY REQUIREMENTS

As with all computer technology, the scope of regulatory control covers IPCs if the technology is involved in the handling of data and functions affecting the manufacture of drugs. IPCs by their nature are used to monitor and support manufacturing and supply chain operations. They would tend to impact product quality and be subject to regulation.

The impact of validation depends on the type of IPC technology and the duty the IPC is performing. A GxP Assessment can be conducted to identify the potential impact on product quality.

Validation of IPCs

Listing the forms of IPC technology, Table 25.1 identifies the main features to be considered when adopting an appropriate validation strategy. Windows terminal's emulation forms of IPC are simple

TABLE 25.1
Features of IPC Technology

IPC Technology	Hardware	Software
PC (ruggedized or otherwise)	Form of construction Resistance to environment Mechanical strength	Operating system resident Local applications Client–server applications Terminal emulation
Windows Terminal	Form of construction Resistance to environment Mechanical strength	Terminal emulation
Rack Mounted	Design for cabinet/rack installation Control room or field computer room Heat dissipation	Operating system resident Local applications Client–server applications Terminal emulation
Panel Mounted	Clean down Sterilization resilience Environmental operating conditions	Operating system resident Local applications Client–server applications Terminal emulation
Hand-Held Portable Device	Mechanical strength Battery life Safety in hazardous areas	Local applications Periodic data connection to network
Portable Terminal with RF Communications	Mechanical strength Battery life Safety in hazardous areas	Terminal emulation
Touch Screens	Form of construction Resistance to environment Mechanical strength	Graphics and operation

devices with no running applications and provide access to server based applications. The logic is confined to firmware required to handle monitor, keyboard, and network communications with a server.

When installed with a conventional processor, memory, and disk space, IPCs are capable of running software applications that may be standard applications (COTS) or customized applications.

This IPC technology offers every IT department or automation manager the possibility to write and install in-house Visual Basic, C++, Delphi, or other coding language–based applications.

Hand-held portable devices require applications to be written specifically for the duty and require synchronization technology to interface with server-based systems. From the generic type of IPC technology, decisions can be made early in a project to determine the validation strategy. These are summarized in Table 25.2.

VALIDATION LIFE CYCLE

Following an established validation life cycle, such as the V-Model promoted by ISPE, is a proven method of building Quality Assurance into software projects. Good project management and validation practices are close cousins.

Validation of an IPC is carried out under the authority of a Validation Plan (VP) and typically follows a V-Model approach (Figure 25.1) or a closely related variant. The phases of the validation life cycle for an IPC and the sequence of activities are listed in Table 25.3.

TABLE 25.2
Validation Strategies for IPC Technologies

IPC Technology	Validation Strategy
Conventional Processor	Life cycle appropriate for COTS or customized application
Terminal Emulation (Windows Terminal)	Configuration and qualification for hardware
Hand-Held Device	Life cycle for customized application

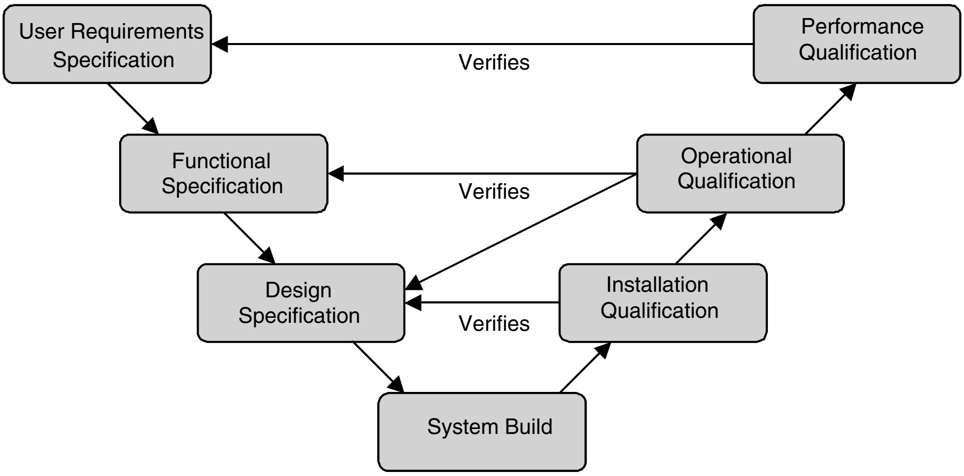

User Requirements Specification
Verifies
Performance Qualification
Functional Specification
Verifies
Operational Qualification
Design Specification
Verifies
Installation Qualification
System Build

FIGURE 25.1 Simplified V-Model.

TABLE 25.3
IPC Validation Life-Cycle Stages and Prime Responsibilities

IPC Validation Life-Cycle Stage	Prime Responsibility
Validation Plan	Manufacturer
User Requirement Specification	Manufacturer
Supplier Audit	Manufacturer
Risk Assessment	Manufacturer
Requirements Traceability	Supplier
Supplier Quality Plan	Supplier
Functional Design Specification	Supplier
Hardware and Software Design	Supplier
System Development	Supplier
Supplier Testing (FAT/SAT)	Supplier
Installation Qualification (IQ)	Supplier
Operational Qualification (OQ)	Manufacturer/Supplier
Performance Qualification (PQ)	Manufacturer/Supplier
Validation Reporting	Manufacturer
Ongoing Support	Manufacturer
System Retirement	Manufacturer

Validation Plan

The VP defines the scope, organization, and timetable of the validation program. Supplier organizations may not be known at this stage and may be generically identified within the VP.

Change control procedures to be followed during the project are defined and any necessary standards or standard operating procedures (SOPs) are referenced in the VP. The VP outlines the general and specific acceptance criteria for validation and establishes the roles and responsibilities (named individuals) for performing the validation. The VP also specifies the documentation requirements for defining and controlling test protocols, results recording, and reporting.

User Requirement Specification

The User Requirement Specification (URS) is a statement of the system requirements in terms of how the IPC application is to operate in the intended environment. The URS may be developed alongside the VP and should identify the following:

- Process to be controlled/supported
- Key environmental challenges for the IPC
- Power supply conditions on site
- List the number and types of signals used in the control system
- System size, including spare capacity (e.g., number of workstations)
- Requirements for data presentation, records, and reporting
- Audit trail requirements and copying of records
- Communication with external devices
- Networking requirements
- Performance requirements
- Availability
- Security
- Maintenance
- Documentation

End user involvement in the development of the URS is strongly recommended to ensure a good understanding of the IPC operating environment and functionality required from the system. Examples of reports required from the system, engineering drawings, graphics, and location drawings support the definition of requirements.

Supplier Audit

The pharmaceutical manufacturer is obliged by the regulations to ensure suppliers are suitable to conduct a project. The Supplier Audit is a means to identify the strength of the supplier's Quality Management System and level of experience in the Life Sciences sector.

A key outcome of the supplier audit is the identification of the degree of reliance a manufacturer can place on the supplier's QMS and whether activities can be delegated to the supplier without incurring risk.

Few suppliers comply with the requirements of validated systems, though many have achieved accreditation to ISO 9001: 1994, or ISO 9000-3 (TickIT) and aspire to achieve ISO 9001: 2000. Often this accreditation is intended to support the vendor in servicing the "automation" sector and not the pharmaceutical industry specifically.

A Supplier Audit is an opportunity to establish good working relationships among the likely project team. A collaborative conduct of an audit can greatly facilitate project strategy through a realistic, constructive assessment of capabilities, reducing overall project risk. An independent assessment provides objective review of the supplier capabilities.

Risk Assessment

The validation approach, scope, and depth follows from a reasoned risk assessment[1] (naturally this must be documented), and here the FDA offers GAMP 4[2] as guidance.

The drug manufacturer must decide to what extent the data or functions the IPC supports affect product quality, safety, or record integrity. The scope and depth of validation is then determined with effort targeted to areas of high risk or areas that may remain undetected.

Requirements Traceability

Requirements must be traceable through the life cycle to confirm that all requirements are fully defined and tested. An effective method for this is the use of a matrix. All requirements, GMP and non-GMP, are identified and traced through the respective design and test documentation. A Requirements Traceability Matrix is a powerful tool to assist in the management of projects as well as serving its main purpose of demonstrating full and complete testing of requirements.

Supplier Quality Plan

The SQP identifies the organization, standards, software, tools, and timetables proposed by the supplier in the execution of the project. Any inconsistencies with the standards and timetables required by the VP must be identified and resolved to support the validation case of the IPC.

Functional Design Specification

The Functional Design Specification (FDS) translates the requirements in the URS into the technical solution proposed to fulfill the project requirements. Operational and performance criteria for the IPC are stated so as to facilitate testing of the installed system. The FDS should specify the following:

- Functional requirements of software development
- System performance (e.g., timing, memory, spare capacity)
- Communications and network protocols (e.g., Ethernet, RS232)
- Method of storing and retrieving data
- Audit trail and copying of records
- Electronic records and signatures specification
- Safety and environmental specifications of the IPC
- Fault diagnosis
- Reporting
- Interfacing
- Performance monitoring and reporting
- Maintenance requirements
- Signal conditioning, signal noise reduction, and power supplies
- Performance and availability data for the IPC
- Numbers of users and response time required
- System application software configuration and design methodology
- Descriptions and diagrams of the software and hardware architecture
- Instrumentation and peripheral devices specification
- Start up and shut down procedures
- New or novel technology used
- Security and access control methods
- Screen layouts

Standardization of hardware and software reduces the validation effort and the risk to the project. The choice of IPC has an impact on regulatory compliance. For example, bar code scanning

TABLE 25.4
Hardware Design Checklist

Signal Types	Stepper motor, analog, and digital inputs and outputs
Earthing	Instrument and factory earths
	Communications earthing
Power Supplies	Redundant, supply filters, fed from >1 transformer, alarms
	Uninterruptible power supply (UPS) support time and maintenance bypass
	Scope of UPS support, e.g., IPC and measurements
Measurement Instruments	Capable of sterilization
	Accurate, accessible, maintainable
Cable Routes	Secure route in factory
	Sealed entry into clean rooms
	Signal marshaling arrangements
Hardware Failure Modes	State of signal outputs on power fail and power up
	Communications failure
Signal Isolation	Opto or galvanic isolation
	Barriers for hazardous areas
IPC Location	User interface location and ergonomics of the IPC
	Purging of the equipment, size considerations
	Cooling fans and air filtration
	Sealed equipment, IP65
	Mechanical vibration and shock
Input/Output Card Channel Allocation	Segregate critical channels to avoid common failures of a single card
Communications	Communications channels and interfaces
	IPC port allocation
	Network capacity and performance

can limit the options regarding password choices which tends to default to the same digits scanned; touch screens require careful graphic design to ensure typing errors are minimized.

With prototyping, the screen layout and "look and feel" of the screens can be quickly built to confirm that the system is fulfilling expectations. Screen input, range checks, and display requirements can also be clarified along with the database definition.

Hardware Design

The hardware design for the IPC identifies the components and connections required to support the system functionality. Table 25.4 shows a checklist of items to consider in developing the hardware design.

The HDS includes lists of equipment, instruments, cables, and cable identification, label schedules and instrument loop drawings of the signal installations. General arrangement drawings identify the location of the IPC displays, keyboards, cable routes, and marshaling cabinets.

Instrument certification and calibration certificates are included along with the stated environmental conditions that the IPC can tolerate. For small-scale IPC systems, the HDS can be included in the FDS. Figure 25.2 shows a typical architecture of a networked IPC system.

Software Design

The Software Design of the project covers system and application software required to fulfill the project. System software models depend on the choice of hardware. Table 25.5 identifies a range of options for software required for implementing IPC applications.

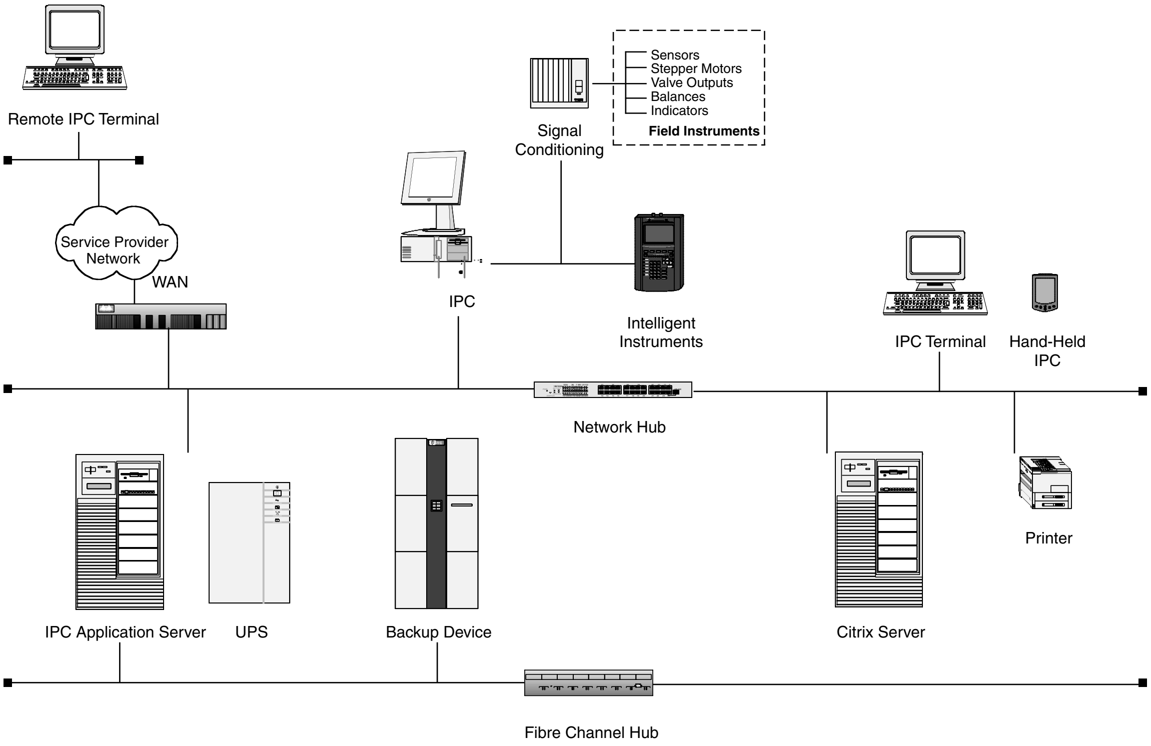


FIGURE 25.2 Typical Architecture of a Networked IPC System.

TABLE 25.5
System Software Models

	IPC Terminal Emulation	Client–Server Architecture	Windows Terminal (Citrix)	Hand-Held IPC
User Interface IPC Software	Terminal Emulation software, e.g., WRQ	Terminal Emulation software, e.g., WRQ Data tables Workflow Configuration	Citrix Client	Custom Application Synchronization
	Operating System, e.g., Windows 2000, XP, Linux		Operating System, e.g., Windows CE	
Server Software	Application-Specific Configuration			
	Standard Application Configuration			
	Database, e.g., Oracle, DB2, MS-SQL			
	Operating System, e.g., Windows 2000, XP, Unix, Proprietary control system			

A typical operating system is Windows 2000 or XP, although Linux is increasing in use. The operating system supports other software components required for the application to operate including a database, e.g., Oracle, and a front-end application to manage the operator interface, e.g., written in VB or Access. Other code, for example code to control instrumentation, motors or drives, may be custom written or standard code modules.

Application related configuration and parameters are to be specified which, together with application specific programming and when integrated, provides the full functionality required by the FDS. Software development tools with the potential to change software should not be installed on the production system. The Software Design should include the following:

- Software modules and interrelationships
- Configuration parameters
- Alarms and operator messages including out of range data input messages
- Calculations
- HMI graphic design and hierarchy
- Software module design specifications
- Input/Output database structure design and signal parameters
- Communications drivers design and interface with the database
- Signal diagnostics and maintenance
- Control functions, interlocks, and permissives
- Hierarchy of module interactions
- Storage of batch record data and reports
- Security including virus protection and password requirements
- System maintenance and administration
- Data key-in checks, e.g., range checks, "Are you sure?" and validation of inputs

Design Review

The Design Review (DR) checks the design documentation, describes the IPC system and its operation comprehensively and accurately, and conforms to appropriate standards. Engineering drawings, operating and maintenance procedures, parts lists, and system descriptions are included in the scope of the DR.

The DR offers the opportunity to ensure specified components are compatible with the target environment and meet electro-magnetic compatibility (EMC) and other regulations. The DR also confirms supplier testing has been conducted adequately.

System Development and Construction

Implementation of the software design commences once the design is finalized. Sandpit (prototyping of application functionality) and development facilities are required to support system definition and development. A validation environment to perform qualification testing and a live environment completes the system development and operational architecture.

Transfer of software modules, configuration, graphics, data, and other software objects needs to be managed and controlled formally when building the validation and live environment. Control is also required over the development environment to avoid rework and inefficient software development.

The software development facilities must approximate the final installation configuration. This is to ensure software developed during the project is capable of execution on the hardware platform used for the IPC installed on site.

System Development

IPC applications benefit from the existence of inexpensive software tools that greatly increase the speed of software production. The tools currently available allow flexibility of graphic design, use of standard programming blocks, and testing routines.

Each module should be uniquely identified with an author, version number, data, and change history and extensive comments within the code to assist debugging and maintenance. The following phases apply to the programming element of the project:

- Graphic development
- Module development
- Database development
- Unit testing (custom code)
- System integration

The risk assessment would identify the need for a configuration or source code review against critical code modules.

Construction

Construction of the IPC system includes the building of cabinets to house the IPC hardware, wiring marshaling cabinets and junction boxes, and cable and instrument installation. Elements of IPC construction include:

- Marshaling cabinet build
- Cable installation on-site from instrumentation to the IPC
- Clean room integrity considerations
- Communications interfaces
- Handover documentation (e.g., as-built drawings, electrical safety checks)

Supplier Testing

Suppliers conduct factory and site acceptance testing (FAT, SAT). Testing carried out at the factory includes hardware and software tests. Successful completion of the tests allows shipment of the IPC system to site. Tests include:

- Hardware tests: earth continuity, high voltage tests
- Radio Frequency Interference (RFI) immunity
- Calibration certificates for instruments
- Loop testing for all inputs, outputs, and spare channels
- Graphics checks
- Source code checks
- Calculation verification
- Power up/power down testing
- Heat soak tests
- Backup tests
- Integration tests
- Functional checks
- Communication interface tests
- Security testing

Formal qualification is conducted against the GxP aspects of the IPC application, identified by the Risk Assessment. The SAT scope covers the installation and commissioning of the system.

Non-GxP aspects of the system are important to the project from a business perspective. These elements of the system would normally be tested in a similar manner to GxP aspects to demonstrate the consistent application of Quality Assurance principles across the full functionality of the system and in the execution of the project.

Installation Qualification (IQ)

IQ consists of documented checks that all equipment, parts, services and documents have been supplied and installed as designed. Checks are carried out to confirm the correct software components are installed and configured according to specification.

The extent of IPC testing depends on the severity of the target environment. A range of tests for the industrial environment may be designed to simulate closely the conditions the IPC will be expected to withstand. These tests include the following:

- Loop testing of sensors. Testing the input up to the screen displays (diagnostic and operating displays).
- Visual inspection of signal shielding and earthing arrangements to eliminate noisy signals. Continuity checks on earth bonding.
- Inspection of component seals in sterile conditions. Confirmation of positive pressure within purged cabinets.

Once the IQ is completed, a report summarizes the findings and allows the next stage of qualification to commence.

Operational Qualification (OQ)

OQ aims to demonstrate that all critical functions of the equipment and software operate as designed. Functional testing represents the major activity of OQ to test all functions identified in the Functional Design Specification. A controlled validation environment is required in order to conduct testing.

Interface tests and integration test scripts are within the OQ scope for an IPC to confirm the system is ready to be subject to Performance Qualification. All test data and test software required to run the OQ testing should be confirmed to have been removed. Careful sequencing of the tests can greatly improve the efficiency of testing. A report summarizes the finding of OQ testing.

Training

Operating and maintenance personnel should have documented training evidence in how to use, maintain, and diagnose faults with the IPC system. System use and administration SOPs are the basis for the developing training materials used during the training event.

Training is best delivered on equipment similar to the intended installation, close to the cutover time, with support during the first weeks of operation, depending on the complexity of the system. Specific training in the GxP aspects of the IPC system is also to be included. Training plans and evidence of training delivery and user certification demonstrate the competence of users to operate the IPC system.

Performance Qualification (PQ)

In the case of an IPC, PQ confirms that all processes supported by the IPC functions operate correctly and deliver the final product. For example, material dispensing systems often require IPCs to support weighing operations within a sterile environment. In this case, PQ can consist of a number of successful weighing operations, documented and compared with the weighing results conducted using a calibrated balance.

Certificates of analysis confirming the final product resulting from these weighings can also be retained as further inferred evidence of the quality of the weighing IPC system. Completing the PQ allows the completion of the Requirements Traceability Matrix. Full traceability from specification through testing should be demonstrated at this point.

Data Migration and Cutover Management

Going live with the system requires careful management and close support to achieve full audit readiness and business benefit from the day the system is operational. For replacement or upgrade projects, cutover occurs following completion of PQ and confirms that all elements of the life cycle are in place before the system is allowed to go into production.

Data migration routines shall be qualified and test protocols completed with a data migration report approved prior to execution of the Cutover Qualification protocol. Figure 25.3 outlines the relationship between legacy and upgrade project life cycles. The Cutover exercise is designed to

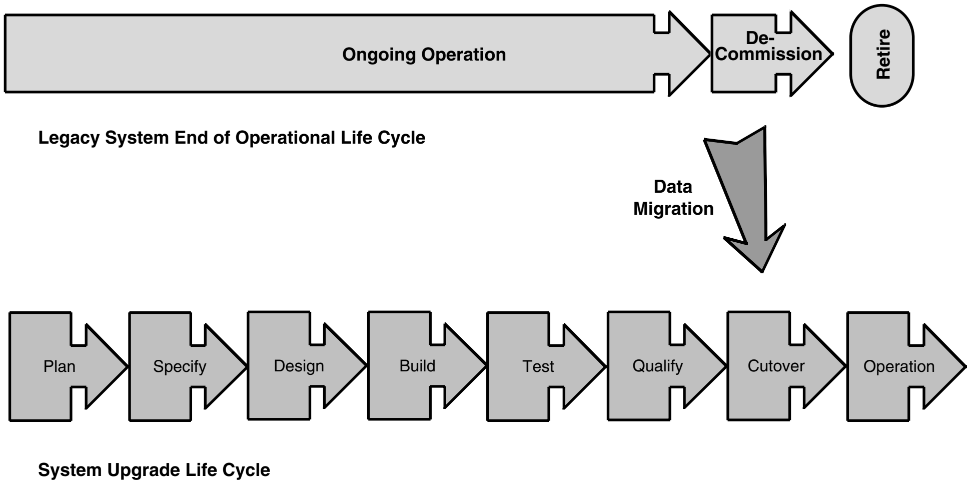


FIGURE 25.3 Relationship between Legacy and Upgrade Project Life Cycles.

minimize impact on business operations. As part of the cutover planning, rollback contingency plans shall be developed and tested prior to initiating the cutover activity.

Validation Reporting

The Validation Summary Report (VSR) concludes the validation activities, detailing the deliverables and identifying any deviations from the Validation Plan. At this point a release notice can be issued and the project closed with activities such as document indexing and storage.

Ongoing Support

IPC based applications require post installation support to ensure the system is used effectively and the validated status of the system is maintained. The system must be governed by the Quality Management System in operation at the system's location, i.e., change control, IT management procedures, and security procedures governing the granting and removal of system access.

Table 25.6 lists the Standard Operating Procedures required to manage the ongoing operation of the system. Many of the procedures would normally be in place but each would need to be assessed in relation to the new system.

TABLE 25.6
SOPs Supporting an IPC Installation

Security Access and Control	• Network security administration • System security administration • Physical security administration • User account administration	• Virus protection • IPC application security and password administration • Unauthorized access monitoring and reporting
IPC System Administration	• Data Backup Restore and Archiving • Data retention period • Data conventions and standards • Contingency, Disaster, and Recovery • Remote Consulting Access • Help Desk and Issues Management • Project document management • Problem recording and management • System Maintenance and Upgrade Management • Computer Room Environment • Configuration and source code review	• Performance monitoring and reporting • Media rotation • Live environment software change management • Workstation and server management (IPC GUI deployment) • Good Programming Practice • IPC Operating System User Administration General Guidelines
Project Support SOPs	• Periodic review • Project Risk Assessment • Master Validation Planning • Supplier Qualification • User Requirements Specification • Functional Specification • GXP assessment • Project change control • Configuration management • Requirements Traceability Matrix Maintenance • Business Process Procedures • Configuration • Project testing	• Project Integration Testing Guidelines • Installation, Operational, and Performance Qualification • Validation reporting • Design Review • Data Migration • Cutover Management • Release Notice • System Retirement • Preparation of a contingency plan • Hardware specification • Software design specification

Retirement

Legacy systems are to be decommissioned in a controlled manner according to a system retirement plan, archiving and storaging of data and documentation. The following is typically produced for retirement:

- System Retirement Plan
- Documented configuration of the legacy system
- A data retrieval procedure with a record retention and destruction schedule
- An index of system-related documentation with the appropriate record retention and destruction schedule
- A Final Summary Report approved for the system retirement
- A Retirement/Decommissioning Notice distributed to system owners affected

Supplier Issues

Few companies offering IPC equipment also provide validation services. The contributions of suppliers, systems owners, technical departments, and validation and quality units are all important in achieving a fully validated, operational system.

It is the responsibility of the manufacturer's Validation and Technical groups to manage the validation exercise on behalf of the client organization. The minimum responsibility for these groups is to produce the VP, URS, PQ, and VSR.

Technical groups can specify the requirements for the IPC system and select a suitable supplier. Coordination with the Quality Unit is recommended during a Supplier Audit to assess the degree of reliance a pharmaceutical manufacturer can place on the supplier's QMS.

Once the supplier is selected, collaboration with the pharmaceutical manufacturer's technical and validation groups is essential to the delivery of a well-engineered, validated system. Close cooperation is required to ensure the IPC functions match the URS and the demands of validation are accommodated within the project process. Following cutover, the validated status of the IPC system must be maintained through change control by the technical, IT, and validation groups.

SUPPORT ISSUES

Maintaining the Validated Status of the IPC

The ease with which IPCs may be implemented implies that change is also straightforward. Removing the software development tool is one method of reducing the likelihood of change and should be confirmed in the live environment prior to OQ.

Software changes are carried out on a separate development system and are subject to the normal validation testing and documentation updates to the IPC application. Changes to hardware are assessed and tested for compatibility with the environment and the system specification is updated to reflect the current installation.

Each change must be assessed with regard to the effect on Product Quality, Safety, and Record Integrity. Major changes warrant a Risk Assessment, and where necessary, detailed analysis of the impact of change against the predicate rules.

More structured means of change control is available using technology such as Citrix Metaframe, reducing virus risk by removing floppy disks, centralizing software administration, and utilizing security features such as 128-bit encryption. Secure gateway products enable IPC technology to be deployed effectively across the Web, with the potential to be installed on supplier networks without configuration changes to firewalls or proxy servers.

If possible, install the IPC application to boot up into the application automatically. Also, avoid installing software other than that required for the IPC application.

Upgrading IPC Software

Implicit with IPC systems is the rate of change of the available hardware and software. Upgrade paths must be controlled to ensure the IPC is maintainable into the future.

Currently, PC hardware has a life cycle of less than 2 years. Operating software typically has a support life of 5 years, then a further 2 years of reduced support before the operating system is no longer supported by the supplier. Management of the IPC upgrade protects the system investment from threats to the system's validated status, serious maintenance problems, and potentially costly fast-track projects.

Data Aging

IPC-generated data and records subject to regulation by 21 CFR Part 11 are required to be retained for a defined retention period, typically for a number of years beyond the expiry date of the product.[3] The retention period is to be determined by reference to predicate rules[4] and a risk-based assessment of the value of the record over time.

Data archiving limits the growth of databases allowing database queries, report generation, data writes, and amendments to be completed within the performance requirements for ongoing operation.

The importance of archiving depends on the amount of data stored, record retention period, and technical considerations of the archive media. A range of archive technologies is available including paper, microfiche, and electronic means.

An IPC for a medium-sized dispensing system would generate low levels of data sufficient for a number of year's data to be stored comfortably on a mid-sized server and to be available on line.

Tested SOPs are required to govern the archive and data recovery procedure.

ELECTRONIC RECORDS AND ELECTRONIC SIGNATURES

Regulations governing Electronic Records and Electronic Signatures apply to IPC installations. If electronic signatures are implemented with the IPC application, this functionality needs to be validated; personnel trained in its use should understand their accountabilities regarding the use of the electronic signature.

Through a Risk Assessment and review of the business processes supported by the IPC, the records falling under the predicate rules[3–5] are identified.

Those regulated records required for normal operation require an audit trail and the means to copy the records to an appropriate portable format.

IPC Security

IPCs rely heavily on the security provided by the supporting computer network infrastructure and need to comply with the regulations.[5] Firewalls, daily virus updates, and operating system upgrades in response to security notification from vendors are typical methods of responding to security needs.

A Risk Assessment directed toward IPC and network security can identify a range of technical and management controls with the aim of preventing, detecting, and recovering from security challenges.[6]

Networked IPC systems rely on a communications infrastructure that is subject to system-wide threats either from externally introduced viruses, compatibility with other applications, or unauthorized network access, e.g., via a vendor support link.

Security can be enhanced by automatically starting up the application when the IPC is powered on, preventing general access to the IPC operating system. Exiting the application shuts down the IPC and security is further extended by a no-reboot option on system failure. In applications of high security requirements, encryption, accountability measures (DAC, MAC), or biometric access control systems may be considered.[6,7]

Procedural controls captured in SOPs, system administration practices, and management audit ensure day-to-day IT operations in support of the IPC and mitigate the identified risks.

The Future Is Bright

IPC systems continue to offer advantages to pharmaceutical manufacturers in providing access to data on the shop floor, hazardous areas, and other hostile environments. IPCs are flexible and allow manufacturers the option to implement a central control room based system or a distributed plant based system. Both options are in use.

IPCs are scalable and allow modular expansion based on PC technology. Rapid Application Development (RAD) is even more relevant with the range and sophistication of development tools available, making developments cheaper and return on investment more attractive.

Equipment manufacturers also see the benefit in IPC technology and have invested in developing products suitable for difficult environments, i.e., Zone 1 or 2, or Safe Areas. Up to IP65 protection can be provided as part of an industrial IT solution, which includes peripheral devices such as keyboards, mouse, trackball, and displays, along with communications options such as RS232. The cost and speed of installation has also been challenged with improved signal conditioning cards utilizing twisted pair cables rather than fiber optics and the elimination of flameproof housing requiring compressed air supplies.

New pocket portals, using WAP or Blue Tooth technology, is also becoming an attractive option. However, a cautionary note needs to be registered with the heightened importance of system security and the risks of transmitting data over open systems. Biometrics and encryption are potential solutions. Also, audit trails need to be consistent across the range of software models adopted, especially client-server architectures and hand-held IPCs.

Networked applications provide flexibility to distribute the IPCs across the organization, and there are opportunities to link trading partners from shop floor to shop floor using this technology. The technology facilitates innovation in improving the supply of drug products. For example, a raw material supplier can record dispatch events using a local IPC terminal which is then readily available to the receiving organization, improving planning and warehousing operational data.

IPCs continue to provide a low-cost technology option that fit into existing IT infrastructure arrangements. With the support of equipment manufacturers and expanded opportunities for manufacturing operational improvement, exploitation of IPC technology is set to increase.

REFERENCES

1. FDA 21 CFR Part II Guidance for Industry, *Electronic Records; Electronic Signatures — Scope and Application*. Draft Guidance. February 2003.
2. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (www.ispe.org).
3. European Community Directive 91/356/EEC (Principles and guidelines of good manufacturing practice for medicinal products for human use) Annex 11.
4. FDA 21 CFR Part 210 Current Good Manufacturing Practice in Manufacturing, Processing, Packing, or Holding of Drugs; General and FDA 21 CFR Part 211, Current Good Manufacturing Practice for Finished Pharmaceuticals.
5. FDA 21 CFR Part 11, *Electronic Records; Electronic Signatures*, Final Rule, March 20, 1997.
6. Risk Management Guide for Information Technology Systems. Recommendations of the National Institute of Standards and Technology. NIST Special Publication 800-30.
7. ISO 17799/BS 7799 Information Security Management.

26 Case Study 8: Supervisory Control and Data Acquisition Systems

Adrian Kavanagh, Independent Consultant
Peter Owen, Eli Lilly

CONTENTS

Regulatory authorities expect that Supervisory Control and Data Acquisition (SCADA) systems are planned, developed, operated, and retired according to a life-cycle model that meets specific regulatory requirements. The goal of this case study is to assist professionals involved in any of the above life-cycle phases to understand the practical validation requirements for SCADA systems and how the various suppliers of these systems can satisfy these regulations. This case study is based on ANSI (References 1–10), FDA (References 11 and 12), and GAMP 4 (Reference 16) guidance. Supporting material has been taken from GAMP Forum's Special Interest Group on Validation of process control systems (see Reference 17), the German GMA (Society for Measurement and Automatic Control) and NAMUR (Standardization Association for Measurement and Control), and the North American JETT Consortium (Joint Equipment Transition Team) looking at validation of the Skid Mounted Plant.

TYPICAL ARCHITECTURE

A typical SCADA system will include the following components, depending on the nature and use of the computerized system. The system may contain many, if not all, of the following elements:

- Hardware
- Operating system
- Network system
- Database management system

The control system and instrumentation may be embedded into items of plant equipment. Some organizations split SCADA systems into computer systems and instrumentation, dividing the validation work between engineering support and plant operation. An advantage of validating the automation as a whole is that it avoids the management interface between two separate validation projects, the necessary agreement on exactly where instrumentation ends and the computer begins, and the determination of responsibilities and accountabilities for segments of the qualification.

This chapter concentrates on the SCADA system and, in particular, its software component. Plant equipment will need process validation, and field instrumentation will require basic qualification such as materials in contact, sterilization and cleaning (if necessary), calibration, and maintenance.

An automated production plant consists of:

- Plant equipment
- SCADA system
- Field instrumentation

Plant input signals are stored in a database that is accessed by sequence control and operator interface software. The operator interface software provides graphical mimics and other information to plant operators via visual consoles. The database of input signals may also initiate alarm handling software and specific operator interface feature. The sequence control and alarm handling software will send messages to either an alarm or event printer. The sequence control software will also interface with batch record software to prepare and store batch record information, which will also have visualization and print capability. Finally, the sequence control and operator interface software will output signals back to the plant to control the manufacturing process. The SCADA system software for a plant computer control system that is fully integrated is illustrated in Figure 26.1.

VALIDATION LIFE CYCLE

System validation is employed as a mechanism to establish objective evidence that a system consistently performs according to its predetermined requirements, specifications, and quality attributes. The GAMP software categories provide some guidance as to activities that should be followed as a minimum to provide the objective evidence of the system meeting performance objectives. The complexity of the computerized system should determine the extent of this effort. There are many life-cycle models published that facilitate the creation of this objective evidence. Acceptable life-cycle models will include phases for Define, Design, Develop, and Operate.

With the exception of documentary evidence, this methodology has parallels with the operational requirements of any well-run business. The documentary evidence required by the industry regulators should always be effectively planned and managed. Validation activities for computerized systems can be placed in two distinct groups:

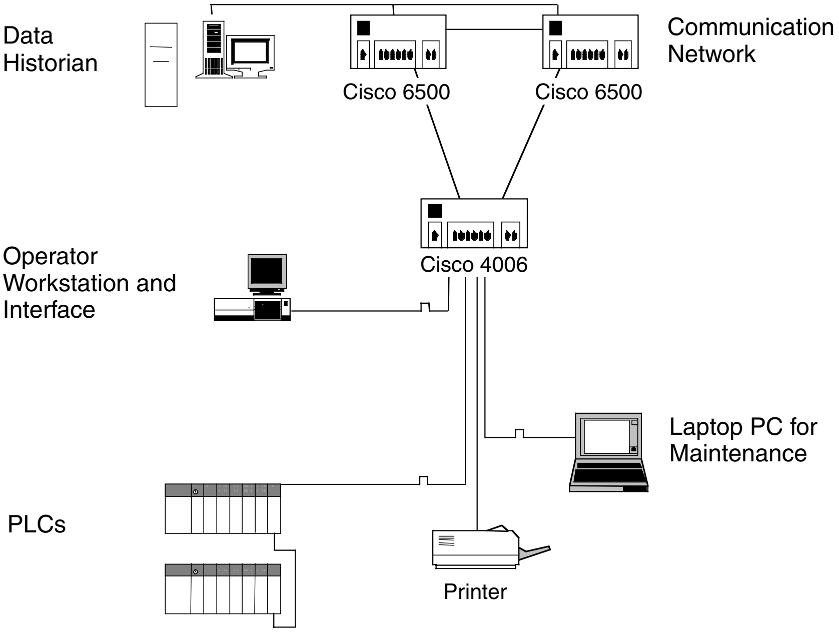


FIGURE 26.1 Typical SCADA Topology.

- Validation of applications (including the application software, interfaces to other applications, equipment, and operational procedures) for their intended use
- Qualification of the infrastructure (computers, system software, and network)

SCADA systems comprising instrumentation, hardware, and infrastructure represent a significant investment in today's manufacturing facilities. It should be noted that the assets of SCADA systems are not limited to the system hardware and software but also include the investment in the validation deliverables, both historical and living, and the data and knowledge that the system generates during its lifetime. All of these assets require appropriate management for cost-effective implementation and compliance.

Validation Planning

Validation necessitates "establishing documentary evidence which provides a high degree of assurance that a specific process will consistently produce a product meeting its predetermined specifications and quality attributes." A Validation (Master) Plan should be developed in accordance with company internal policies and procedures, including both infrastructure and applications. Standard Operating Procedures (SOPs) should be in place together with a formal and appropriate system life cycle. A typical life-cycle model as described in GAMP 4 is shown in Figure 26.2, and this may be employed for a SCADA system in order to establish the required documentary evidence. Procedures and training should be in place that describe all the relevant activities for creating, operating, and maintaining validated application and qualified infrastructure. In order to realize the benefits of validation, the following will need be taken into account:

- Quality is "built into" a system, not "tested into" a system.
- A plan for validation is established early in the System Development Life Cycle (SDLC).
- Risk assessment, mitigation rationales, and action plans are developed at an early stage.
- A process for clarifying who does what and who should do what is established.
- Documentation is produced to demonstrate that a system is validated.

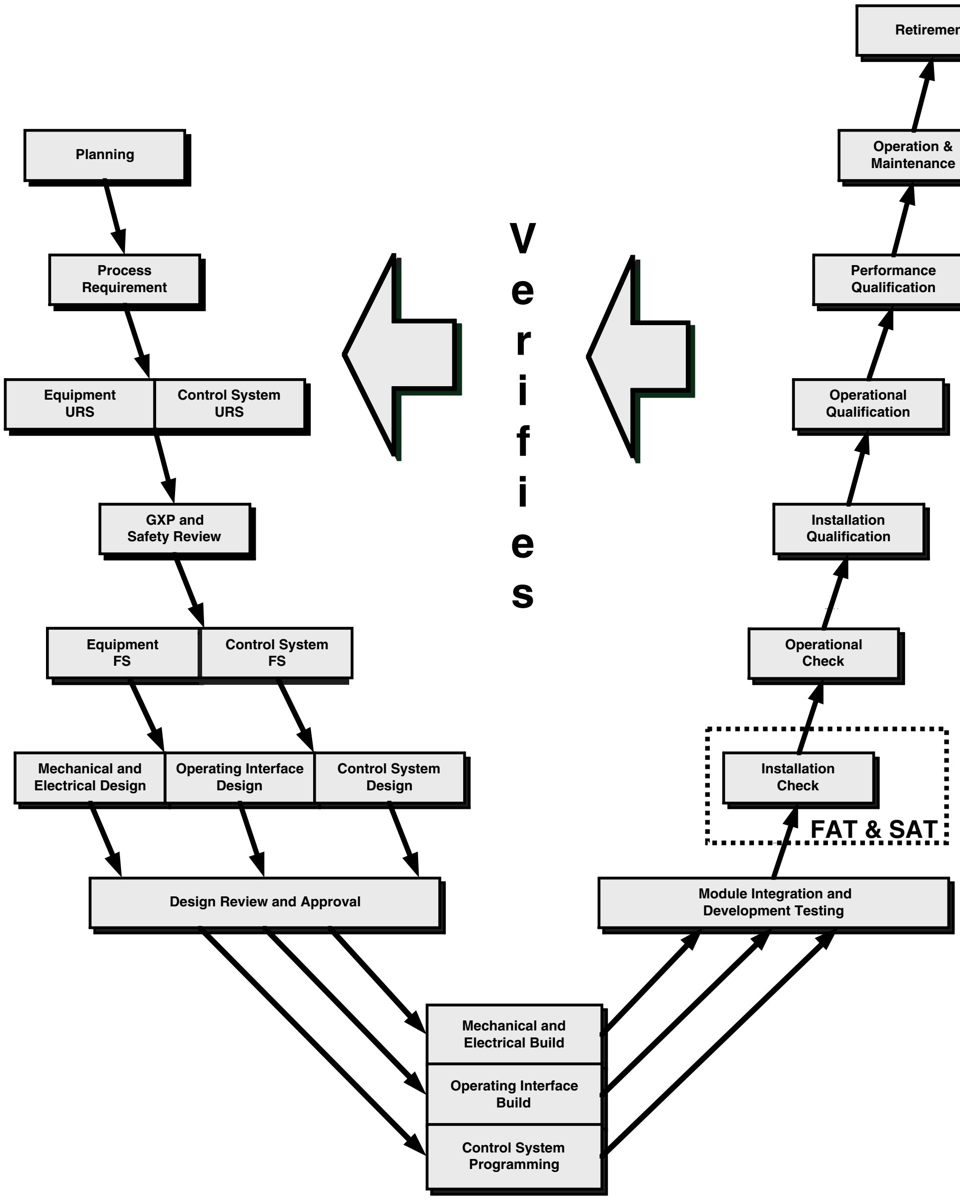

Retirement
Planning
Operation & Maintenance
Process Requirement
Performance Qualification
Equipment URS
Control System URS
Operational Qualification
verifies
GXP and Safety Review
Installation Qualification
Equipment FS
Control System FS
Operational Check
Mechanical and Electrical Design
Operating Interface Design
Control System Design
Installation Check
FAT & SAT
Design Review and Approval
Module Integration and Development Testing
Mechanical and Electrical Build
Operating Interface Build
Control System Programming

FIGURE 26.2 A Typical Life-Cycle Model Based on GAMP 4.

- Requirements (regulatory and end user) are established up front.
- A plan for ongoing maintenance and review is established.
- Good Documentation Practices are used and understood by all.

All engineering and system documentation should be reviewed and approved; the documentation should:

- Contain reviewer's name and approver's name
- Keep document reviewer and approver lists at a manageable number
- Contain signatures and date for each reviewer and approver with an explanation and meaning of the signature

TABLE 26.1
Components and Categories for Software and Hardware

Component Description	GAMP Software Category	GAMP Hardware Category
	Operator Interface	
Workstation Hardware	N/A	1
Workstation System	1	
SCADA Software	4	
Batch Engine Software	4	
SCADA Config File	5	
Batch Engine Product Recipes	5	
VBA	5	
	PLC	
PLC Hardware	N/A	1
Operating System/Firmware	1	
PLC Config File/ PLC Program	5	
	PC Laptop for Maintenance	
PC Hardware	N/A	1
PC Operating System	1	
SCADA Config Tools	3	
Network Tools	3	
PLC Programming Tools	3	
	Data Historian	
Server Hardware	N/A	1
Operating System	1	
Historian Application Software	4	
Historian DB	5	
	Communication Network	
Switches and Routers	N/A	1 and 2 for the system architecture and configuration
Firmware	2	N/A
Proprietary Software	3	

It may seem obvious to those of us who work in the industry on a day-to-day basis, but this requirement is not always obvious to vendor or supplier personnel. The omission of good document practices and the minimum document attributes described above can cause great problems at any stage of the project. Reworking of documents may cause regulatory issues where the documentation is forced out of the defined signing sequence.

The diagram in Figure 26.1 represents a typical SCADA-based control system and can be broken into five main system components:

- Operator Interface
- PLC
- Communication and network
- Data Historian
- Laptop PC for maintenance

These components can then be assessed against the GAMP software and hardware categories defined in Chapter 5. The results are shown in Table 26.1 and form the backbone of the validation strategy.

Risk Assessment

The depth and scope of validation planning should be commensurate with the significance of the functionality, impact, and criticality of the computerized system. This should be established by means of a formal risk analysis at an early stage of the validation process. Key critical compliance points to be considered may include:

- Objective evidence that a system is fit for purpose
- Access control/user management
- Electronic signature integrity, including prevention of deletion, poor transcriptions, and omission
- Authorized/unauthorized changes to electronic records, data, and documents
- Monitoring of system for performance and security violations
- Critical alarms handling
- Audit trails
- Disaster recovery including backup and retrieval
- System maintenance (including SOPs) and change control
- Training

Evidence of sufficient control of these issues should be demonstrated in the validation documentation. Compliance must be integrated using a formal methodology and an appropriate system life-cycle approach that is clearly identified in the user requirements phase for any new computerized systems. The priority for validation activities can be established by analyzing the control scheme system and subsystem inventory for the criticality, validation status, software category, and system type. This analysis aids validation planning and prioritization.

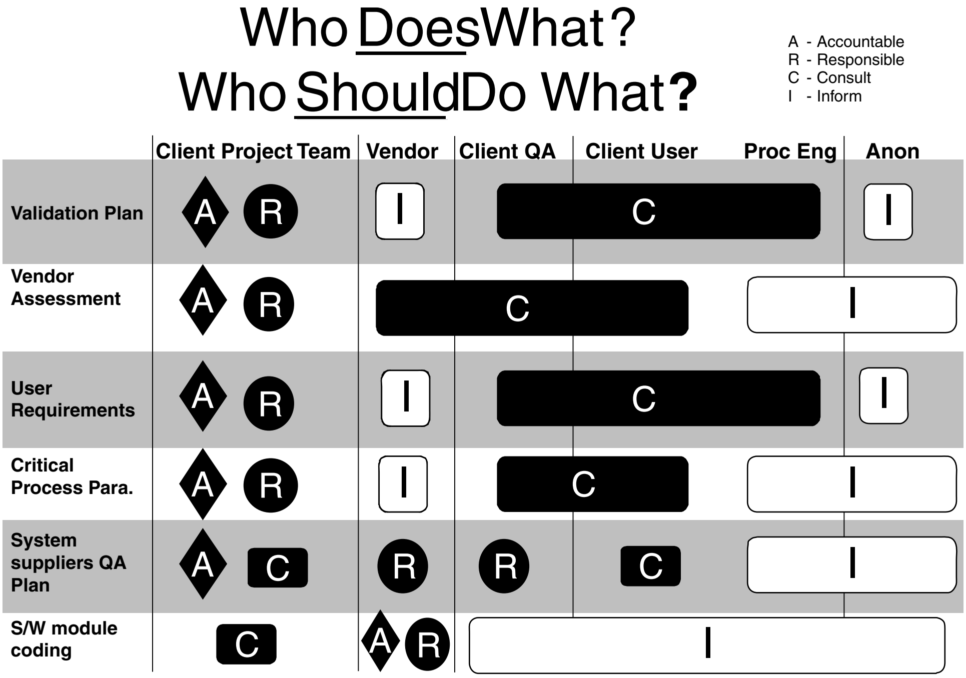
Project Plan

The key to a successful validation is ensuring that all the tasks and activities associated with the control systems validation are fully accounted for in the project plan. Consideration should be given to:

- A description of the project objectives
- The organization required to support and deliver the validation
- Definition of the validation schedule
- A Gantt chart or similar tool for managing tasks and milestones against a time line
- Measures and metrics to track progress
- Identification of quality assurance reviews
- Communication of the processes such as document transmittals and control mechanisms
- Identifying GMP quality management systems
- Availability of document describing the current computer validation situation

It is equally important to understand the roles of the various interested parties in relation to these activities and tasks — who does what and who should do what — in order to avoid excuses such as:

- It is not my job.
- I thought *they* were doing that.
- Whose job is this anyway?
- If I had known about this *earlier* …
- I cannot take the final decision.
- I have already done that.



Who Does What?
Who Should Do What?

A - Accountable
R - Responsible
C - Consult
I - Inform

<table>
<tr><th></th><th>Client Project Team</th><th>Vendor</th><th>Client QA</th><th>Client User</th><th>Proc Eng</th><th>Anon</th></tr>
<tr><td>Validation Plan</td><td>A R</td><td>I</td><td colspan="3">C</td><td>I</td></tr>
<tr><td>Vendor Assessment</td><td>A R</td><td colspan="3">C</td><td colspan="2">I</td></tr>
<tr><td>User Requirements</td><td>A R</td><td>I</td><td colspan="3">C</td><td>I</td></tr>
<tr><td>Critical Process Para.</td><td>A R</td><td>I</td><td colspan="2">C</td><td colspan="2">I</td></tr>
<tr><td>System suppliers QA Plan</td><td>A C</td><td>R</td><td>R</td><td>C</td><td colspan="2">I</td></tr>
<tr><td>S/W module coding</td><td>C</td><td>A R</td><td colspan="4">I</td></tr>
</table>

FIGURE 26.3 Validation Roles.

Responsibilities for writing, approving, and authorizing should be assigned. This can be in the form of a matrix that identifies the specific tasks for the role and the meaning of the role relating to a specific task. Four generic roles are defined:

- Responsible: gets the work done.
- Accountable: has the power of veto, "carries the can" for the outcome.
- Consulted: they must always be consulted before a decision is taken.
- Informed: they must always be informed that a decision has been taken.

Figure 26.3 provides an example of such a matrix for a number of the typical tasks and activities. Responsibilities for writing, approving, and authorizing should be assigned and documented. Good practice would be to formalize these roles in an SOP.

Risk Mitigation

As early as possible in the development of a SCADA system or scheme, an assessment should be made to determine the impact of the system or its subsystems on product quality. This assessment will provide the criteria and rationale on where to focus the validation and qualification effort. The assessment process will assist in ensuring that the appropriate resources are made available and directed to areas of the control scheme that have the potential to affect the product either directly or indirectly. The assessment should also provide a documented rationale on where to focus qualification and validation effort related to product quality functionality while still ensuring compliance for the products. It is not always self-evident that a product may be impacted directly, indirectly, or not at all. System assessments should be conducted in order to determine, document, discuss, and gain approval for the rationale for subjecting, or not subjecting, a control scheme or its sub systems to a validation or qualification process. The outcome of this process should ensure that all interested parties have actively participated in this process and given their informed approval to system assessments and subsequent validation rationales. The process should provide a method for challenging and documenting the assessments.

Consideration should be given to managing and documenting the assessment process. There may be significant numbers of systems and components, and the method of documenting the process should be determined in advance. Typically, this would be documented in the form of a standard operating procedure; this should be formally reviewed and approved by the QA unit. It is essential that the appropriate people are trained prior to commencement of this assessment process. The assessment output will have significant impact on the validation approach undertaken by the project team. Changes that occur during the project will need to be managed and tracked using a change management process. Particularly, changes to the control scheme will need to be reviewed to determine any consequential impact on the assessments and subsequent validation rationales.

The assessment activity should be identified in the control scheme project plan. The lapsed time, required resources, and effort required to complete this task will vary based on the specific control-scheme size and scope. The assessment team should involve individuals that have appropriate knowledge of the overall project activities and deliverables. Participants may include:

- User representatives (senior end user operations and management staff)
- Process experts (senior designers, scientists, and operations staff)
- Engineering representatives (client, vendors, contractors)
- Validation experts (process, equipment, and computers)

Quality Assurance

The following process may be used to assess the control scheme in relation to the impact on product quality:

Step 1: Evaluate the impact of the control scheme and its "subsystem" on the product quality. This involves the creation of a system inventory or register. The register may be in the form of a validated, controlled, and approved spreadsheet or database. Systems that are deemed to have "No Impact" on product quality would normally be dealt with by applying Good Engineering Practices.

Step 2: This step of the process would be to evaluate the criticality of the components that have a direct and an indirect impact on product quality.

The direct and indirect impact systems are now considered further in this step. This involves the creation of component lists of the control scheme, its systems, and sub systems. The direct impact on product quality can be determined by answering any of the following questions with an "affirmative."

Does the control system monitor, control equipment, or regulate a process that:

- Comes into contact with the product?
- Provides an excipient, or produces an ingredient or solvent?
- Is used in cleaning/sterilizing?
- Preserves product status?
- Is involved in product identification?
- Generates data that is evaluated to accept or reject product?
- Manipulates the process in such a way as to affect product quality without independent verification of the control system performance?

This process will categorize the systems into three groups: No Impact, Direct Impact, and Indirect Impact. Systems that are categorized as having "No Impact" would normally be dealt with by applying Good Engineering Practices.

This process should not be conducted in isolation. Typically, this is a multidiscipline activity that would involve all of the interested parties in order to ensure a complete and full evaluation of each component with the relevant knowledge and insight. The systems register could be leveraged to document this process and to track whether each component is critical or not. The critical component can be determined by answering any of the following questions with an affirmative:

- Is the component used to demonstrate compliance with the registered process?
- Does normal operation or control of the component have a direct effect on product quality?
- Will failure or alarm of the component have a direct effect on product quality and efficacy?
- Is information from this component recorded as part of the batch record, lot release data, or other GMP documentation?
- Does the component come into contact with product or product components?
- Does the component control critical process elements in such a way as to affect product quality without independent verification of the control system performance?
- Is the component used to create or preserve a critical status of a system?

The assessment should be performed at the requirements capture stage and developed more fully at later stages in the life cycle. This assessment should cover in its broadest sense GxP, safety, and environmental issues. It should also cover an evaluation and categorization of the software. SCADA systems need criticality assessment whether they stand alone or network to Industrial Personal Computers or Programmable Logic Controllers.

The critical process parameters need to be determined and then applied to the SCADA system. The manner in which the SCADA system addresses the critical process parameters needs to be evaluated and an outcome determined to ensure that the critical process parameters are accurately monitored and controlled.

The critical safety process parameters need to be determined and, in conjunction with statutory requirements, applied to the SCADA system. The manner in which the SCADA system addresses the critical safety process parameters needs to be evaluated and an outcome determined that ensures that the safety parameters are accurately monitored and controlled.

The critical environmental parameters need to be determined and then applied to the SCADA system. The manner in which the SCADA system addresses the critical environmental parameters needs to be evaluated and an outcome determined that ensures that the critical environmental parameters are accurately monitored and controlled.

System Specification

When developing URS, Functional Specifications, and design documentation, it is essential to define and classify all of the manufacturing parameters and data for an application that controls and monitors a GxP manufacturing process. Table 26.2 identifies example parameters that need to be addressed in the system specification. Definition prompts are given in Table 26.3. All parameters, data, and measurements, whether controlled, monitored, or recorded by the SCADA system, should be defined as falling into one of the following categories if they influence:

Product quality — **product critical**
Control of the process — **process critical**
Abnormal or unsafe conditions — **safety/environmental critical**

This will assist in focusing the validation effort by ensuring that the appropriate resource and attention is given to the critical parameters. Typically, the exact actual value and the ranges will not be available until later in a project. However, it should be known at what stage this information will become available, or in which documents this is expected to be contained, and these references

TABLE 26.2
Example SCADA Parameters

Manufacturing/Process Parameters	
Measured range	Set points
Ramp rates	Alarms
Trips	Processing times
Processing quantities	Process and batch identifiers
QC release codes	Passwords
SCADA System Parameters	
Highway addresses	Module addresses
Memory allocation	Dip switch settings
Software configuration	PLC rack configuration

TABLE 26.3
Prompts for System Specification

Required Definitions

Parameter Descriptor
Data Descriptor
Tag Number Descriptor
Parameter action levels and the required action on reaching defined levels
Process flow drawing reference
Units of measurement
Operating ranges for instrumentation and control equipment
High and low values within which production conditions should be maintained
Required accuracy, integrity, and/or redundancy for safety and product critical parameters
Plant and Instrumentation Drawing reference

Additional Considerations for Quality Critical Process Parameters and Data

Consequences of exceeding the Proven Acceptable Range and the action required
References to any related parameters that may affect the ranges specified or aggravate the consequences specified
Proven Acceptable Range (PAR) — all values of a given parameter that fall between proven high and low worst-case conditions (the limits of which will be tested during Operational and Performance Qualification)

should be stated in the validation plan. It may be prudent to develop a "cross-reference matrix" for traceability of parameters and data from the process development documentation through to the production system documentation.

Application Software

Application software may be both Commercial Off-The-Shelf (COTS) software as well as customized software. Application software includes the configuration of COTS packages (e.g., spreadsheets, smart operator interfaces) but not the "off-the-shelf" software package itself. Application software may require equipment to render it human-readable and may include:

- Procedural language source code
- Operator interface configured software

Suppliers developing software should follow project software conventions and standards. System software is typically supplied by the original SCADA product supplier. Generally, system software is independent of the application and may include the following:

- Operating system and file managers
- Network support and network information management software
- Diagnostic software
- Compilers, editors, and software development tools
- Database management software
- Configurable packages for batch and continuous control
- Graphical operator workstation software
- Data collection, archiving, and report generation packages

The International Society for Measurement and Control (ISA) has published the S88.02 standard titled "Batch Control Part 2: Data Structures and Language Guidelines," which provides standardized data exchange and user interface formats between competing batch systems.

It is essential that application software has been developed using a programmer's guide detailing programming standards. The use of standards often expedites project development and facilitates reusable source code and system maintenance.

Programming standards are generally language- or system-specific and address programming issues such as:

- Naming conventions for variables, symbols, programs, and files
- Annotation/commenting/documentation conventions
- Display conventions (e.g., color standards, symbologies)
- Provisions for adherence to modular design principles
- Identification of approved languages or development software
- Provisions for elimination of "dead code"
- Commonly used control strategies (e.g., alarming, interlocks)
- Process interface conventions

Modularity will facilitate testing and maintainability. It will ensure that similar systems have the same touch and feel to operators and hence provide a consistent face to the regulators. The ISA has published the S88.01 standard titled "Batch Control Part 1: Models and Terminology," which provides a framework for modular software design addressing terminology, models, and functions. For COTS software such as Microsoft Excel, using these programs is no different from writing application software.

The FDA has published Compliance Policy Guide (CPG) 7132a.15 dealing with process control application software source code issues. This guide establishes such source codes as one of the most important software validation documents. The guide specifies that drug manufacturers must:

- Consider source code (and its supporting documentation) a part of the master production and control records
- Review and approve source code prior to implementation
- Verify the absence of "dead code" in source code

Identify equipment and software to access or maintain application source code and data files. This information should be included in the business continuity plans.

Regression testing is one method to verify the integrity of source code following removal of "dead code." Dead code should be removed by the software author prior to testing and formal System Acceptance. Some systems, particularly vendor-provided packaged systems, may include configurable software that allows for enabling and disabling of functions without physically adding or removing source code. Removal of this source code could impact overall system integrity.

Establish effective measures such as labeling and version control to ensure the production system is operating with the correct software. Ensure segregation of production and development system source code. Version control should also ensure that qualification activities use the correct “baseline” software. Users should be able to produce or recreate the process control application source code associated with the production of each historical lot of the product.

System Testing

Testing requires advance planning and should cover the life-cycle phases of the system. Typical testing will occur in a hierarchy, beginning with high level testing, whereas test execution occurs in a bottom-up sequence, beginning at the lowest level. The first step is to identify the major system components that require testing:

- Field instrumentation
- Computer hardware
- Computer software
- Process equipment

Typically, this can be documented in an overall test plan. This is where the overall test approach will be documented to ensure that all components are adequately tested. There is no single right way to complete system testing. However, it is essential to understand the capabilities and experience of the testing staff, and this should be clearly specified during test protocol development. It is also important to define the tasks to be performed and the responsibilities for accomplishing these tasks. Testing is typically a cross-functional multidisciplined activity for SCADA systems. Ensure that the testing staffs are adequately trained in:

- Test procedures
- Deviations
- Testing tools
- Process equipment operations
- Area safety measures
- Test record handling

It is common to use test equipment and simulators during the testing of process control computer systems; these should be identified and responsibilities associated with these test devices clarified:

- Calibration equipment
- Process simulators
- Signal generators
- Terminals and printers

Tests may be conducted in a number of environments:

- Process simulators and development systems
- Using the production system
- Some combination of the above

The test plan includes all aspects of the control system. This will ensure that users assess the scope, content, and extent of the qualification testing and hence verify that all requirements have been tested.

An objective should be to develop an overall test plan that ensures maximum test coverage, while optimizing the test effort. A QA review should be conducted to ensure that the scope and the adequacy of the testing is complete. Typically, the following groups of individuals should be involved in this review and approval process:

- End User
- Quality Control (QC)
- Technical Services
- Engineering

A key element of the overall process computer system testing effort is the concept of traceability. Testing should be traceable back to design features and system requirements. Include traceability matrices to show complete coverage of requirements and design. Formal review activities should include review of these matrices. The GAMP 4 Guide provides direction on requirements traceability.

Acceptance criteria should be based on the Requirements and Design validation products. Do not use "pass/fail" acceptance criteria exclusively for functionality that generates a variable or analog result (e.g., 4–20 mA signals, algorithms results, and calculations). Good practice would require the recording of indicated range of values that is acceptable and the expected value, recording the observed value, and indicating "pass" or "fail" in the test result documentation.

The prerequisite requirements for testing should be clearly defined (e.g., interlocks and utility systems). It is a good idea to coordinate Factory Acceptance Testing (FAT) along with the mechanical system checkout. Consider using the same testing approach for both in-house developed systems and vendor-provided packaged systems in situations when both types of systems exist on the project.

Installation Qualification (IQ) provides documented verification that instrumentation, computer hardware, operating system software, and application software have been purchased, received, and installed according to requirements and design.

IQ testing for a SCADA systems may include:

- Receipt verification of purchased items, including software
- Installation verification against environmental conditions
- Installation verification against design specifications (e.g., DIP switch settings)
- Installation verification against electrical specifications
- Software installation verification
- Computer hardware power-up checkout
- Software diagnostics reports containing memory partitioning information, disk fragmentation summaries, directory structures, version identification, patch and driver file verification, hardware configuration reports, etc.

Following IQ, Operational Qualification (OQ) provides documented verification that the process equipment, instrumentation, computer hardware, and software operate as expected. OQ testing may include the following verifications:

- Functional testing verifies that processes can be performed correctly.
- Usability testing verifies that user interfaces and documentation are understood by users and meet the user needs.
- Display testing demonstrates visual information and screen ergonomics.
- Response testing demonstrates acceptable man/machine, network access response times, and system throughput under normal loads.
- Integrity testing demonstrates that the system is able to manipulate and control data accurately and reliably (e.g., invalid input testing).
- Security testing demonstrates that access to the system, data, and system databases are appropriately controlled.

- Interface testing confirms transfer and/or conversion of data both within the system and between systems.
- Data historian testing demonstrates the ability of the system to electronically capture process and environmental data.
- Stress/load testing (e.g., maximum number of users logged on to system).
- Regression testing verifies that modifications have had no impact on function or operation of the system.
- Calibration of both computer hardware and digital-to-analog converters, power supplies, and field instruments.

Loop checking includes both dry and wet loop checks. Dry loop checking includes verifying that power is available at the field device, verifying the field device input signal electrical continuity through the Input/Output (I/O) device to the operator interface (reversed for output devices), and verifying the "baseline" software.

Calibrations and dry loop checks of all loop devices must be successfully completed prior to beginning a wet loop check. Wet loop checks apply an actual load on individual control loops to verify proper operation under process conditions. Wet loop checking usually precedes some OQ testing.

Water batching combines multiple systems (process and utility) to simulate production conditions using water or process materials, if necessary. Water batching may be done in combination with other OQ tests. Wet loop checking and water batching often use water to simulate process loads but, if necessary, may use actual process materials.

During Performance Qualification (PQ) testing, the computer system is being exercised, but the computer system is also being tested by inference during this test phase. Automation personnel typically do not actively participate in PQ testing but instead perform support/consulting services.

Based on the evaluation of the individual test reports, indicate acceptability on the test documentation. If necessary, authorize additional tests to complete the validation. Meaningful conclusions are those that are based on a comparison of test data against specific criteria and specifications that were defined at the outset.

System Acceptance

The Validation Plan indicates the beginning of the validation project; System Acceptance indicates that a system is validated. The system life cycle moves into the Operation and Maintenance phase. The System Acceptance documents often consist of a checklist, punch lists, and a summary report (commonly known as the Validation Report).

Typically, the validation evidence for a SCADA system is created and approved through system development. Large systems are often formed of smaller subsystems and integrated during development and start-up efforts. Care should be taken to avoid using checklists as a substitute for validation deliverables or quality assurance reviews; checklists simply indicate the presence or absence of required validation products.

It is essential to document all outstanding issues and include a plan to resolve these, no matter how small. This plan should include the name of an accountable person responsible for the action item and the estimated completion date. In the system acceptance report, specific reasons for determining that the system is validated should be stated; this statement should take into account outstanding issues. Many minor issues can render a system invalidated ("death by a thousand cuts").

Documents should include written statements by responsible personnel regarding the acceptability of validation evidence. These conclusions may be documented at various stages. When reviewing validation documents, look for evidence that data are available and consistent; these should have a stated conclusion. Validation checklists, punch lists, and the final summary report are considered "historical" documents.

SYSTEM SECURITY

Identify security requirements and attributes during the requirements phase, establish access control guidelines during the design phase, challenge security features during the test phase, and routine audit security measures during the operation and maintenance phase of the system life cycle. Use procedures for maintenance of accounts and access authorization after system acceptance and as described in the operational system support procedures.

Security should encompass both development and production systems. Implement necessary security measures during the preimplementation phases of a project. For example, install key locks on field construction offices and establish access control guidelines for system developers (and other project team members, as necessary) who are using tools such as development systems and process simulators.

Align security measures for the computer system with your corporation's Information Asset Protection Policies (IAPP) and coordinate with the building and/or plant security plans. For some systems, strict compliance with specific security requirements mentioned in the IAPP may not be possible given proprietary system limitations. Examples may include unique user identification codes, specified lengths for user identification codes and passwords, or inactivity time-outs. Use a deviation change process to document and justify these situations.

GMP-Regulated Systems Security

The ability to assure that authorized, identifiable individuals perform specified actions is an important aspect of GMP compliance.

Systems such as Manufacturing Execution Systems (MES) or Electronic Batch Record Systems (EBRS) that rely upon electronic identification of individuals should ensure that "electronic identification/signatures are secure from abuse and falsification" and that "substitutes for handwritten signatures should nonetheless be as secure as conventional handwritten signatures."

Assess the risk involved when implementing any security scheme. Identify system entry points and sensitive or confidential information assets. Risks include threats from intentional acts as well as from unintentional and accidental threats. An overly elaborate scheme may be ineffective since it may discourage use or encourage circumvention of the system.

Layers of Security

Security can be thought of as a system of layers that protect the computer system that must be secured. The outer layer constitutes the physical security and the innermost layer constitutes the logical security. Figure 26.4 arranges these layers in a hierarchy structure.

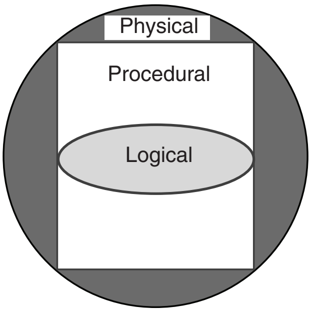


FIGURE 26.4 Hierarchy of Security Layers.

Physical Security

This includes the access control to servers, which may be located in a separate room and/or in locked panels. Generally, companies should have a policy/procedure to be used to control access to the facility, server, and control rooms.

Asset protection relating to fire protection is generally handled through fire suppression systems and manual fire extinguishers. However, storing backup media in an area of the facility should mitigate losses and be part of the archiving, backup, and restore SOPs for the system.

Physical security should, as a minimum, take account of the following considerations not only from a compliance perspective but as a good business practice, e.g., business continuity planning:

- Building and room access considerations
- Key control considerations
- Fire protection considerations
- Environmental control considerations
- Building and room access considerations
- Operations/control room access controls
- Maintaining an entry/exit log for the above
- Methods to prevent and monitor "piggybacking" room access, e.g., video surveillance of your automation system
- System servers may be located in a more restricted and secure separate room
- System servers, racks, routers, and switches should be in a restricted and secure separate room

These "headline" considerations can be expanded in more detail.

Key Control

- Policy and procedures for building key/code control process
- Keys registered and tracked
- Keys periodically changed
- Process for loss or theft of keys
- Periodic audits of the status of keys
- Process for issue and revoking keys

Fire Protection

- Fire suppression systems within automation control system areas
- Smoke and vapor detector systems
- Fireproof safe provision for storage of critical media
- Backup media stored in an alternative location

Environmental Control

- Standards and requirements for automation computer system climate control
- Standards and requirements for area dust control
- UPS systems on automation systems
- Temperature control within the installed area

Physical security addresses location, access, and protection of tangible system components. Physical security should address the following:

- I/O cabinets and panels
- Control rooms
- Spare parts inventories

- Documentation libraries
- Software media storage facilities, both off-site and local
- Programming or configuration terminals and keyboards
- Process simulators and system development tools
- Configurable devices
- Network devices (e.g., routers, bridges, and gateways)

Devices such as key locks, magnetic card readers, cipher locks, and network firewall systems (which may include both computer hardware and software) are physical security measures used to limit access to authorized individuals.

Procedural Security

Written security procedures establish the accountability, audit trails, and the separation of duties necessary to safeguard a computerized system. Refer to the operational support product for related information. Security procedures may include the following:

- User account additions, modifications, and deletions
- Password modification
- Card/key control and distribution
- Cipher lock combination changes
- Routine computer virus scanning

If keys, cards, or other devices are issued to individuals for unlocking programming devices or changing the system configuration, describe the control provisions in security procedures. Ensure that individuals will relinquish control of the devices when they are no longer authorized (such as when reassigned to another department).

The creation of access control guidelines is a user-centered activity. Access control guidelines are often developed as a matrix identifying the various classes of users, the various classes of data or information, and the access privileges (e.g., write, alter, read only) for each combination of user class and data class.

Access controls must exist not only for individuals who use or operate the system but also for those individuals who are authorized to modify programs or change system configurations.

Most companies are putting procedures in place to authorize user access to their automation systems including vendors and third parties. Audits of these accounts are necessary to maintain their accuracy and currency. Levels of access are created, including individual accounts, to limit the ability of each group to make higher levels of changes to the automation systems. Authentication for access has been primarily in the form of passwords with little effort in the area of biometrics or proximity/electronic key cards. Computer system time clocks can be restricted with synchronization to a NIST-traceable source. Most sites provide training on their automation systems and customize that training based on roles and responsibilities of the target audience. Password lengths are commonly a mix of six alphanumeric characters. A system administrator changes preset passwords. Required passwords change every 60 to 90 days. Password history and reuse of passwords can be implemented, and inactivity timeouts and password-protected screensavers exist at most sites. Virus protection is governed by policies and procedure at typically three out of four companies. Periodic assessment of automation control system vulnerabilities is not often undertaken within the industry. Account deactivation due to incorrect login attempts and user-password expiration is common policy. Many companies have a policy/procedure in place to back up their automation control system software (unencrypted) with a trend toward central backup vs. individual processor backup.

Skid-mounted automation systems have been identified as the major Part 11 outage as users have voiced concern over vendors struggling to meet compliance. Separating manufacturing systems on the LAN seemed to be more desirable through routers, Virtual Private Networks (VPNs), switches, gateways, etc. The security policies in NT, Windows 2000, and XP are being leveraged to transition to single sign-on and single point of administration for automation systems. Many within the industry implement site/building and room access systems to monitor and control anyone who has access to critical SCADA systems and site entry/exit. User access levels frequently have three or four levels of access for everyday usage: maintenance technician, operator, engineer, and administrator.

Logical Security

When developing the technological and procedural requirements for logical security, it is important to consider conducting a security risk assessment for the system. This would typically include the following risk categories:

- Human error and accidents
- Dishonest and disgruntled employee
- Outside intrusion
- Eavesdropping data tampering
- Virus attack

Logical security requirements and procedures for migrating these risks in a SCADA system may include defining access privileges, authorization, and controlling user access to the SCADA system:

- Role-based access control
- Discretionary access control
- Rosters that define what hosts can connect to SCADA systems
- Granting of temporary access privileges
- Audit-access accounts to maintain accuracy and currency
- Monitoring for unauthorized activity
- Every user and administrator should have an individual account
- System authenticates users, e.g., entering user-ID and password, electronic key card, or biometric control

Additional “headline” considerations include:

Access to the System Clock

- Restriction of access to modifying the system clock by administrators
- Automatic system clock synchronization of servers and clients to a known source

Training Requirements

- How to access the SCADA system
- Automation control system users to sign a valid security or appropriate use agreement
- Customized training to meet the needs of automation system role use
- Monitoring of the control system to ensure compliance to security procedures and policies

User-ID and Password Requirements

- Standard minimum length requirement for your automation control system passwords
- Criteria for automation system passwords formats, e.g., those not to be found in a dictionary
- Must include one or more numerals

- Is not a date or commonly expected format
- Does not identify the owner by a mix of characters
- System administrator changes all preset passwords built into automation control system software
- System passwords should be changed at periodic intervals
- Reuse of automation control system passwords prevented
- Control system user-IDs revalidated periodically
- Changing of the operating system administrator account name — one of the most overlooked security precautions
- User-ID and password are not identical

Timeouts Associated with Unattended Use

- Automatic inactivity timeouts on your automation control systems
- Emergency access requirements and processes — is there an emergency route to controlling the plant if it or personnel are endangered?

Virus protection activities defined in a policy and procedure for governing virus protection for SCADA systems include methods for checking automation system data and software integrity, e.g., standards and requirements for installing fixes/patches for known automation control system problems. Periodic assessment of automation control system vulnerabilities are to be undertaken in the light of emerging knowledge.

Automatic account deactivation policy and procedure for governing the automatic account deactivation for SCADA systems include:

- Deactivating a control system access account based on number of incorrect login attempts
- Limiting the number of login attempts for each system port or client
- Deactivating an automation system account upon expiration of user passwords

Backup media policy and procedures for governing the backup and archiving of control system data and/or programs include:

- Maintaining at least one copy of all automation system data files
- A backup of the control systems application
- Maintaining system backup logs
- Random system test restores

ELECTRONIC RECORDS AND ELECTRONIC SIGNATURES

Solutions to ER/ES requirements can be provided through implementing technological or procedural methods. It is likely that a combination of the two will provide the final solution for a SCADA system owing to available technology restrictions. Therefore, when developing the ER/ES requirements for the system, consideration should be given to both:

- Procedural Requirements and
- System Technological Requirements

Security and access privileges must cover all component systems making up the integrated solution. Issues here include passwords unique to the individual, that must not be shared among user classes. Electronic records passed between systems must be accompanied by their audit trail requirements. A centralized database may ease management of electronic records. The use of Web

TABLE 26.4
Example Extracts of an ER/ES Matrix

21 CFR Part 11 Reference	Description	System Requirement Explanation
11.10(a)	Validation of systems to ensure accuracy, reliability, and consistent intended performance	Validation is a life-cycle activity and validation documentation expectation No functional requirement to be stated
11.10(a)	Ability to discern invalid records	The control system must verify that input values are within predefined ranges The system must display an error message when the value is outside of the limits The system must detect null or error inputs from any instrument
11.10(a)	Ability to discern altered records	The system must create an audit trail that includes date/time of the altered record Operator ID of the individual that creates, modifies, or deletes an electronic record When an electronic record is modified, the system must record the data before and after the modification as part of the audit trail Users must not be able to modify the audit trail created by the system The system must use the format DD-MM-yyyy hh:mm (24 h) for the date and time Administrative functions must be recorded by the system and include name of individual performing administrative function and date/time the function was performed System must make apparent to the observer if the record being displayed has been modified

features must not compromise security or integrity of the application. GxP data such as set points, recipes, configuration parameters, etc. must be carefully managed.

One method is to develop a matrix that identifies each of the requirements referred to in 21 CFR Part 11 (see Table 26.4). The system requirements are then listed in a corresponding "System Requirement Explanation" column. Since many ER/ES requirements are procedural in nature, not all requirements will have an associated "technological" or functional system requirement. Each 21 CFR Part 11 requirement is then evaluated to determine whether or not it leads to a functional requirement. This leads to the development of the functional requirement for the system. The system requirement should describe a compliance method for each relevant 21 CFR Part 11 requirement to be achieved.

Items that need to be covered include how access is controlled, a password policy, and audit trails. At all three levels there should be continuously updated roster lists of approved users and administrators and their authorization levels. At no level should there be shared passwords up to and including administration privileges.

There needs to be proper access control procedures on three levels:

- Wide Area Network (WAN) and/or Local Area Network (LAN) level
- System/application/PC level
- Administration level

It is normal practice for the system users (i.e., process operators) to interact with the system through the user interface (panel, monitor) in order to initiate the process and respond to alarms, etc. Users will gather data from the system; critical parameters such as temperature, pressure, and time may be observed, or data may be transmitted without retention to another system/equipment such as a data historian.

Electronic records are any retained data written to durable media, e.g., on CD ROM, hard disk files, and memory cards. Data held on EPROMs may also be considered electronic records. This includes storage of set points, alarm limits, and configuration parameters. Electronic records may also include data memory cards used in embedded control systems such as steam sterilizers. These cards are used to transfer data (critical parameters, cycle times, etc.) from the control system to a main database application. Manual installation, removal, and transfer of the memory cards must be conducted in accordance with defined procedures. Reuse of cards must be carefully considered — data must not be deleted until confirmation of successful download to the database. The local operating environment and the possibility of Electro-Magnetic Interference (EMI)/Electro-Static Discharge (ESD) damage to data integrity of the memory card must also be evaluated. A record held in volatile memory, e.g., temperature profile from an Autoclave, and retained for a period (say five batches) to enable reprint, should not be discounted as an electronic record on the grounds that it is not committed to durable media. Although the record may not yet have been saved to durable media, it may still be vulnerable to unauthorized alternation. The FDA's durable media concept was intended to exempt the keyboard buffer from audit trails. The example cited raises data integrity issues. Audit trails should be secure and computer generated. Paper-based change control and configuration management controls will not suffice. Electronic signatures are any approvals made electronically, required for GxP (e.g., sequence stages of an electronic batch record).

If the data are transmitted to other system/equipment without use of internal files, this does not constitute an electronic record. Legacy systems are frequently described as "hybrid systems" where the batch/lot record is printed and approved by handwritten signature.

MANUFACTURING EXECUTION SYSTEMS

Integrated SCADA systems are sometimes referred to as Manufacturing Execution Systems (MES). Figure 26.5 provides an example topology. The complexity of communication interfaces between the elements may be simplified by using standard (open system) interfaces, but the number of interfaces often means effective design of work, and information flows are critical. The basic validation requirements remain unchanged. The propagation of change and configuration management is key to the successfully validating and maintaining systems of this type in compliance. The need to assess ripple effects from changes and regression testing mean maintenance and change control are likely to be complex.

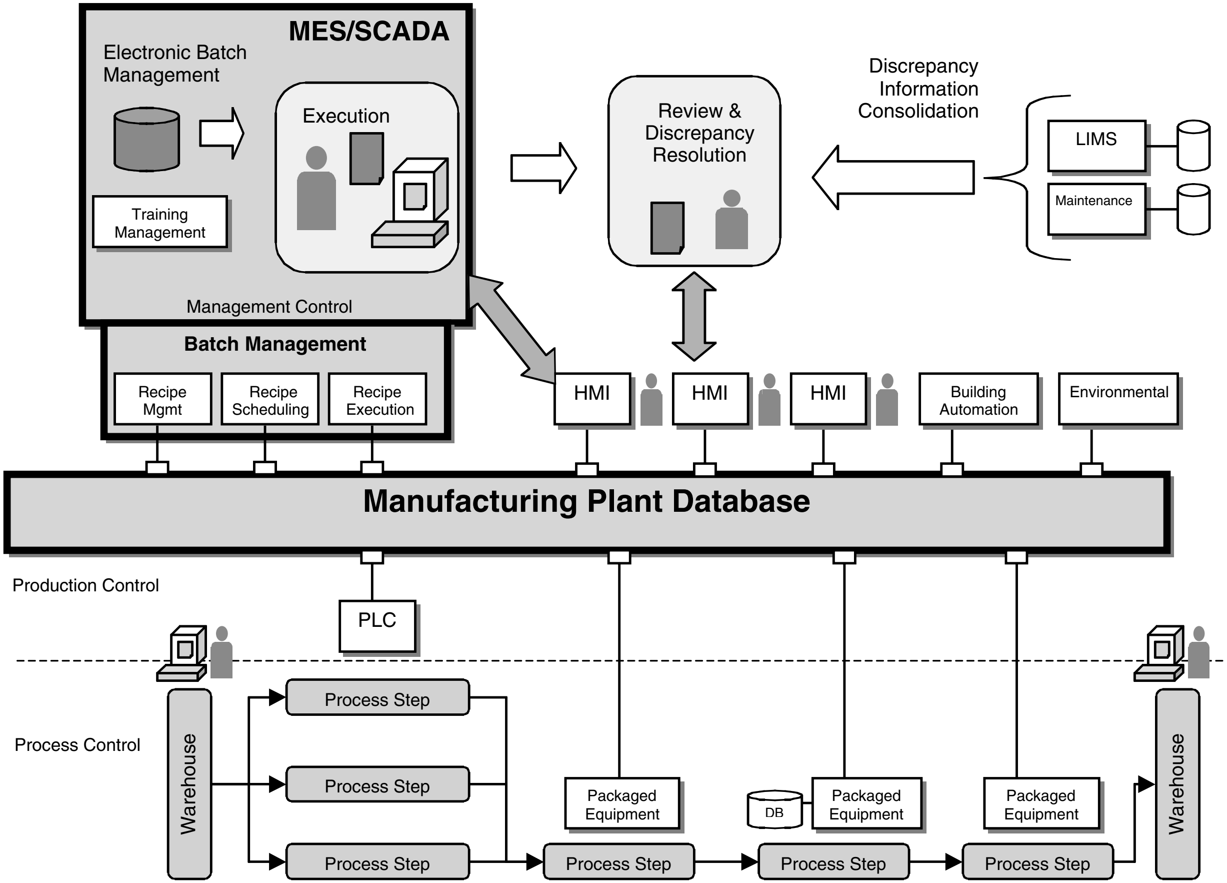


FIGURE 26.5 Integrated System.

REFERENCES

1. ANSI (1983), IEEE Standard for Software Test Documentation, ANSI/IEEE Std. No. 829-1983, The Institute of Electrical and Electronic Engineers, New York.
2. ANSI (1987), IEEE Guide to Software Configuration Management, ANSI/IEEE Std. No. 1042-1987, The Institute of Electrical and Electronic Engineers, New York.
3. ANSI (1987), IEEE Standard for Software Unit Testing, ANSI/IEEE Std. No. 1008-1987, The Institute of Electrical and Electronic Engineers, New York.
4. ANSI (1987), IEEE Standard for Software User Documentation, ANSI/IEEE Std. No. 1063-1987, The Institute of Electrical and Electronic Engineers, New York.
5. ANSI (1987), IEEE Standard for Software Verification and Validation Plans, ANSI/IEEE Std. No. 1012-1987, The Institute of Electrical and Electronic Engineers, New York.
6. ANSI (1988), IEEE Standard for Software Reviews and Audits, ANSI/IEEE Std. No. 1028-1988 (corrected 1989), The Institute of Electrical and Electronic Engineers, New York.
7. ANSI (1989), IEEE Standard for Software Quality Assurance Plans, ANSI/IEEE Std. No. 730.1-1989, The Institute of Electrical and Electronic Engineers, New York.
8. ANSI (1990), IEEE Standard for Software Configuration Management Plans, ANSI/IEEE Std. No. 828-1990, The Institute of Electrical and Electronic Engineers, New York.
9. ANSI (1993), IEEE Recommended Practice for Software Acquisition, ANSI/IEEE Std. No. 1062-1993, The Institute of Electrical and Electronic Engineers, New York.
10. ANSI (1993), IEEE Recommended Practice for Software Requirements Specifications, ANSI/IEEE Std. No. 830-1993, The Institute of Electrical and Electronic Engineers, New York.
11. FDA (1997), Current Good Manufacturing Practices for Finished Pharmaceuticals, 21 Code of Federal Regulations, Parts 210 & 211, Federal Register, U.S. Government Printing Office, Washington, D.C.
12. FDA (1997), Compliance Policy Guide No. 7132a.15, Computerized Drug Processing: Source Code for Process Control Application Programs, Food and Drug Administration, Freedom of Information Staff, 5600 Fishers Lane, Rockville, MD.
13. FDA (1997), Good Laboratory Practice Regulations, Final Rule, 21 Code of Federal Regulations, Part 58, Federal Register, U.S. Government Printing Office, Washington, D.C.
14. FDA (1997), Good Manufacturing Practices for Medical Devices: General, 21 Code of Federal Regulations, Part 820, Federal Register, U.S. Government Printing Office, Washington, D.C.
15. FDA (1997), Electronic Signatures and Electronic Records, Code of Federal Regulation Title 21, Part 11, Food and Drug Administration, Rockville, MD.
16. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (www.ispe.org).
17. GAMP Forum (2003), Validation of Process Control Systems, Good Practice Guide, International Society for Pharmaceutical Engineering.

27 Case Study 9: Distributed Control Systems

Mark Cherry, AstraZeneca

CONTENTS

This case study provides guidance on considerations when installing and validating a Distributed Control System (DCS) for a pharmaceutical batch process.

Most pharmaceutical manufacturing facilities designed and constructed in the mid- to late 1980s are likely to have taken advantage of the automation offered by the DCSs or PLCs available at the time. Such systems are often seen as being inflexible, and any recipe/reporting and data collection (alarms, trends, operator actions) are unlikely to be technically compliant with the FDA regulatory requirements on electronic records and electronic signatures.[1] Therefore this case study also considers the options for implementing a replacement DCS system on an existing facility.

OVERVIEW OF A DISTRIBUTED CONTROL SYSTEM (DCS)

There is no single, clear definition of a Distributed Control System. However, the attribute of having distributed processing capacity is a good focal point. The processing capacity is not constrained to "traditional" DCS controllers and may, for example, be executed within Programmable Logic Controllers (PLCs).

Within the pharmaceutical industry, the use of distributed control systems is mainly (although certainly not exclusively) within bulk API manufacturing plants. The scale of DCS systems can vary tremendously; systems may cover multiple manufacturing facilities or just a single process stage. In terms of input/output (I/O) count, systems range from having field device I/O in the tens of thousands down to systems of perhaps 100 I/O. A stand-alone system controlling a single process unit (e.g., a single PLC/SCADA system for a packing machine) would not normally be considered a DCS. However, multiple PLCs controlling a process stage connected to a SCADA system could be considered a DCS.

Historically, the distinction between PLC systems and DCS systems tended to be fairly well defined. PLCs were predominantly used for smaller applications, being lower in cost and often embedded within packaged equipment. DCS systems tended to be used for larger applications, the configuration being process specific (often containing much sequence control logic). More recently, the cost of DCS systems (particularly for hardware) has fallen significantly with a corresponding increase in the processing, memory, and communications capability available within PLC systems. In particular, for medium-size systems there is now little to choose between the PLC or DCS Controller options.

The main areas of functionality provided by a DCS are typically:

- Operator interface: the provision of graphical and textual information on the plant status, also providing the operator the ability to control plant devices, either directly or by automatic sequences.
- Sequence control of process operations, and recipe/batch management and tracking.
- Alarm and device interlocking (often in addition to separate hard-wired systems).
- Event and alarm recording, and historical trend recording of process variables.
- Control of analog process variables (e.g., temperature, flow, pressure, etc.).
- Interface to embedded control systems provided as part of packaged plant units such as filters, driers, centrifuges, etc.

Figure 27.1 illustrates a typical DCS architecture.

INTRODUCTION TO S88.01

Modern process control systems for batch process frequently make use of the ISA S88 standard for batch control.[2] This provides a layered, structured approach to the system architecture and configuration, and provides both a high degree of automation and flexibility within the process control system. The generic, modular nature of the architecture also provides opportunities for streamlining the validation process. Most examples of S88 implementations are based on new green field plants/installations; this case study considers the application and validation of an S88-based solution both to an existing facility (brown field application) and to a new (green field) facility.

The S88 approach to batch control provides a framework for the architecture of the system, having essentially four layers of control that operate on plant units (e.g., reactors, filters, driers) within a process cell (a collection of plant units within a facility or used for a process stage).

The control module layer is the lowest level and defines how field devices (e.g., valves, pumps, controllers, etc.) interact with the process control system. Phases are at the next layer and describe small (often generic) sequences (e.g., fill, transfer, initiate temperature control, etc.) that operate on a unit. At the next layer up the hierarchy, phases may be combined into unit operations to perform more complex functions (e.g., distillation, crystallization, etc.).

The top layer is the procedural layer and this generally defines how unit operations are combined across plant units for the overall process. A feature of S88 is the ability to generate equipment modules, essentially common arrangements of control modules to provide a specific function (e.g., skid-mounted temperature control units for reactors or valve/pump arrangements for transfer routes).

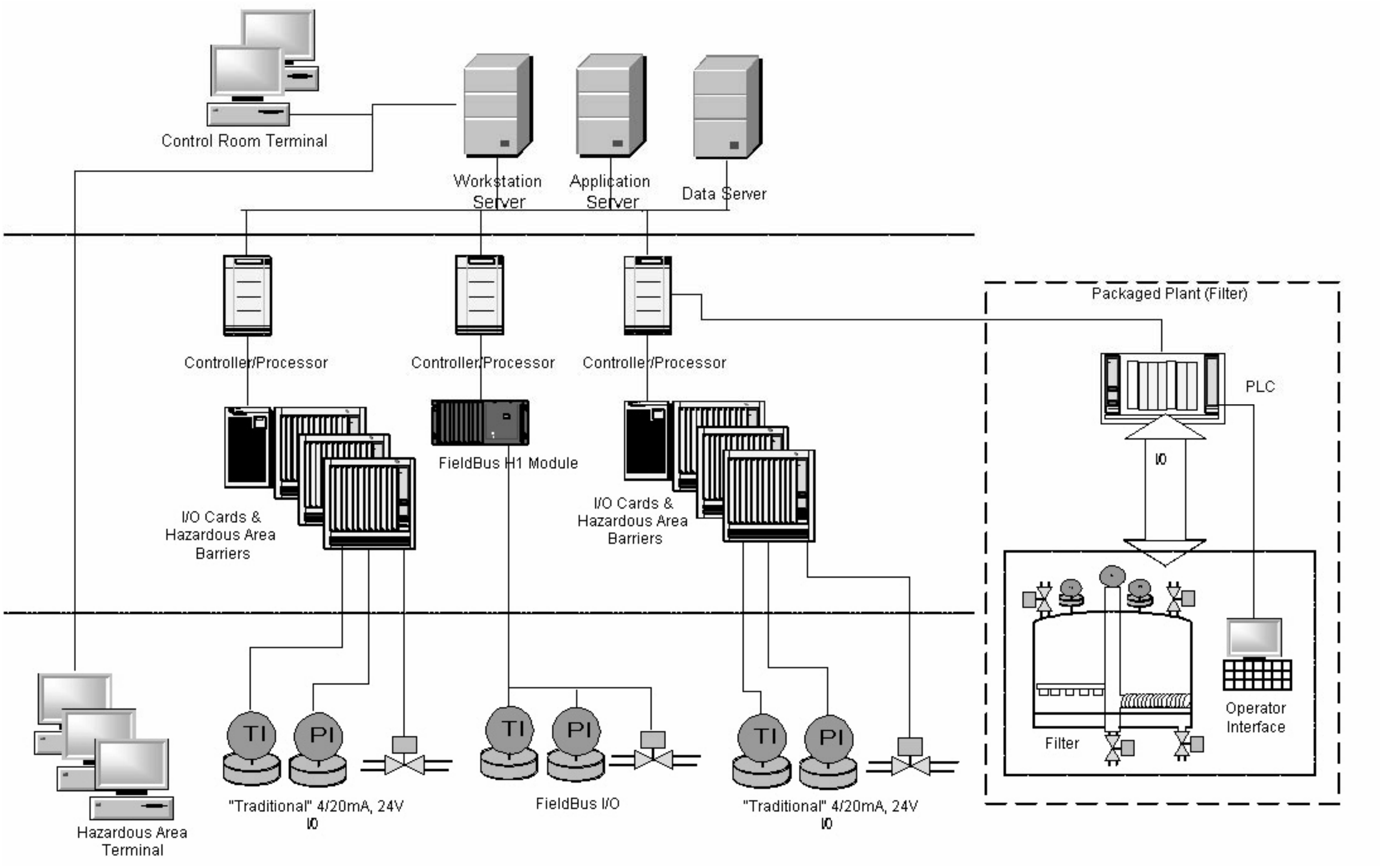


FIGURE 27.1 Typical DCS Architecture.

One of the most challenging aspects for a brown field of the project is how to apply generic principles to a physical plant designed and installed years ago. Models have to be designed that encompass the physical arrangements for unit types but avoid the scenario where each plant unit has its own unique model. Taking the most complex units of a particular class, basing the generic unit model on these, and then using "dummy devices" for simpler units of the same class is one way of achieving this. Examples of Unit Classes that could be identified are:

- Reactor
- Header
- Receiver
- Filter
- Dryer
- Transfer Unit

USER REQUIREMENTS

The majority of User Requirements for a DCS are likely to be common, regardless of whether the DCS is for a brown or green field application. A basic requirement that needs to be determined is the level of automation to be provided. For DCS systems this can vary from providing essentially "remote manual" operation of plant units to automation of individual process operations within units, finally leading to automation of all operations and transfers across process units within a process cell.

The S88 methodology described in the previous section can provide both highly automated and remote manual modes of operation. With careful design and implementation, S88-compliant systems should cope with process changes without major reconfiguration and validation effort being required.

A key consideration to enable system flexibility is to design around the plant unit capabilities rather than focus too narrowly on the specific process requirements. With a well-designed implementation, process changes not requiring physical changes to the plant should be able to be accommodated at the procedural layer of the system.

In order to gain the maximum advantage from an S88-based solution, it is not only the DCS configuration that needs to be considered but the physical plant configuration as well. If the physical configuration of plant units adhere (as far as practicable) to standard arrangements (e.g., valve arrangements for venting/purging reactors), then common coding techniques can be adopted. This reduces the requirements for configuration, documentation, and validation.

For brown field applications the main challenge is to accommodate the S88 principles within the constraints of an existing physical plant configuration. This can be done successfully but the degree of generic coding that can be applied will be more limited than for a green field application.

With regard to electronic records, DCS systems usually store alarm activity, record operator events, and process parameters (trends); additionally batch recipe information and the execution of phases are also stored. All of these types of records are potentially GMP, and hence within the scope of U.S. FDA 21 CFR Part 11. With regard to electronic signatures, particularly for brown field applications, there may be a temptation to just duplicate the batch sheet within the electronic system. Often the requirement for initials (indicating who has performed an action) on a paper batch sheet is confused as being a signature requirement on the electronic system. With the DCS, the use of individual user accounts will provide the records of who has performed an action; the requirement for signatures should be carefully reviewed against the regulatory predicate rules, and the system design should ensure that the application of electronic signatures is compliant with the requirements of U.S. FDA 21 CFR Part 11.

Alarm/event handling needs careful consideration and design. Where alternative (often hard-wired) safety/interlock systems are installed, consideration should be given to mirroring their actions within the DCS or providing a status input into the DCS. In this way there is less chance that an activated hard-wired trip will be misinterpreted on the DCS.

Unless absolutely necessary, automatic recovery routines within DCS configuration should be avoided as these can be confusing to plant operations staff and often involve complex additional configuration, testing, and validation. For sequence logic it is often sufficient to have just a Hold state and Emergency Stop state to address abnormal operating situations. When developing the alarm philosophy, consider the requirements of International Standard IEC 61508[3] and:

- What alarm conditions would indicate notification only
- For what alarm conditions would the impact depend on the stage of the process
- What alarms should initiate a Hold condition on a sequence
- During a Hold sequence, what state should I/O devices move to
- How to initiate an Emergency Stop
- What state I/O devices should move to in the event of an Emergency Stop
- How to deal with adjacent process units when a Hold or Emergency Stop is initiated on a plant unit

For brown field applications, there is likely to be a temptation to "just copy what we had" in terms of alarm philosophy; while this may appear to make the design phase simpler, it is recommended that the strategy is reviewed considering the points above. One final key point with regard to alarm system design is to consider how to ensure that the plant does not routinely run with significant numbers of live alarms.

For green field applications, the type of field instrumentation needs to be determined. Modern systems such as ProfiBus and FieldBus allow multiple field devices to communicate digitally to the DCS. This results in reduced field wiring, allows devices to be configured and calibrated via the DCS, and offers enhanced diagnostics capability. Modern I/O subsystems also allow higher I/O density in rack-rooms, good where space is at a premium. Clearly the benefits of FieldBus/ProfiBus systems in terms of reduced field wiring and higher I/O density are much less likely to be of benefit for an existing facility, incurring significant time and expense in terms of rewiring, installation, and validation costs in addition to the hardware cost.

For brown field applications, therefore, the extent to which the DCS hardware is to be replaced needs careful consideration, particularly at the I/O level. With many modern systems, it is possible to interface different types of I/O subsystem to the controllers. The major manufacturers tend to support field instrumentation and I/O subsystems for extended periods of time, and therefore it will be worthwhile investigating how these can continue to be utilized within the new system. When contemplating the reuse of existing I/O and field devices consider the following:

- How reliable is the legacy I/O subsystem?
- How reliable is the interface to the legacy I/O?
- Is the interface well proven or virtually bespoke?
- Are there likely to be timing issues with the interface (e.g., for PID control loops)?
- Is the interface designed to handle bulk I/O?
- How long will the supplier support the legacy I/O subsystem?

Another alternative is to consider retaining the legacy system for bulk digital I/O, but replace the analog I/O system with modern I/O, perhaps also taking advantage of FieldBus type technology for these loops. Particular attention needs to be given to analog I/O with PID control (more so where the loops are fast acting). Using an interface to a legacy I/O subsystem could affect the ability to accurately control the process variable due to nondeterministic timing/scanning of the I/O by the controller.

VALIDATION APPROACH

For DCS systems, the process and computer validation usually progress in parallel, converging at the Operational Qualification stage. Often there is an overall validation plan for the process with a separate validation plan for the process control system as a subset of this. The starting point for both plans is often a common User Requirement Specification for the facility.

The GAMP Guide[4] provides excellent guidance for the validation of automated systems, and a supporting best practice guide for the Validation of Process Control Systems[5] provides additional information specific to process control systems. The general validation stages are illustrated in Figure 27.2.

Focusing on the process control system validation, and following the traditional "V-Model" approach as defined within the GAMP Guide, the next step would be to develop the functional specification(s) for the system after approval of the User Requirements Specification. An alternative strategy to consider, particularly for brown field applications, is the development of a prototype before functional specifications are generated.

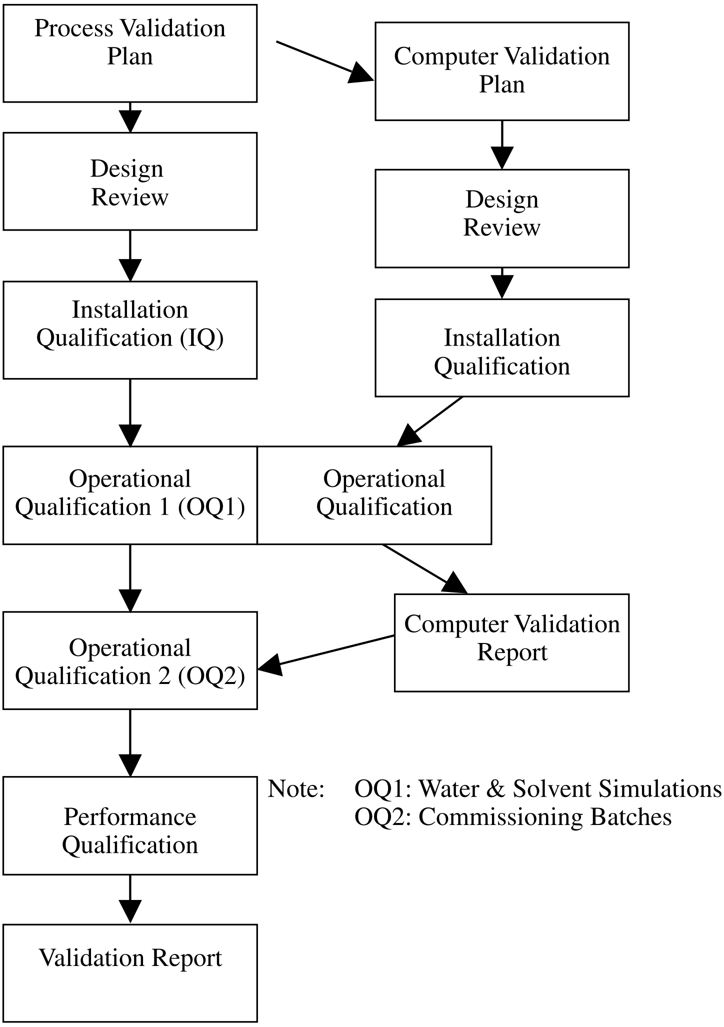

Process Validation Plan
Computer Validation Plan
Design Review
Design Review
Installation Qualification (IQ)
Installation Qualification
Operational Qualification 1 (OQ1)
Operational Qualification
Computer Validation Report
Operational Qualification 2 (OQ2)
Note: OQ1: Water & Solvent Simulations
OQ2: Commissioning Batches
Performance Qualification
Validation Report

FIGURE 27.2 DCS Validation Life Cycle.

A prototype can be a valuable tool for clarifying understanding between the client and system vendor on areas such as:

- The level to which generic code/configuration can be adopted
- How to define plant units, transfer units, and equipment modules
- Phases
  - What phases are required
  - How complex to make each phase
  - What parameters to pass to each phase
  - What batch data to record within a phase
  - How to deal with operator interaction with phases
- Graphics standards
- Alarm and interlock philosophy
- Hold and emergency stop strategies
- Documentation structure

REQUIREMENTS TRACEABILITY

The need to have traceability of requirements from specifications through detailed design and testing is a basic validation requirement.

A requirements trace matrix (RTM) is a tool to map requirements. However, these can become large and difficult to maintain. An alternative approach is to provide implicit traceability by having a common structure and numbering system for all system design and test documentation. The one area where this is often difficult to achieve is with the User Requirements Specification (URS). The URS often covers not only the DCS requirements but those of the wider project, and is also produced before the system vendor has been selected. In this case a requirements traceability matrix should be generated to confirm requirements relevant to the DCS are addressed within the functional specifications.

FUNCTIONAL/DESIGN SPECIFICATIONS

The size of most DCS systems makes it impractical to have a single functional specification. Specifications are often separated with regard to their area of functionality (e.g., graphics, sequences, I/O schedule, etc.). However, with S88-based systems it is usually most efficient to structure the specifications around the S88 model. The use of generic models can also be useful. Table 27.1 provides an example of a structure for functional specifications.

Associated project documentation used by other engineering disciplines is also essential, and it is important that an effective change control system is in place to communicate changes made to Engineering Lines Diagrams (ELDs/P&IDs), I/O Schedules, and Process Descriptions.

Detailed Design Specifications may be required; this depends on the level of detail included within the functional specifications. The main principles are that there should be sufficient information within the specification to enable the system to be configured and to provide sufficient information for the configuration to be subsequently maintained.

SYSTEM CONFIGURATION

System configuration needs to be structured and planned carefully. It is not necessary to have all specifications in place prior to commencing any configuration, but if parallel build and design activities are to take place, they need to be carefully planned. It is not necessary, for example, to have all phases specified before configuration of control modules can take place.

TABLE 27.1
Example Functional Specification

Content	Generic/Specific	Comments
Control Modules Model	Generic	Defines the generic types of control modules, their functionality, alarm attributes, and faceplate displays
Graphics Model	Generic	Sets display standards, colors layout, etc.
Alarm and Security Model	Generic	Sets standards for alarms and interlocks; describes User Profiles and system security settings
Batch Model	Generic	Describes how the batch executive interfaces with units and phases, how batch reporting is to be configured
Equipment Model	Generic	Specifies the functionality of generic equipment modules such as temperature control units (TCUs)
Unit Model	Generic	Describes each phase in structured English; describes how the Unit Hold and Emergency Stop function
Unit Specification(s)	Specific	For each plant unit defines the unit type, phases, control modules, equipment modules, Hold and Emergency Stop states, interlocks, and transfer routes
Procedural Specification	Specific	For each process stage describes the unit operations and phases together with the recipe values of parameters to be passed to phases
Hardware Specification	Specific	Defines the hardware and network architecture including server specifications, UPS, I/O subsystem and operating system, and application software versions

The generic specifications are essential to successful implementation of specific instances. Consider for example, the Control Modules Model. This defines all control module types such as valves, motors, and control valves. Specific control module instances are subsequently propagated from the “typical” specified in the model. The propagation of all control module instances can be automated, which speeds up system configuration, but any error subsequently identified with the typical would apply to all instances, once propagated. It is therefore recommended that all generic models be thoroughly tested prior to propagation of instances. This particularly applies to phases and control modules.

One opportunity to streamline validation that is presented when using the generic approach is to essentially perform an Operational Qualification on the generic model prior to its being used to propagate each instance. The Operational Qualification is essentially a thorough prespecified set of tests conducted on a single instance from the generic model. Testing must be formally conducted and recorded (as detailed within GAMP), and all qualified configuration must be placed under strict configuration control on completion of testing. Where the generic model contains options (e.g., accommodating both fail open and fail closed block valves), then sufficient instances should be generated and tested to ensure that all options are covered, and stress/boundary tests also included as applicable.

By performing the thorough OQ testing on the generic model, it is then considered acceptable to subject each individual instance to lower levels of confidence testing. Table 27.2 considers some tests that could apply to the generic model for a block valve and those that would need to be conducted for each instance generated.

As S88 systems are modular, there is a potential benefit for both new and particularly brown field facilities in terms of the delivery and installation of a partially configured system to the site to enable an early start to commissioning activities.

Such a partially configured system could, for example, have all control modules and graphics configured, but with no phases or procedures. The hardware supplied could be a subset of the final

TABLE 27.2
Example Tests

Block Valve Test	Test Generic (Yes/No)	Test All Instances (Yes/No)
Verify Valve Instance generated	Yes	Yes
Output assigned to correct I/O address	Yes	Yes
Limit Switch Input assigned to correct I/O address	Yes	Yes
Valve Appears on correct graphic	No	Yes
Correct faceplate display is activated	Yes	No
Verify alarm functionality	Yes	No
Verify operation in auto, manual, interlock	Yes	No
Verify operation of any interlocks	No	Yes

system, e.g., only one controller, one server, and a couple of displays. Such a system would provide the ability on site to verify the graphics and operation of all plant I/O devices from the graphic displays, and allow plant operations staff an early opportunity to become familiar with the system.

Clearly, precautions would need to be considered in order to maintain control over the system. With this approach the potential of having two parallel systems, both undergoing change, could exist. The development system of the vendors would be continuing to have the phase logic developed, and changes on the test system, at site due to correction of any errors identified. A solution might be to simply not allow any changes to be made to the test system; all errors found during testing on site would be logged on fault reports, and these passed back to the system vendor for the development system to be corrected. Verification of correct operation in the live environment would then be deferred until delivery of the final system to site.

CODE REVIEW

The subject of code review is often one of some debate with regard to process control systems. Frequently it is argued that as the systems are essentially "off the shelf" and configurable that code reviews are not necessary. While this is definitely the case for some aspects of the system (e.g., graphics configuration), there are usually some configuration items that should be subject to review. For the majority of modern DCSs, phase logic is configured using sequential function charts (SFCs). SFC logic, although at first glance a relatively high-level configuration language, can be considered code rather than configuration. SFCs should therefore be developed in accordance with predetermined standards, and be subject to peer review.

TRAINING

Systems support staff should be trained on systems as part of demonstrating competence to support the system; formal training courses are usually available from system vendors for DCSs. This represents considerable expenditure, both in time and financially, and thought should be given to the timing of these courses. Experience has shown that support staff participating in Factory Acceptance Testing (FAT) after attending courses gain the most benefit. The FAT tests provide an ideal opportunity to consolidate the knowledge gained on the courses, especially as the systems vendor's experienced development staff will be on hand during this time.

For Operator training, often the standard DCS vendor-supplied training packages can be too generic. If this is thought to be the case, then tailor-made courses can be developed for the system. The advantage of this is that training will be based on the actual system graphics, phase and recipe logic, and alarm system. This more focused training is often both more effective and can be delivered in less time than the standard offerings.

The disadvantage of this approach is the considerable time, effort, and expense to produce the customized courses, and also that the courses cannot be fully developed until all system design has been completed.

One final, and perhaps most significant advantage of having customized training developed is that the Client then owns the training material. Post-implementation of the DCS courses can be periodically run on site for new operations staff, and the material maintained to reflect the system configuration.

PREDELIVERY TESTING

There are a number of stages of testing associated with a typical DCS. System configuration should be subjected to tests by the system's vendor (including code review as discussed above) before any form of client testing.

As elements of the system are released for testing, it is important that they are then subject to secure configuration management and change control.

FAT is the usual point at which Client testing commences; this should be a confirmation exercise rather than a debugging activity. FAT can be used as part of the formal system qualification, but where this is the case, testing should be performed against predetermined specifications and under controlled conditions. Test specifications should be prepared in accordance with GAMP 4 and all results, and wherever possible evidence (e.g., screen dumps, alarm/event printouts, etc.) recorded and collected.

As discussed in the section on System Configuration, thorough testing of generic models can be used to reduce the test burden for each individual control module or phase instance; again, this needs to be well documented and controlled.

Tests should include individual configuration components and integration and stress testing.

Simulation packages can be very worthwhile in order to effectively test both the phase logic and recipes. The package emulates the plant inputs, providing both analog inputs (e.g., temperatures, flows, pressures) and digital inputs (e.g., valve position confirmation, pump, agitator running signals, etc.).

The question "Do we need to validate the simulation package, and its configuration?" can sometimes arise. This is probably not necessary if the following points can be verified:

- Phase logic is retested as part of water and solvent trials during Operational Qualification.
- The package is not capable of changing the configuration of the DCS in any way.
- Any errors in the simulation package would be more likely to result in test failures rather than mask a true error in the DCS configuration.

One area with simulation packages that does need to be carefully controlled is the (usually minimal) reconfiguration of the controllers to enable them to "look" at the simulated I/O rather than the real I/O cards, and to ensure that this is reinstated on completion of testing.

The following checklist provides an illustration of the areas to be tested for a phase:

- All paths through the sequential function charts
- All recipe parameters passed to the phase
- Operating modes to be tested, i.e., run in Manual and Automatic
- Operation of Hold and Emergency Stop states
- Correct Operation of phase abort and resetting of all parameters

- Operator messages to be verified
- Correct recording of batch data

For most of the above, and in order to satisfy boundary and negative testing criteria, multiple runs of the phase would be necessary. Evidence collected during the test would typically comprise:

- The completed test script
- Extract of the pages from the functional specification for the phase

And for each test run:

- Print out the recipe used to run the phase
- Copy the batch report generated by the phase
- Give an alarm print
- Copy the phase logic (SFC)

On completion of FAT, all software versions (application and configuration) should be recorded to enable these to be verified on delivery of the system to site.

QUALIFICATION

System qualification should be performed against preprepared and approved protocols. During this phase the DCS should be subject to the Client's formal change control process.

Installation Qualification (IQ)

Following final installation of the system on site, Installation Qualification verifies that all system hardware has been delivered and installed correctly, and is to the correct specification It is also important that the delivered system matches that signed off at the end of FAT. Table 27.3 suggests some typical IQ test areas.

Operational Qualification (OQ)

Operational Qualification of the system often comprises two phases and runs in parallel with process OQ:

- OQ1: Verification of basic functionality and calibration of instrument loops, verification of alarms/interlocks, followed by verification of process sequences and loop tuning — often using water to simulate the process.
- OQ2: Process sequence verification and further loop tuning using solvent simulations and then commissioning batches.

As discussed in the previous section on predelivery testing, Factory Acceptance Tests can be used to supplement formal system qualification, but only where the tests have been well specified, documented, and performed, and under controlled conditions.

Factory Acceptance Tests cannot be used in place of OQ tests that clearly need the system to be in the operational environment, e.g., control loop testing, instrument calibration tests, and sequence testing in conjunction with the process equipment. Table 27.4 suggests some typical OQ test areas.

On completion of this stage often the Validation Report for the DCS must be completed as a prerequisite of the process validation entering the formal PQ stage.

TABLE 27.3
Typical IQ Tests

Area	Test	Scope
Hardware	Confirm all hardware has been supplied and installed in accordance with specification.	Servers Workstations Network Hardware Controllers I/O Racks I/O Cards Barrier Systems Field Instruments Power Supplies Uninterruptible Power Supplies
Cabling	Confirm all cabling is installed and labeled in accordance with drawings and specifications.	Network Cabling Data Highways Power Supplies Cabling to I/O racks Field Cabling
Power-up/Diagnostic Checks	Confirms that all systems power up correctly. No error messages are present from system diagnostics. EMI/RFI checks (usually susceptibility to rather than emission).	Servers Workstations Network Hardware Controllers I/O Racks I/O Cards Barrier Systems Field Instruments Power Supplies Uninterruptible Power Supplies (UPS)
Software	Confirm all software has been loaded and versions are correct.	Operating Systems Application Software Configuration Bespoke Code

Note: OQ2 can be considered Performance Qualification (PQ) of the DCS. During this stage of testing, only very minor issues should be apparent with the system.

Performance Qualification (PQ)

During formal PQ of the process, only very minor (formally controlled) changes should be required to the process control system as the system should be considered validated at the end of OQ2. Any changes made to the process control system during this phase should be assessed as to their impact on PQ and documented within the process validation report.

OPERATION AND MAINTENANCE

Once in operational use, the validated status of a process control system must be maintained and periodically reviewed to verify continued compliance with regulatory requirements.

Procedural controls would normally be established to cover the following areas:

TABLE 27.4
Typical OQ Tests

OQ Phase	Test
OQ1	Confirm that all control modules operate "end to end" from the operator displays to the field; verify correct device status is indicated on the graphics and that the graphics are a correct representation of the plant configuration. Confirm that all analog instrument loops are correctly calibrated (over their entire measurement range) from the field instrument to the DCS displays any other indicating devices. Where process trending functionality is included this may also be verified during this test. Verify correct operation of alarms, trips, and interlocks. Verify correct operation of process sequences during water simulations; tune control loops. Ensure all process paths are tested, including any "emergency stop" and "hold" conditions. Verify correct operation of any recipe management and batch data recording during these tests. Consider any stress tests that could not be performed during off-site testing; for example, verify how long a UPS will support the system following power failure.
OQ2	Verify correct operation of process sequences during solvent simulations. Fine-tune control loops. When there is confidence that the system is operating satisfactorily and process validation is ready, product may be introduced into the plant and commissioning batches processed. Such tests are conducted against the batch processing sheet and are often shared tests for process and computer validation.

- Change Control
- Configuration Management
- System Backup/Restore
- Data Archiving/Restore
- System Access

The general requirements are similar to those for any validated computerized system. However, large DCSs do tend to undergo significant numbers of changes. It is therefore important to review the effectiveness of the change control and configuration management processes, and, in particular, ensure that system specifications are maintained in an accurate state.

Remote access support agreements are often available from system vendors. These can be worthwhile in enabling faster resolution to system problems. However, consideration must be given to ensuring that system/data integrity and security are not compromised.

DECOMMISSIONING

When a system reaches the end of its operational life, record retention requirements should be an essential consideration before destroying all hardware, data, and associated documentation.

Even for a system that is deemed not to contain electronic records, there will still be a requirement to retain validation and associated (e.g., specification, change control, etc.) documentation for the retention period following completion of manufacture of the final batch of product.

For systems deemed to contain electronic records, provision must be made for ensuring that all such records archived remain secure and can be retrieved for the required record retention period. Table 27.5 lists some options that could be considered.

ACKNOWLEDGMENTS

The author would like to acknowledge the contribution and valuable discussions of Ian Talmadge, Team Lead Engineering, Invensys Systems (U.K.) Limited**.**

TABLE 27.5
Archiving Options for Electronic Records

Option	Advantages	Disadvantages
Migrate all data to the new system.	No requirement to maintain old hardware. Ready access to information. If same manufacturer, then likely to be a standard upgrade/migration path.	Migration method needs to be secure. Not an option if the old system is not being replaced.
Migrate all data to standard format, e.g., PDF.	No requirement to retain old hardware. Ready access to information.	Migration will need to be validated — possibly bespoke software required to transfer records. Unlikely to support secure transfer of electronic signatures.
Retain sufficient elements of the old system to enable continued data retrieval capability.	Low cost option (at least initially).	As time progresses, continued support of the legacy system will become more difficult and expensive.
Migrate old records to paper.	No requirement to retain an electronic system.	Regulatory risk — unless this can be clearly demonstrated as the "last resort." Will not be acceptable for records with electronic signatures.

REFERENCES

1. FDA (1997), *Electronic Signatures and Electronic Records*, Code of Federal Regulation Title 21: Part 11, Food and Drug Administration, Rockville, MD.
2. ANSI/ISA-S88.01 (1995), Batch Control Part 1: Models and Terminology.
3. IEC 61508-1 Corr.1 (1999), Functional Safety of Electrical/Electronic/Programmable Electronic Safety-Related Systems — Part 1: General Requirements.
4. ISPE (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), GAMP Forum, December (www.ispe.org).
5. ISPE (2003), *GAMP Good Practice Guide: The Validation of Process Control Systems*, GAMP Forum — Control Systems Special Interest Group (www.ispe.org).

28 Case Study 10: Electronic Batch Recording Systems (Manufacturing Execution Systems)

Peter Bosshard, F. Hoffmann-La Roche
Ulrich Caspar, F. Hoffmann-La Roche
Robert Fretz, F. Hoffmann-La Roche

CONTENTS

The pharmaceutical industry is obliged to document carefully every single step in the drug manufacturing process. This is one of the mandatory activities to ensure that a drug is safe for use. An Electronic Batch Recording System (EBRS) is a system that creates electronic batch records. However, the system described in this case study does far more. It could be considered a Manufacturing Execution System (MES). The name is given based on the approach. If plant automation is the main driver, the term MES is more often used. EBRS as a term is used more to describe the electronic batch recording capability. Either way, EBRS facilitates efficient documentation by making good use of the possibilities offered by today's Information Technology (IT). The following case study gives an insight into the business needs leading to the introduction of an EBRS. Furthermore, it describes the system and the realization concept, the system specification, and the approach used for computer system validation.

BACKGROUND

In 1992, the decision was taken to introduce an EBRS in the galenical production of the pharmaceuticals division of F. Hoffmann-La Roche Ltd. in Basel, Switzerland. The project was launched as "PK System" in August 1994. By mid-1996, a major part of the system was in operation and working as intended. Back in 1992 there were no commercial systems available for this detailed functionality. Today there are several competitors working in this field. Some of the most important ones are Consilium, ProPack Data, SAP, and Werum. The implementation of a Commercial Off-The-Shelf (COTS) solution would have been the preferred option.

BUSINESS NEEDS

The functions of the planned computer system have to satisfy the current requirements of Good Manufacturing Practices (GMP) and associated regulatory guidance.[1] Regulations such as the GMPs in the European Union (EC GMP) or the Code of Federal Regulations (CFR) by the U.S. health authorities require the recording of batch-specific information during the production stages (EC GMP 4.17[2] and 21 CFR 211.188[3]). The manufacturing procedures and the batch records must be properly reviewed and electronically signed in conformance with 21 CFR 11.[4] Then the product is released for further processing (e.g., packaging [21 CFR Part 211.192[3]]). At the point of decision, the galenical production process, from active ingredient to the galenical dosage form, led to 4000–5000 batch records per year, consisting, on average, of ten pages each. These records were reviewed manually by the responsible pharmacists, the plant supervisor, and other pharmacists.

PROJECT OBJECTIVES

The main objective of the project "EBRS" was to develop a computer-aided batch recording system. This system should deliver the benefits described below.

Time Savings

- Batch record review is performed immediately, not delaying the following production steps (e.g., packaging).
- The review process of the electronic batch records becomes much easier, with all data available in an orderly and structured form.
- Discrepancies (if any) are easier to analyze if they are all listed automatically by the computer.
- Recording the entire process and control data allows the easy performance of investigations for failure analysis and production optimization.
- Batch records can easily be sent by electronic mail, provided that this function is validated and secure (no changes possible to the record). This is important in case product batches are exported to other countries.
- All signatures can be done electronically.

Improvement of Process Control

- The completion of every production step, the corresponding results, and procedure parameters can be checked immediately.
- The sequence of the production steps can be defined by the EBRS, thus managing the manufacturing process (if necessary).
- The state of the production equipment is controlled. This means, for example, that a production vessel can be used only if it is clean, and a balance can be used only if it is calibrated.
- The success of process validation or preventive maintenance is monitored.
- All equipment and materials can be fully traced.

Improvement of Security

- Boundary checking of process parameters is performed automatically. This means that whenever manufacturing data are outside the specified limits (alert and in-process control), the application reports this discrepancy (on screen or by electronic mail). This increases the certainty that any irregularities are detected properly. Thus, it gives management the possibility to react immediately so that timely and cost-saving corrective actions can be taken.
- The automatic data capture of the relevant process reduces input errors.
- Every batch record must be accessible at least for the shelf life of the drug (21 CFR 211.198[3]). The introduction of the new system gives the possibility to store the batch records electronically on various storing devices such as optical disks. Compared to paper-based batch records, electronically stored records need much less room and also increase the safety of the data.

VALIDATION LIFE CYCLE

Validation

The validation was performed according to the "Roche Computerized System Validation Policy and Guidelines." This comprised the definition of a Validation Plan, the performance of the planned activities, and the creation of a Validation Report. In addition, the project was accompanied by external consultants providing knowhow regarding the validation-specific aspects of the development, including the management and auditing of the software developer. The scope of the validation activities was defined by GMP Analysis (also known as a GMP Assessment).[5] The system requirements were analyzed regarding their GMP relevance. The network functionality was beyond the scope of this validation.

System Description

System Concept

A batch record, as stored in the database, is represented by a collection of different data on the actual production procedure and the current production environment. Back in 1994, at the point of decision, the data available electronically were stored in several databases. Database update was time-consuming and difficult; data analysis even needed different program interfaces. Therefore, the database for the new system had to be a central, uniform database for the whole production plant.The bill of material is fed from ERP as well as general planning. LIMS is another interface for the input of analytical data that is important in the calculation of correction factors for the content of API.

Realization Concept

This project was divided into two phases to ensure management control of cost, time, and resources.

Phase 1

- Administration of the general data
- EBRS for bulk production (granulation, ointment, syrup, and sterile solution manufacturing)
- Data processing of in-process control and environmental monitoring tests results

Phase 2

- EBRS for the sterile filling plants, capsule filling plant, and tablet compressing plant
- Administration and controlling of the maintenance data of production equipment
- Administration of personnel education data
- Automatic data capture from production and measurement equipment
- Controlling of the filter test procedure

System Specification

Figure 28.1 gives a schematic overview of the various functional modules of an EBRS.

General Requirements

- Definition of authorizations for the specific functions of the system to ensure the appropriate workflow at the production line
- Reporting system for fast information regarding encountered discrepancies from the specified limits
- Identification of the materials used with unique identifiers (such as raw material, intermediates, filters, spare parts)
- Interface to the production planning system (MRP II)
- Identification of production staff (attributes, resources, and training)
- Identification of production equipment (status, cleaning, calibration)
- Identification of the types of production rooms used (cleaning, sterile, control, etc.)
- Identification of desktop workspace (workstation, screen, bar code reader, etc.)

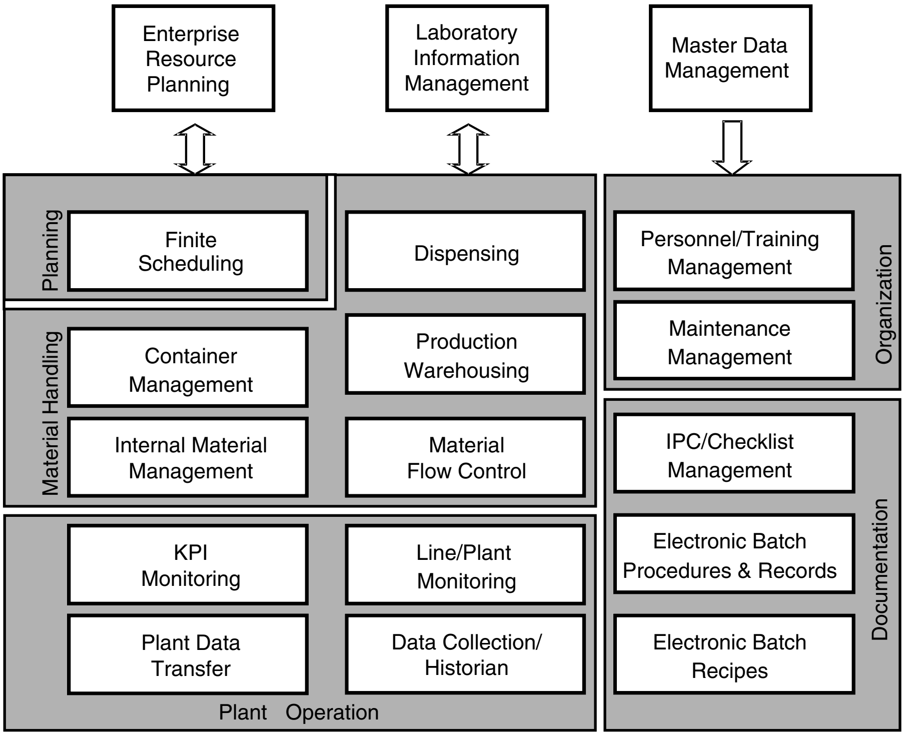


FIGURE 28.1 Schematic Overview.

Bulk Production

- Definition of master production protocols (21 CFR 211.186[2]), through which the operator is instructed (through individual operating procedures) how the various production steps must be carried out.
- Creation of the master production record by copying the valid master production protocol before production is started.
- Filling in data by workers turns the master production record into the batch production record.

In-Process Control/Environmental Monitoring Test

The following parameters must be documented by the system:

- Environmental monitoring tests
- In-process control
- Validations and calibrations

Risk Assessment

Risk assessment is an important step in identifying the depth of the validation effort. Scientifically, risk is the product of the probability of an incident multiplied with the possible impact of the consequences. For this EBRS, all modules were identified as relevant for GMP. However, for the testing, the functions were classified into three classes — one was direct product influence, e.g., interfaces to the balances and the other was indirect influence on the product e.g., maintenance and training. The third category was the one without relevance to the product like performance reports, capacity utilization reports, etc.

Prequalification Phase

The prequalification phase consists of the following:

- Analysis and definition of user requirements
- Definition of the system delivery specification
- Technical system design[6]

Qualification Phase

In the qualification phase, programming was performed according to user requirements and the system delivery specification. The finished program was tested by the developer using unit and integration tests. In addition, Installation Qualification (IQ) of the hardware and the Operational Qualification (OQ) of the complete system (hardware and program) performed on-site. The qualification activities included the following:

- Definition of programming standards
- Ensuring the independent functioning of each individual software
- Module
- Vendor Audits (Supplier Audit)
- Program description (source code)
- Source Code Review
- Ensuring the proper integration of software and hardware
- Definition of the necessary installation procedures
- Definition of the hardware components used

Test Phase

In the test phase, user acceptance tests were performed. The goal of these tests was to verify whether the completed system was performing according to the user requirements defined in the prequalification phase. Successful test completion was documented in the Validation Report, confirming that the system had been validated for use in daily business. Also included in this phase was the development of various operating procedures. The activities carried out are summarized below.

GMP Assessment

User requirements must be analyzed regarding their risk potential and GMP relevance. The analysis, as defined by Heinrich Hambloch,[5] determines if a function:

- Has influence on the pharmaceutical technical quality
- Affects the medical safety of the drug
- Has influence on the data that become part of the registration documents
- Is critical for another important reason

Test Strategy

A precondition for any testing is the availability of a document that correctly specifies the functions of the system. It is the basis for the verification test specification. In the case of EBRS, this was the User Requirement Specification (URS). The major testing was done as black box testing. White box testing was limited to the modules ranked as most critical in the GMP Assessment. This included the formulas of active ingredient strength and verification of algorithms in the source code.[7]

Verification Test Procedure

The first step toward the validation of the system was the development of a testing procedure to be used for the different software modules and for future revalidation. This procedure defines how test plans must be specified, how the tests are performed, and how they are documented. The goal of testing is to establish documented evidence that the system is performing according to the specifications.

Defining the Verification Test Specification

Once the functions to be tested had been completely identified based on their GMP relevance, they were added to the verification test specification.

For each test case, the following information was added:

- Verbal description of the goals that a specific test must achieve
- Detailed description of the test procedure
- Definition of the required test data and the expected results
- Definition of the test protocol
- Listing of any related documents referred to in the tests

The following areas and functions were tested during the validation of Phase 1 of EBRS:

- *Daily usage of the system*, including authorizations and security, Windows™ menu control, error reporting, and communication with other devices (e.g., peripherals)
- *Characterization and handling of materials*, including the definition of raw materials, products, and auxiliary material (e.g., packaging)
- *Workflow at the production line*, including production steps, line type, line status, and line schedule
- *Production equipment and locations*, including definition and current state of buildings, production areas, and individual workstations
- *User specifications*, including the definition of user characteristics, groups, responsibilities and privileges, training, and scheduling
- *Control of auxiliary materials and equipment*, including characteristics and calibration of computers, containers, scales, and so on
- *Supplier management*, including the analysis of the vendor's Quality Management System (QMS) and its ability to deliver the requested system
- *Lot data control*, including the identification and maintenance (corrections, deletions, restrictions) of the lot data and content/ingredients calculations
- *Further testing*, including areas such as material storage, MRP II, cleaning protocols, product content, and archiving

Corrective Actions

Any discrepancy from the expected results that were encountered during the testing had to be analyzed for their relevance and documented. Problems that prevented GMP conformance had to be corrected immediately.

Special Testing

Stress Testing: A stress test regarding the data volume was performed by expanding the database to the possible volume of a half year's production records. The acceptance criterion was that it would still be possible to use the system with reasonable response times.

Client Interrupt Testing: The system was tested to determine what would happen in the case of a sudden breakdown of a client's Personal Computer (PC). The acceptance criterion was that the database would not be corrupted. Data not saved properly should thus be rolled back automatically to the latest secure version of the data.

Automated Testing

Splitting the project into two phases made it necessary to integrate several modules sequentially. This process led to frequent revalidation activities. To reduce the testing time and to prevent typing errors during test execution, an automated test tool was used. However, this tool did not deliver all the benefits initially expected because manual editing (programming) of the generated scripts for the testing tool could not be avoided completely and this proved to be very time consuming. Such editing was necessary because:

- Windows™ objects (e.g., buttons, menus) reacted differently from testing tool expectations.
- Scripts had to be commented to ease later editing, such as the insertion of additional test cases.
- Date- and time-related functions, which are quite frequent in batch recording systems, led to problems during testing. For example, running a test script on a Friday and creating a production order for the following day triggered the "unexpected" question of whether the production order really should be started on a Saturday or the next Monday. Such questions would not arise Sunday through Thursday.
- Furthermore, the execution of the test scripts was halted due to Microsoft Windows™ problems. Finally, maintenance of the test scripts became difficult once the system had been handed over to the business since the knowhow required to maintain and rerun the test scripts was no longer available.

Example

The following example (Table 28.1) describes how testing was done for a specific function of the system. The situation described in the example is such that within the EBRS menus can be specified for the combination of:

- Organization (plant, production line, etc.)
- Workplace type (weighing, drying, sterile conditions, etc.)
- User group (= access level), where each user group has access rights for its own or all lower access levels

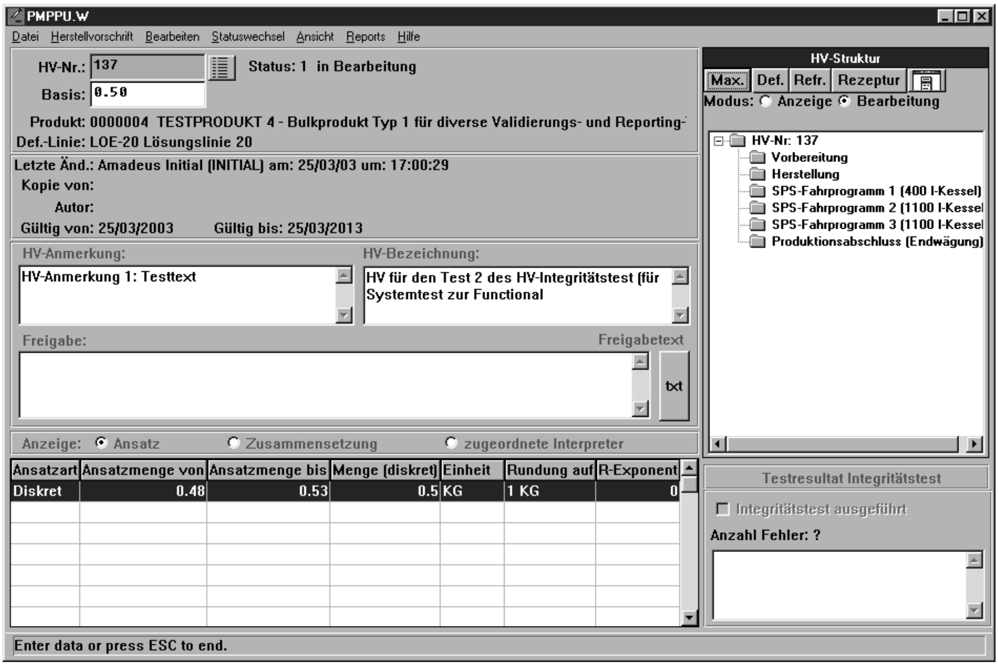
A validation database was set up. This validation database always contains a defined amount of data and leads to precisely predictable results. For example, in this validation database, person 1 was set up with user name = TESTP1, belonging to user group 1 with the access level 1. The PC on which the validation testing was run was set up as workstation WSOOOl, belonging to the organization B1 with the workplace type AT2. There is no menu prepared for organization B1 with workplace type AT2 and user group level 1.

Ongoing Evaluation

Once the system was successfully implemented and partly in operation in the daily business, the validated status of the system was maintained using the necessary technical and organizational procedures.

TABLE 28.1
Example Test Case

Function	Log-in of person 1 on the workplace defined by organization (plant) = B1 and workplace type 2 = AT2
Test Procedure	1. Fill out log-in mask using the test data 2. Fill out workplace identification mask (log-in 2) using the test data (see Figure 28.2) 3. Check if the menu displayed corresponds with the expected result (see Figure 28.3)
Test Data	4. Log-in mask: User name = TESTP1, password = XXXX00 5. Log-in 2 mask: workplace type = AT2, organization = B1
Expected Result	No menu should be displayed, because there is no menu defined for user group 1 at the workplace B1/AT2 (see Figure 28.3)
Test Documentation	Printout of the workplace identification mask and printout of the main menu mask (sign off with initials plus date)


PMPPU.W
Datei Herstellvorschrift Bearbeiten Statuswechsel Ansicht Reports Hilfe
HV-Struktur
Max. Def. Refr. Rezeptur
Freigabetext
txt

HV-Nr.: 137 Status: 1 in Bearbeitung
Basis: 0.50
Produkt: 0000004 TESTPRODUKT 4 - Bulkprodukt Typ 1 für diverse Validierungs- und Reporting-
Def.-Linie: LOE-20 Lösungslinie 20

Letzte Änd.: Amadeus Initial (INITIAL) am: 25/03/03 um: 17:00:29
Kopie von:
Autor:
Gültig von: 25/03/2003 Gültig bis: 25/03/2013

HV-Anmerkung: HV-Anmerkung 1: Testtext
HV-Bezeichnung: HV für den Test 2 des HV-Integritätstest (für Systemtest zur Functional
Freigabe:

Anzeige: ● Ansatz ○ Zusammensetzung ○ zugeordnete Interpreter

Ansatzart	Ansatzmenge von	Ansatzmenge bis	Menge (diskret)	Einheit	Rundung auf	R-Exponent
Diskret	0.48	0.53	0.5	KG	1 KG	0

Modus: ○ Anzeige ● Bearbeitung

- HV-Nr: 137
  - Vorbereitung
  - Herstellung
  - SPS-Fahrprogramm 1 (400 l-Kessel)
  - SPS-Fahrprogramm 2 (1100 l-Kessel
  - SPS-Fahrprogramm 3 (1100 l-Kessel
  - Produktionsabschluss (Endwägung)

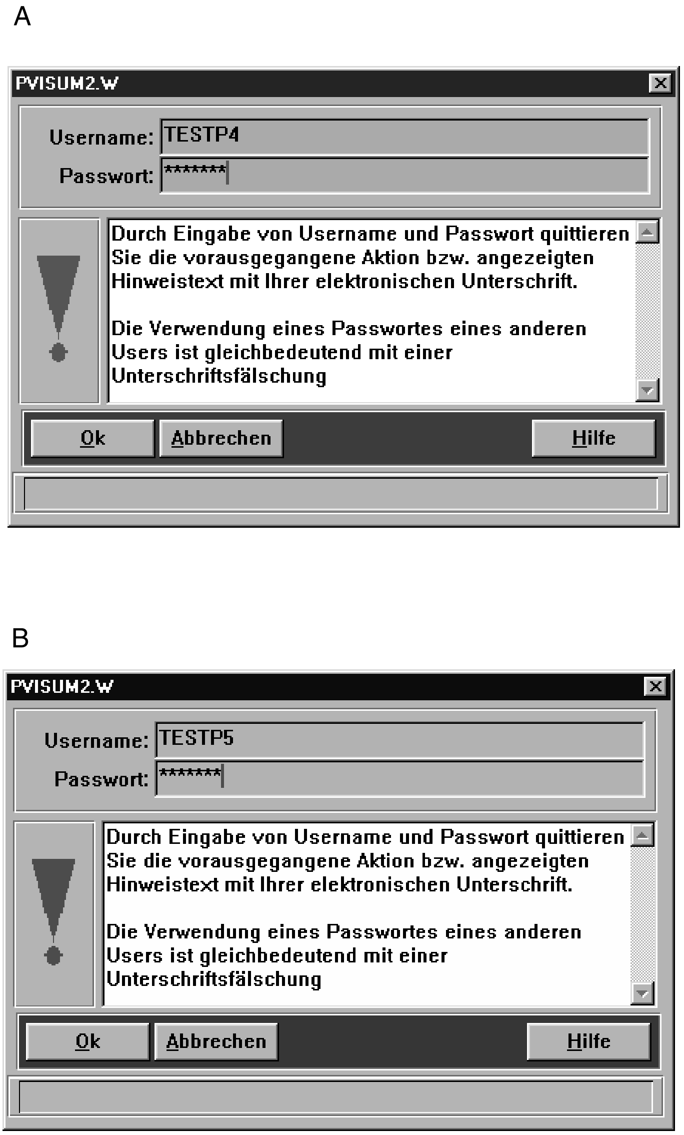
Testresultat Integritätstest
□ Integritätstest ausgeführt
Anzahl Fehler: ?

Enter data or press ESC to end.

FIGURE 28.2 PK System Workplace Identification Mask. *Note:* This screenshot illustrates a master batch record that is ready for approval of quality assurance and production.

Periodic reviews are performed to verify that the system is operating as specified (performance, disk space), that it is, properly administered (e.g., authorizations), and that the documentation is accurate.

Change and configuration management involves procedures that control and report the implementation of changes that may affect the validation status of a system. This includes the tracking of problem handling, resulting from fault reports or change requests, to their solution. Change management ensures that the configuration of the system is identifiable and reproducible.[3]



A

PVISUM2.W

Username: TESTP4
Passwort: *******

Durch Eingabe von Username und Passwort quittieren Sie die vorausgegangene Aktion bzw. angezeigten Hinweistext mit Ihrer elektronischen Unterschrift.

Die Verwendung eines Passwortes eines anderen Users ist gleichbedeutend mit einer Unterschriftsfälschung

B

PVISUM2.W

Username: TESTP5
Passwort: *******

Durch Eingabe von Username und Passwort quittieren Sie die vorausgegangene Aktion bzw. angezeigten Hinweistext mit Ihrer elektronischen Unterschrift.

Die Verwendung eines Passwortes eines anderen Users ist gleichbedeutend mit einer Unterschriftsfälschung

FIGURE 28.3 PK System Main Menu Mask. *Note:* The signature screens pop up for Production (Testp4) and Quality Assurance (Testp5).

Currently, the system is still under implementation, and revalidation is done frequently each time a system module is handed over to operations. Experience with ongoing validation of the system has yet to be gained.

CONCLUSIONS

Realization of the Expected Benefits

High Degree of Automation Required

To be efficient, electronic batch recording must eliminate more than half of all manual collection, analysis, and review work. Otherwise, there will not be any significant time reduction.

Creation of a Central Uniform Database

To ensure the efficiency and effectiveness of the system, all data required for batch review and release should be stored in one central database, eliminating the necessity of interfaces between different databases.

Efficient Validation

Availability of a Company Policy on Computer System Validation

Validation was performed according to the Roche company policy on CSV.[3] A Validation Plan was established, the activities were performed and documented according to this plan, and the results were entered in the Validation Report. As a result, the product (drug) manufacturing process is better documented and analyzed, thus contributing to the safety of a product.

Analysis of GMP Relevance and Risk Assessment

The validation approach should be based on the analysis of the GMP relevance for each function of the system. The results should then be used to define the test strategy and the corresponding test cases.

Proper Vendor Selection

It is important to select a vendor who is capable of delivering the system in compliance with the requirements of the CSV policy.[3] This ability should be verified by proper vendor selection and auditing.

Ensuring Ongoing Validation

Once the system is in a validated state and handed over to operation, it is important to ensure that the system remains in such a state. The available operating procedures must take into account that the unit operating the system does not have the profound system knowledge that the project team had.

REFERENCES

1. FDA (1983), *Guide to Inspection of Computerized Systems in Drug Processing*, Food and Drug Administration, Department of Health and Human Services, Rockville, MD.
2. "The Rules Governing Medicinal Products in the European Union" (1998), Volume 4 Good Manufacturing Practices, European Commission, Directorate General III — Industry, Pharmaceuticals and Cosmetics. http://pharmacos.eudra.org/F2/eudralex/vol-4/pdfs-en/cap4en.pdf
3. U.S. Code of Federal Regulations Title 21, Part 211 (Revised as of April 1, 2002), *Good Manufacturing Practices for Finished Pharmaceuticals*, 211.198. http://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfcfr/CFRSearch.cfm?FR = 211.198
4. U.S. Code of Federal Regulations Title 21, Part 11 (Revised as of April 1, 2002), *Electronic Records and Signatures*. http://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfcfr/CFRSearch.cfm?CFRPart=11&showFR = 1.
5. Hambloch, H. (1994), Existing Computer Systems: A Practical Approach to Retrospective Validation, in *Good Computer Validation Practices: Common Sense Implementation* (Eds. T. Stokes, R.C. Branning, K.G. Chapman, H. Hambloch, and A.J. Trill), Interpharm Press, Buffalo Grove, IL.
6. *Summit-D Handbook* (1994), Version 2.6. Coopers & Lybrand Associates Ltd. (now IBM).
7. FDA (1987), *Software Development Activities, Technical Report, Reference Materials, and Training Aids for Investigators*, Food and Drug Administration, Rockville, MD.

29 Case Study 11: Integrated Applications

Arthur D. Perez, Novartis

CONTENTS

Computer systems are becoming increasingly sophisticated and more integrated, making the application of maximum validation rigor to the entire system impracticable. By the same token, applying a philosophy of “validation lite” to the entire system can severely dilute the value of the validation, both from the standpoint of the business value of the validation exercise and from the perspective of regulatory compliance. These considerations point to a need to optimize the efficiency of validation processes and development of appropriate validation strategies.

RISK ASSESSMENT

Risk assessment is a tool that can aid firms striving for this goal, helping to focus validation effort where it is needed most, i.e., on functions and processes with the either the highest chance or the least palatable consequences of failure. While this alone should be enough to convince firms to adopt a risk-based validation strategy, there is added impetus provided by the fact that basing validation on risk factors is a regulatory expectation, as illustrated by these two FDA citations:

- May 1996 FDA 483: “Failure to identify and analyze the system/software critical functions. *No documented risk assessment and hazard analysis* was done …”
- November 1997 Warning Letter: “The software test plan currently in use included *no description of how test cases were developed* or how thorough test coverage is to be achieved.”

It is significant that the FDA has itself adopted a risk-based strategy for inspections in order to concentrate resources and effort where it provides the most benefit[1] — a clear indication that this is a philosophy acceptable to it.

There are many places during a validation project where it is appropriate to use risk assessment as a basis for key decisions. This case study will follow a hypothetical implementation of an integrated chromatography data system (CDS) in a Quality Control laboratory (see Figure 29.1). In accordance with the documented user requirements, this system will:

- Run on Windows NT 4 (Service Pack 6; the current version installed is Service Pack 5)
- Run on the corporate LAN
- Control *new* HPLC (High Performance Liquid Chromatography) and legacy GC (Gas Chromatography) equipment
- Employ some customized software elements
  - Integration algorithms tailored to known elution characteristics of one of the company's major products
  - A home-grown interface to the corporate LIMS
- Generate reports to Microsoft Office products (Word and/or Excel) for further processing

High-Level Risk Assessment

The first risk assessment that needs to be done when implementing a new system is one that may often be skipped because the result often seems quite obvious; this is the decision as to whether or not the computer system requires validation. However, leaving this assessment out can lead to regulatory liability in regard to systems that a firm decides do not need validation, so it is advisable to make this assessment a standard expectation within the company's system development methodology. It is not an onerous expectation, as it usually takes only a few minutes to document the decision properly.

Typically the validation determination can be made based on the answers to seven questions centered on the general nature of the system.

Figure 29.2 shows how a high-level risk assessment may be documented (see Chapter 6 for Validation Determination Statement). The seven questions cover virtually every contingency that could necessitate validation. A yes answer to any of the questions indicates that the system requires validation. The integrated chromatography data system used as an example in this case study clearly meets a regulatory documentation expectation and impacts release decisions, and thus must be validated.

Some firms may choose to apply validation requirements to non-GxP systems for other reasons, e.g., management of controlled substances, critical pollution control systems, or other legal- or business-critical systems; if so, criteria can be added describing those risk conditions.

SUPPLIER SELECTION RISK ASSESSMENT

A risk assessment is appropriate at the point of selecting a supplier for a system as well. Some aspects of risk only apply to the decision as to whether to patronize a particular supplier, such as price or potential return on investment, but other points on which a supplier is evaluated clearly represent risk factors that should be considered in formulating a validation strategy. Such elements include, but may not be limited to:

- **The state of the supplier's Quality Management System:** Although it is a tenet that quality must be built in and not tested in, if the deficiencies in the supplier's processes are not egregious, they may be mitigable through increased testing, or possibly even by adjusting the criteria for later risk assessments.

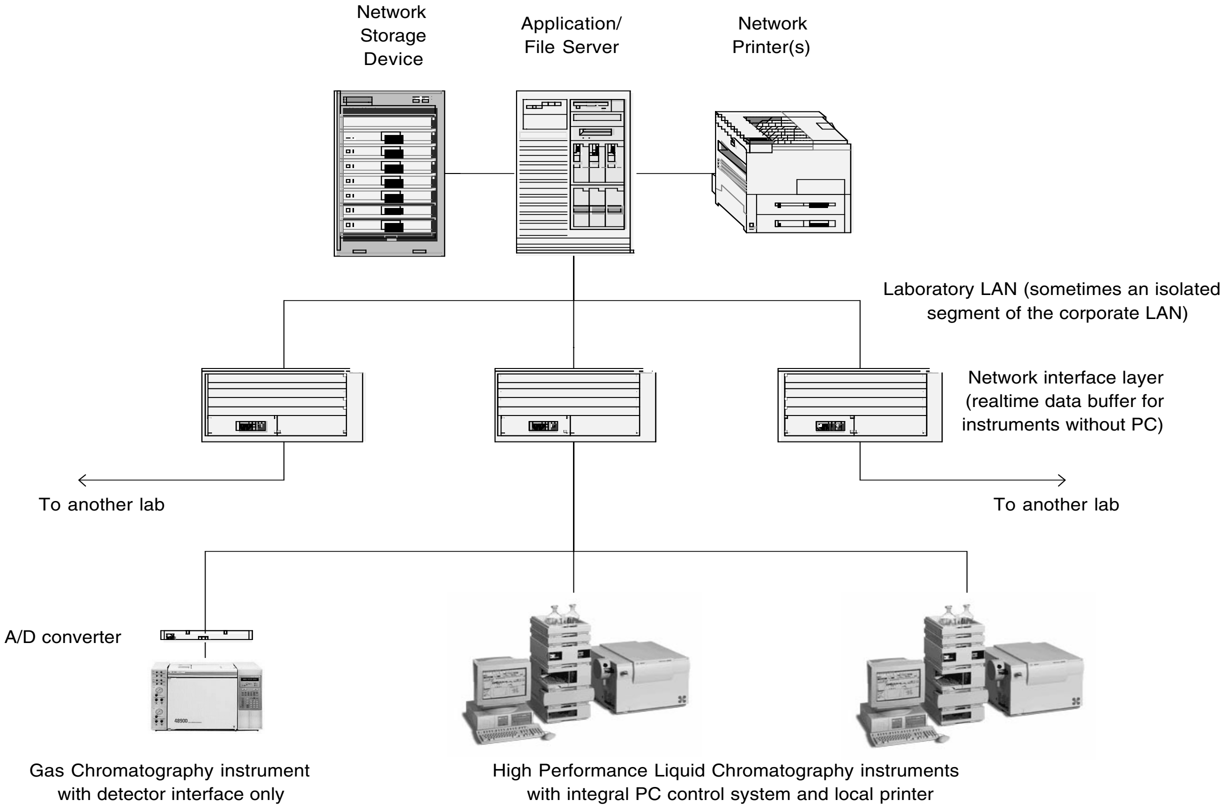

Network Storage Device
Application/ File Server
Network Printer(s)
Laboratory LAN (sometimes an isolated segment of the corporate LAN)
Network interface layer (realtime data buffer for instruments without PC)
To another lab
To another lab
A/D converter
Gas Chromatography instrument with detector interface only
High Performance Liquid Chromatography instruments with integral PC control system and local printer

FIGURE 29.1 A Potential Configuration for a Chromatography Data System.

GMP / GLP / GCP requirements		Yes	No
1.	Is the system used to produce, manipulate, or store data that may be used in any documentation required by a drug regulator, e.g., production records or drug regulatory submissions?	□	□
2.	Is the system involved in the manufacture, control, or release of pharmaceutical products?	□	□
3.	Is the system used in the collection, analysis or storage of data from clinical trials?	□	□
4.	Is the system used to control or monitor the environment in a production area, finished goods or raw material warehouse, or a research animal care facility?	□	□
5.	Is the system used to provide distribution information in the event of a commercial product recall, or in patient follow-up in clinical trials?	□	□
6.	Is the system vital to the exercise of statutory responsibilities, such as adverse drug event reporting?	□	□
7.	Is the system part of a process liable to regulatory audits (e.g., FDA, EU, or PIC GxP)?	□	□

FIGURE 29.2 Sample Questions for High-Level Risk Assessment.

- **The quality of the supplier's testing:** If it is poor, it is probably appropriate to compensate by strengthening the validation testing done by the client firm, but if it is of high quality, it may be possible to leverage some of their testing and reduce validation testing.
- **The supplier's customer support mechanism:** Most firms depend to a degree on the supplier to support the computerized system after it has been placed in production, and this is likely to impact the maintenance of the system's validated state.
- **The financial viability of the supplier:** If the supplier is providing support, it would be to the validating firm's advantage to be sure that the support will not disappear at an inopportune point.

An important consideration regarding supplier assessment is whether more than one audit is sufficient to develop the validation strategy. In the integrated CDS example, the use of customized software needs to be assessed. If custom coding is to be done by the supplier of the data system, then a second audit will be needed if the software is being developed at a different site (or by a different group) using different procedures. If a different group or site under the same procedures as were used by the data system developers is developing custom software, it is likely that evidence of adequate compliance may be provided by a postal audit.

If the custom software development is contracted to a third party, then a full audit is highly advisable. If the development is being done by the validating firm's own IT department, there should be some mechanism (normally internal audit) to ensure that internal processes are adequate. If development is being done by a third party under *direct* supervision of internal project managers and according to internal policy and procedures, an audit is probably unnecessary.

SOFTWARE RISK ASSESSMENT

The GAMP Categories

Computerized systems requiring validation are generally composed of multiple elements of varying risk. It is usually the case that validators can take advantage of this multifaceted character by determining those components of the system which are by their nature more or less risky. A useful classification mechanism has been provided in GAMP 4,[2] along with guidelines for testing each type of element. The principle tenet of the GAMP categorization is that basic assumptions can be made about the reliability, and hence risk, of software and hardware based on the nature of the

TABLE 29.1
Validation Testing Requirements Associated with GAMP Categories for Classification of Software

Software Categories	GAMP Guidelines for Validation Strategy
1. Operating Systems	• Not subject to specific validation challenges • Installation qualification requires evidence of correct loading, including who did it and when, and a record of the version
2. Firmware	• May require configuration • Installation qualification verifies name, version, configuration, or calibration • Test functionality against user requirements and functional specifications in operation qualification • May need supplier audit for critical applications
3. "Standard" Packages	• Installation qualification verifies name and version • Test functionality against user requirements and functional specifications in operation qualification • May need supplier audit for critical applications
4. Configurable Software	• Requires full life-cycle approach to validation (installation, operation, and performance qualifications) • Define strategies for mitigating supplier weaknesses or exploiting supplier strengths • Address layered software • Supplier audit normally required
5. Custom (Bespoke) Software	• Requires full life-cycle approach to validation (installation, operation, and performance qualifications) • Define strategies for mitigating supplier weaknesses or exploiting supplier strengths • Address layered software • Supplier audit normally required • Account for higher risk of "one-off" software (no proof of function in the market)
Special Software Cases • Spreadsheets • Software Development and Diagnostic Tools	• High-level validation approach is dependent upon how these types of tools are used by the application

system or subsystem. The GAMP classifications for software, along with general guidelines for testing, are shown in Table 29.1; hardware classification is shown in Table 29.2.

Classification of Hardware

Analyzing the example system against the GAMP categories identifies four of the five software categories represented, as well as one of the special cases. This breakdown is summarized in Table 29.3.

Analyzing the system in this fashion provides us with justification for a strategy that may seem intuitive to experienced validation professionals, but nonetheless should be documented. The bulk of the work is concentrated on the more complex and inherently riskier Category 4 and Category 5 elements. No resources need be dedicated to testing the Category 1 OS functionality; that challenge occurs when the higher level software is tested. By breaking out the Category 2 (firmware controlled) components of the HPLCs, their functionality can be challenged off-line from the data system's control functions. This tactic simplifies the testing process (and if the firm already possesses identical equipment an even greater benefit may be accrued by referencing prior validation work). A further benefit is that any problems controlling these components using the data system functionality will be readily attributed to that software, since the firmware will be a known quantity.

TABLE 29.2
Validation Testing Requirements Associated with GAMP Categories for Classification of Hardware

Hardware Categories	GAMP Guidelines for Validation Strategy
1. Standard components	• Document manufacturer/supplier details • Installation qualification verifies installation and connections • Record model, version, and serial number of preassembled hardware • Can use hardware data sheet or other specification • Challenge hardware during SW OQ/PQ as necessary
2. Custom (bespoke) hardware components	• All requirements for category 1, plus … • Design specification required • Subject to acceptance testing • Supplier audit for hardware development • Assembled systems from different sources require verification of compatibility • Configuration defined in design documents

TABLE 29.3
Example System Elements According to GAMP Classification

Category	Element of Example Integrated Chromatography Data System
1. Operating Systems	• Windows NT® Service Pack 6
2. Firmware	• HPLC components, e.g., column heaters, pump controllers, autoinjectors
3. Configurable Software	• Chromatography data system package
4. Custom (Bespoke) Software	• Customized integration algorithms, customized interface to corporate LIMS
Special case	• Reports exported to spreadsheet files

The data to be exported to spreadsheets present a unique element, since the validation approach will vary greatly dependent upon how the spreadsheets are intended to be used. There are essentially three levels at which spreadsheets must be evaluated:

- **Level 1:** If the data is to be manipulated using spreadsheet macros, the *spreadsheet application* must be considered as a GAMP Category 5 component. Macros in Microsoft Excel® are Visual Basic® computer programs written specifically for that application. While they may be simpler than the other customized elements of the integrated CDS, they still carry the same liabilities.
- **Level 2:** If the data is to be manipulated using only the native calculation functions of the spreadsheet, the validation approach can resemble that used for Category 3 software. All calculations should be verified and challenged through documented IQ/OQ testing. Particular attention needs to be paid to logic functions (IF, AND, OR, etc.), lookup tables, or database functions, as these are common error points. Boundary testing should also be done, as a remarkably common error is improper use of < vs. ≤ or > vs. ≥.
- **Level 3:** If the spreadsheet is to be retained as a data repository (this could apply to the previously discussed cases as well), then it will almost assuredly be an electronic record and must therefore comply with 21 CFR 11 or any similar regulations governing the use and retention of electronic data.

TABLE 29.4
Impact Analysis for Incorrect Sample Size Error

Risk	Possible Undesired Result	Possible Outcome
Sample too small	Impurities exist but are below detection threshold	Firm releases adulterated product to market
Sample too large	1. Integration inaccurate due to column overload 2. Impurities masked due to peak spread	1. Good product rejected 2. Firm releases adulterated product to market

In the integrated CDS example, data will only be manipulated using native spreadsheet functions, and the resulting record must be retained to comply with a GMP predicate rule. Thus Levels 2 and 3 requirements for spreadsheet applications must be met.

Function Risk Assessment

Using the GAMP categories as broad risk indicators has enabled the validation strategy to focus considerably and to direct most attention toward certain selected components of the system. However, even concentrating on the data management and control system plus the customized features still leaves a lot of ground to cover in testing. Now the analysis shifts to determination of where there is significant risk and/or hazard associated with the functionality of these software elements.

Again, the GAMP Guide provides a potential mechanism for this, and one that is particularly elegant for its simplicity. The first step of this process is to identify the critical functionality that is to be assessed. This is can typically be done quite easily by following the process or data flow of the tasks being performed by the system as a whole, and then identifying the GxP-critical operations of the computer system. Each of these operations is then analyzed to determine possible risk scenarios, and potential outcomes resulting from system failure are identified. For instance, in the integrated chromatography data system, assess the possibility of injecting the incorrect amount of substrate onto a chromatography column. There are two possible ways that this would be detrimental to the chromatographic analysis: either too much substrate or too little. Table 29.4 shows the impact analysis for this error.

The next step in the risk assessment process is to determine the *risk level* by cross indexing an evaluation of risk likelihood (as an estimate of probability and frequency) with an appraisal of the severity of the outcome. A three-by-three matrix is normally used, although there may be situations where it could be appropriate to look for either more or less granularity. In the integrated CDS example, rejecting good product is clearly a consequence to be avoided, but the release of adulterated product to the market is anathema. Assuming that the chromatographic analysis is the primary test of purity, it is a pretty safe statement to say that potential impact is high. (It might be lower if this were merely one in a battery of chemical tests that would call an anomalous result into question.) The likelihood of this occurrence, however, is probably quite low. It can be assumed that the accuracy of the autoinjector is good, based on the (presumably successful) functional testing of the Category 2 firmware that was done independent of the data system. Therefore any error in injection size will be the result of either a software bug in the application, or more probably the result of operator mistakes, e.g., programming an injection of 25 μl instead of 2.5 μl. The likelihood of this can be estimated as low. Using the GAMP matrix as shown in Figure 29.3, the risk level resulting from low likelihood and high impact is moderate.

The next step of the assessment process is to determine the probability that the error will be discovered before the consequences actually result. Detection probability is often dependent upon the business process, and most Quality Control laboratories have highly formalized review processes wherein all analytical results are assessed by responsible authorities before product is released. Ergo, in this case it is quite likely that injection of the incorrect size sample would be

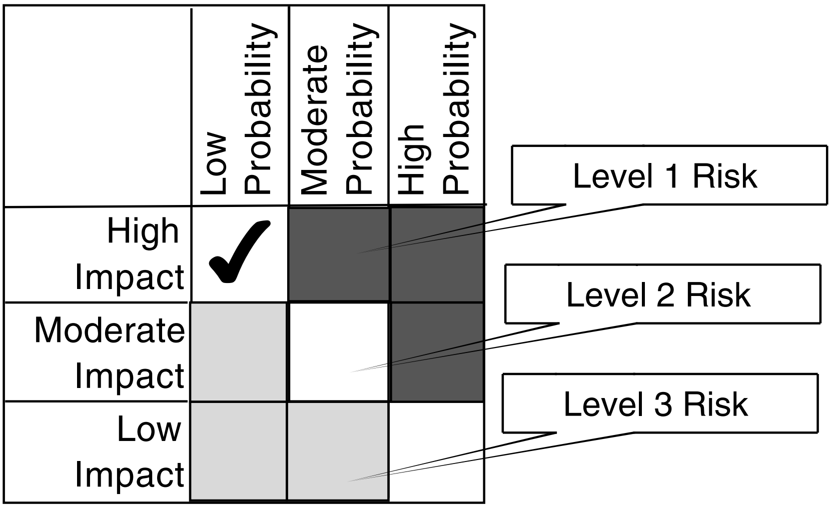


FIGURE 29.3 Risk Level Assessment for Chromatographic Injection Error.

detected. Cross-indexing the risk level classification derived in Figure 29.3 with the probability of detection using the chart shown in Figure 29.4, leads to the assignment of a low priority to inaccurate injection.

Several actions are available to mitigate or exploit the assessed risk priority. If the priority is deemed high, it might be necessary to redesign either the software or the business process to lower the risk. Had this been the case for injection volume, one possibility might have been to modify the software to include configurable error traps, perhaps rejecting input injection volumes outside of a configurable range. Another possibility might be to revise the business process to require building a library of preapproved analytical methods, and only allowing lab analysts to run analyses using these processes. However, given that the risk priority is low, it may be reasonable to reduce the testing of this function, perhaps only a limited challenge of the boundary conditions for the operation.

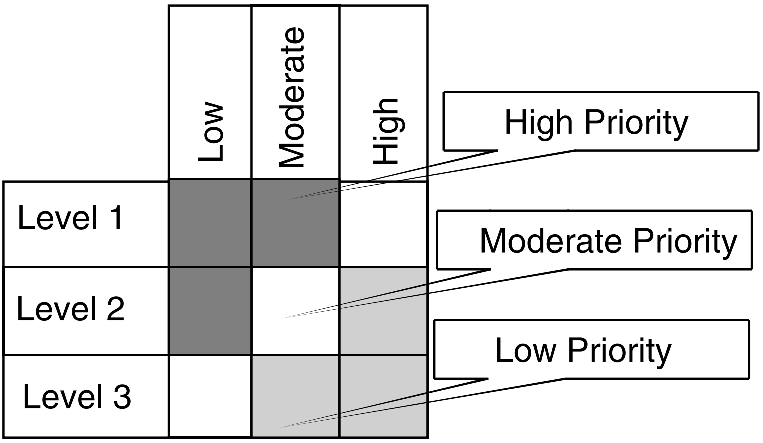


FIGURE 29.4 Risk Priority for Chromatographic Injection Error.

This is certainly not the only possible approach to a function risk assessment; others can provide equally valid analyses. The key to effective risk assessment is to apply criteria uniformly and to avoid fudging results in the name of convenience or intuition. From a regulatory compliance standpoint it is more important to retain the integrity of the process than to avoid a few tests that someone may feel are unnecessary.

COMBINING RISK ASSESSMENT INFORMATION INTO A VALIDATION STRATEGY

While it is unlikely that all of the risk assessments that go into an effective Validation Plan will have been done by the time the Validation Plan needs to be written and approved, the plan can still describe how all of the risk assessments discussed above will be used to determine an overall strategy for the validation.

Certainly the High-Level Risk Assessment that defines the need for validation will have been completed, and this should be appended to the Validation Plan.

It is always beneficial for a validation effort to have the supplier assessments done as early as possible, so ideally the strengths and weaknesses of involved suppliers should be known. If there are generic weaknesses, e.g., the supplier has generally poorly documented designs for all software modules, one approach to mitigating this risk might be to assume a generally stricter interpretation of risk level when doing the function risk assessment. If there are weaknesses that can be related to specific modules, it may be appropriate to strengthen the test challenges applied to that module. If the supplier's support processes are weak, or if the supplier is on shaky financial ground, the validating firm should consider developing good internal support processes for the system, or at least contingencies for setting them up. Of course, if the supplier's processes are really uncontrolled, the validating firm should think long and hard about whether it is advisable to do any business at all with that supplier.

Conversely, if the supplier's Quality Management System is very well written and there is documented evidence that the supplier adheres to it, the definition of risk levels might be relaxed a bit in the Function Risk Assessment. If the supplier's own acceptance testing is well documented and thorough, the Validation Plan should document the intent to reference some of the supplier testing in lieu of internal tests. It is not necessary to specify exact tests to be used at this point; that can wait until the test plans are being prepared later on.

When defining the test cases, there should be direct traceability for each test case back to one or more Function Risk Assessment line items. This should include any test cases that are dependent upon testing executed by the supplier. If the validation team decides that a test case is needed where there is no risk assessment, one should be done. This traceability of test cases to risk assessment is an important part of being able to justify what is tested and the degree to which it is challenged.

Looking to the integrated CDS example, the high-level validation strategy will be thus:

- High-Level Risk Assessment determines that validation is required.
- Supplier audit(s) will be a guide to developing later risk assessment criteria. Ideally, some of the supplier's own testing will be of high enough quality to reference in lieu of some of the testing by the validating firm.
- Application of GAMP categories and Function Risk Assessment will help in determining the general testing approach:
  - **Category 1 (Windows NT® Service Pack upgrade):** Record version number and evidence of correct installation.
  - **Category 2 (firmware in HPLC equipment):** Record version numbers, execute functional testing of components off-line; if the firm has tested identical firmware in past validation efforts, reference that.

- **Category 4 (data system configurable software):** Base test case development on function risk assessments; leverage as much supplier testing as possible.
- **Category 5 (custom algorithms and interfaces):** Evaluate the need for supplier evaluation on developer of custom code. Base test case development on function risk assessments; account for lower reliability of custom code in defining risk criteria.

LATER APPLICATION OF RISK ASSESSMENT METHODOLOGY

Risk Assessment continues to play a key role in keeping a system in a validated state once it has become operational.

- It should be required that all change control processes include a risk assessment as part of the basis for deciding how, and how thoroughly, to test the change. The Function Risk Assessment is a good tool for this.
- Systems should be subjected to a periodic evaluation against current regulatory standards. Such an evaluation should include:
  - Assessment of whether corporate or regulatory expectations have evolved to the extent that the existing validation is not longer adequate.
  - Assessment of whether the cumulative level of change since the validation report was issued is such that change control testing is deemed to be inadequate; in other words, is the effect of many small changes greater than the sum of the parts?
  - Assessment of whether multiple addenda to original specification documentation have made it difficult for a reviewer to understand the true configuration of the system.
  - Some support decisions, such as how often data should be backed up, should be subjected to risk analysis.

CONCLUSION

Risk assessment is an important tool for maximizing the business benefit and regulatory compliance value of validation work. It is important to remember that the purpose of validation is not to satisfy regulators but rather to find problems or errors in computer systems before they are deployed and thus before they can become critical compliance issues. Risk assessments help validating firms to focus effort where it is needed to achieve this. An added benefit that is sure to warm the cockles of senior management's heart is that it is also more efficient, helping to minimize the need for internal resources.

ACKNOWLEDGMENTS

I would like to thank my wife Jeanne for proofreading my early work, and Guy Wingate for final editorial comments.

REFERENCES

1. "FDA Unveils New Initiative to Enhance Pharmaceutical Good Manufacturing Practices," *FDA News*, August 21, 2002.
2. GAMP Forum, *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), ISPE, 2001.

30 Case Study 12: Building Management Systems

John Andrews, KMI/PAREXEL

CONTENTS

The control systems associated with building environmental management, typically known as building management systems (BMS), has always presented a difficult challenge to those responsible for the validation. Whereas the equipment involved may well be straightforward to validate on their own, the control systems themselves have presented a more difficult challenge. This has been because cGMP and noncritical facilities are generally housed in the same building. The control systems therefore have generally been mixed, thus making it very difficult and expensive to validate. Segregating the control system between cGMP and noncritical is also very difficult because the air-handling equipment and other such equipment may be common to both facilities. This has led to the common generalization that "building management systems cannot be validated" — the common get-out clause; however, the regulators are not convinced.

BMS FUNCTIONALITY AND REGULATORY REQUIREMENTS

The objective of the BMS is to centralize the monitoring, operation, and management of a process area or unit. One of the criteria for installing such a system is that critical process and building environmental parameters, such as room pressures and temperatures, can be maintained and recorded. Another benefit of installing a BMS is that energy is used in a more efficient manner and costs are, thereby, reduced. In the process of meeting these objectives, the BMS has evolved from a simple relay and timer-based system into a fully integrated microprocessor-controlled system with many features such as environmental optimization, PID (proportional, integral and derivative control) with full data recording and archiving facilities. Figure 30.1 presents a typical BMS layout.

Environmental control in drug manufacturing facilities has drawn increased attention from the FDA and other regulatory authorities in the 1990s. Section 46 of the U.S. Code of Federal Regulation for Good Manufacturing Practice states that:[1]

> *a. Adequate ventilation shall be provided.*
>
> *b. Equipment for adequate control over air pressure, micro-organisms, dust, humidity and temperature shall be provided when appropriate for the manufacture, processing, packing, or holding of a drug product.*
>
> *c. Air filtration systems, including pre-filters and particulate matter air filters, shall be used on air supplies to production areas when appropriate.*

European GMP Directives and associated regulatory guidance have very similar expectations.[2]

RECENT INSPECTION FINDINGS

Recent Warning Letters issued by FDA to pharmaceutical manufacturers highlight some regulatory concerns:

1. Our inspection revealed that the … computer system is used to monitor temperature, conductivity, water pressure and time (hours) for replacement of … for the … system has not been validated. Additionally this system monitors the differential pressures between the aseptic core and surrounding areas. The … system, which has been in place since January 1998, has not been validated. [FDA Warning Letter, 1999]
2. The [BMS] program is run locally at production buildings to monitor and indicate alarm conditions in production areas for temperatures, humidity, and air pressure [BMS], and is used to perform these functions in production buildings for manufacturing operations in xxx, yyy, and zzz. Controls for [BMS] were evaluated in building XX (purification) in xxx with the following observations:
   a. The firm produced no approval documentation for [BMS] version upgrades, e.g.:
      1. No change control process was followed to upgrade [BMS] version changes from version 1.3 to version 2.0. No change request form approving this change was filled out.
      2. No change control process was followed to upgrade [BMS] version 2.0 to 3.1. No change request form approving this change was filled out.

   b. Configurations controls:

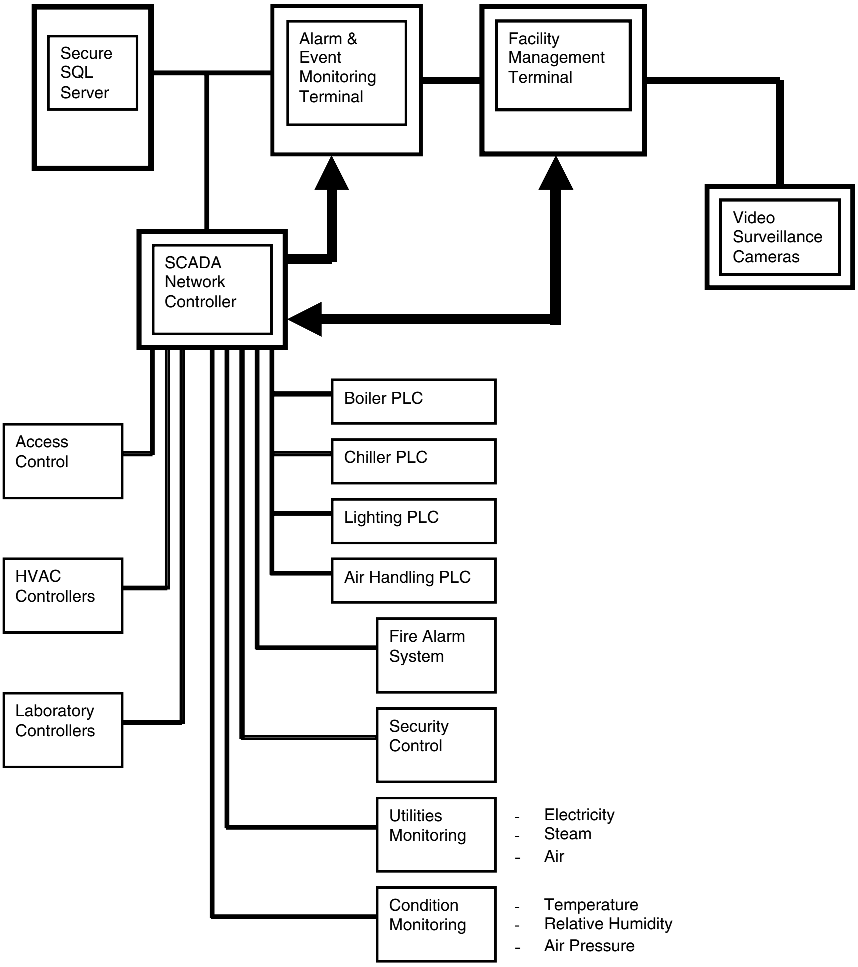


FIGURE 30.1 Typical BMS Layout.

1. The firm failed to evaluate setpoints following upgrade from [BMS] version 2.0 to version 3.1 upgrade.
2. Validation documentation fails to include printouts of setpoints from the [BMS] program for historical or current configurations.
3. The firm has failed to procedurally define setpoints settings for the [BMS] program.

c. Security issues for the [BMS] program:

1. The firm has failed to put in place procedures for periodic review of users/users level of access to the [BMS] program.
2. Evaluation of the users currently with access to the [BMS] program in building XXXX found one user who was not on the list presented as the currently recognized list of users with access to the [BMS] program.

[FDA 483 Observation, 1999]

3. The unit used to compare the computer line's pressure measurement readings with equipment air pressure measurements has not been calibrated. Additionally there has been no periodic maintenance to assure that the unit is operating appropriately. The issue becomes even more critical because the … computer system is not validated. It is essential that this unit … be accurate and reliable. [FDA Warning Letter, 1999]

TABLE 30.1
BMS Compliance Strategy

BMS Implementation	Quality Dependency	Compliance Strategy
Aseptic Manufacturing	Critical product quality dependency on control and monitoring of BMS regardless of any independent monitoring	*New BMS:* Expectation is to validate entirely new BMS implementations* *Existing BMS:* Qualification of existing BMS should be reviewed and revised as necessary; implement and validate independent monitoring; note that independent monitoring is not needed if BMS is validated*
Nonaseptic Manufacturing	Product quality dependency on monitoring	Implement and validate independent monitoring, review as necessary for existing BMS; adopt good engineering practice for BMS

* Unlike a few years ago, validatable BMS are now available as Commercial Off-The-Shelf products.

4. The alarm system that communicates, records, and controls alarms such as air balance and temperatures for production, warehouse, and testing areas lacked validation documentation. [FDA Warning Letter, 2001]

BMS COMPLIANCE STRATEGY

There has been some debate on the appropriate Compliance Strategy for Building Management Systems (BMS). Regulatory guidance has suggested that BMS, used to control the environment for aseptic manufacturing, have a critical impact on drug quality and should be validated.[3] Draft ISPE Baseline Guidance further suggests that BMS applications with indirect impact on drug quality do not require validation and that documented Good Engineering Practice (GEP) is sufficient.[4] This requires that there is no critical product quality dependency on the BMS and that qualified/validated independent monitoring systems are performing those functions critical to making decisions about the quality of product. Table 30.1 summarizes a suggested way forward.

ASEPTIC MANUFACTURING BMS

Product quality is critically affected by BMS control and monitoring for aseptic manufacturing such as parenterals. Reliance on alarming out-of-specification environment conditions is insufficient to support high integrity product. Independent monitoring does not relieve the basic reliance on BMS operability and should therefore be validated.

NONASEPTIC MANUFACTURING BMS

Product quality is dependent on monitoring the manufacturing environmental conditions; there is no critical product quality dependency on BMS environmental control. This scenario allows the implementation of validated independent monitoring.[4] Independent monitoring systems can be complex or simple depending on monitoring requirements. Highly toxic, terminal steriles and inhalation manufacturing often have sophisticated monitoring requirements involving multiple environmental parameters. Independent monitoring in these cases is best served by implementing a Supervisory Control And Data Acquisition (SCADA) system. Oral dosage, liquid, and topical manufacturing which have much simpler monitoring requirements are probably best served by stand-alone chart recorders.

APPROACH TO VALIDATION

Fear of FDA intervention certainly is a compelling reason for a company to validate its environmental controls. Accomplishing business goals may be a better reason. According to the Landis Division of Siemens Building Technologies, Inc.,[5] "It just makes good business sense to make sure the facility operates as designed to ensure quality products are consistently produced." The Operations Manager for Siemens explains it this way. "Aside from the risk to the life and health of employees, the cost of product failure due to not meeting quality standards can be very high. Years ago, humidity, pressure, and temperature were not considered part of quality control. Today, it is realized that controlling the environment boosts the production yield. It's not just the process that must be validated."

However, is validation still required for everything? If more than one building is to be constructed, all processes that must be validated by GLP (Good Laboratory Practice) or cGMP could be segregated to the same building and noncritical facilities housed in the other. If critical and noncritical areas are mixed within the building, the critical processes could be segregated to one area. Do offices, research and development labs, storage areas, and corridors really need to be validated? It may not be considered necessary. Finally, are all the hardware components critical (some may well have direct impact on quality whereas others may have indirect or nonimpact in the way that interacts with the process/product)?

Hardware and software change control must be addressed early on because it will affect the entire process. If thermistors are specified and then sealed behind drywall during construction, calibration will be a very expensive and time-consuming process (they must be replaced when they are out of specification). RTDs (Resistance Temperature Detectors), which can be calibrated in place and have field-replaceable parts, may be a more cost-effective solution in the long run, even though the initial cost is higher. If the software change control procedure requires revalidation with every minor modification, updates will be very difficult and costly. One should remember that the maintenance staff must live with the change control procedures for the life of the facility. Flexibility should be built in and subcontractors must be trained on the correct maintenance procedures.

RISK ASSESSMENT

The compliance requirements for BMS systems should be commensurate with how they are used. A risk assessment can be performed to determine whether the parameters controlled and monitored by a BMS application have direct or indirect impact on drug product quality (processing, storage, and distribution). Should the assessment reveal that the BMS is controlling and monitoring any parameters with a direct impact on product quality, there are two alternative courses of action:

- Validate the BMS
- Relieve the BMS system of its critical function

The risk assessment process can be divided into two steps. The first step would be to evaluate the impact of a system on the product quality. The second step of the process would be to evaluate the criticality of the components in the Direct and Indirect Impact systems, as they relate to product quality.

The determination of system impact as direct or indirect, and the result of criticality assessment, should be documented. Review and approval by Quality Assurance personnel are expected.

The application of this process helps to ensure that, first, if the validation route is chosen, the appropriate resources are applied to the parts of the system that have the potential to affect product quality. Second, it provides the rationale to focus qualification effort on quality-related functionality while still ensuring compliance for the product(s).

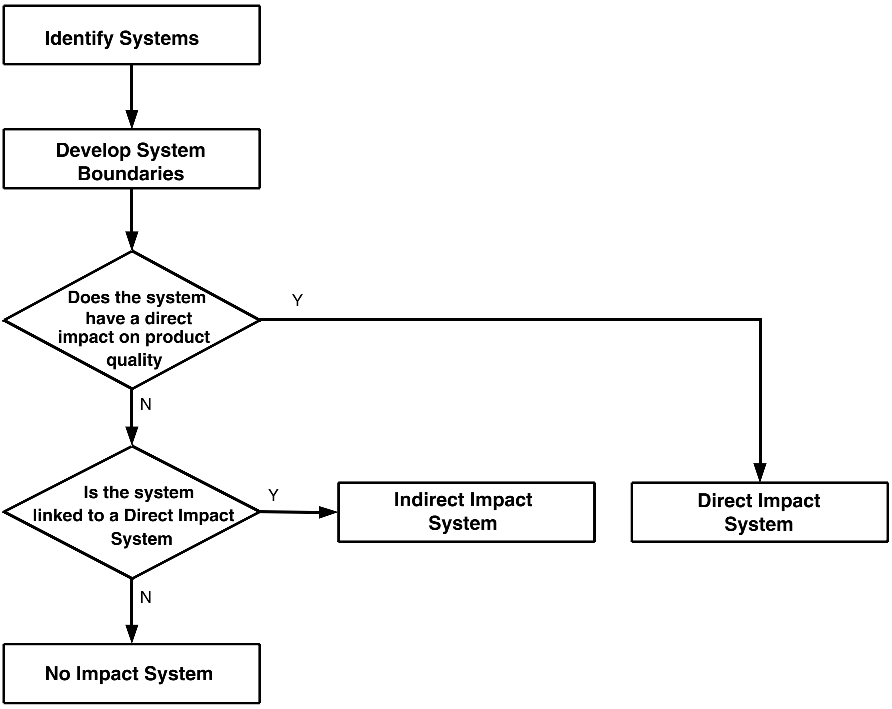

Identify Systems
Develop System Boundaries
Does the system have a direct impact on product quality
Y
N
Is the system linked to a Direct Impact System
Y
N
Indirect Impact System
Direct Impact System
No Impact System

FIGURE 30.2 Impact Decision Tree.

Impact Assessment

Review the system within the project, define the boundaries, and perform the system assessment outlined in Figure 30.2. The systems should be identified as:

- Direct Impact
- Indirect Impact
- No Impact

The No Impact systems fall out into a classification where it makes good business sense to apply Good Engineering Practices. The Direct Impact and Indirect Impact systems enter a preparation phase where the component lists of the systems are prepared for the second step of the process if the validation option is chosen.

Component Criticality Assessment

The second step of the risk assessment involves the criticality assessment of the system components. The components will be either critical or noncritical.

It may be easier to begin by creating a list of all the instruments, equipment, components, etc., in the Direct and Indirect Impact systems to perform the criticality assessment. A series of questions and discussions may take place to evaluate each component and its associated control requirements. A checklist may be used to track whether a component and its control features is critical or not during this process. Considering that there may be significant numbers of components and functionality, the method of documenting the process should be determined in advance.

Table 30.2 includes example questions to help determine component criticality. Specific products may require additional considerations.

TABLE 30.2
Example Criticality Questions

1. Is the component used to demonstrate compliance with the registered process?
2. Does normal operation or control of the component have a direct effect on product quality?
3. Will failure or alarm of the component or the associated functionality have a direct effect on product quality or efficacy?
4. Is information from this component recorded as part of the batch record, lot release data, or other GMP documentation?
5. Does the component (e.g., sensor) come into contact with product or product components?
6. Does the component control critical-process elements in such a way as to affect product quality without independent verification of the control system performance?
7. Is the component and associated functionality used to create or preserve a critical status of a system?

Component Criticality vs. System Impact

The results of the criticality assessment may then be checked against the matrix shown below. This matrix represents the relationships between systems and system components. Components are permitted to exist in three of the four boxes, and cannot exist in the lower left box. The relationships and their interpretations need to be understood before progressing with the assessment process.

Figure 30.3 illustrates a summary of the impact assessment process described so far. Additional points to note are:

- Indirect impact or no impact systems should not contain any critical components or associated functionality.
- Direct impact systems may well have both critical and noncritical components and associated functionality. The noncritical components and associated functionality can be reviewed and tested with a lower level of scrutiny.
- “Design for Impact” reduces the scope of the components and functionality that are subject to a focused validation effort, allowing appropriate focus on the components and functionality presenting the greatest risk to produce quality.

“Design for Impact” is the term used to describe the practice of making conscious design decisions with respect to the impact that a system and its associated functionality have on quality.

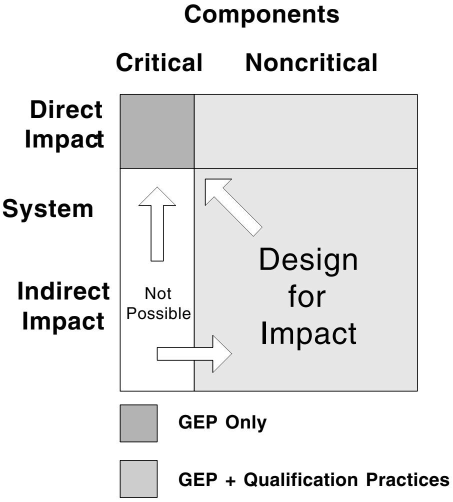


FIGURE 30.3 Criticality Determination.

By careful design, the number of direct impact components and functionality can be reduced, thus reducing any unnecessary qualification and validation efforts.

It is essential that this process is documented and approved (or at least reviewed) by QA. The process of assessment would normally be conduced by a team of qualified staff including representatives from Engineering, Production, and Quality. The documented conclusions will be used to position the validation approach that will be described in the Validation Plan.

Control System Considerations

Now that it has been established that BMS applications and their associated hardware should be designed in anticipation of the potential impact they may have on the quality of the product, the control system validation can be considered. Factors to consider include:

1. Are the hardware and software platforms supporting the BMS application suitable for validation?
2. Does the supplier implementing the BMS application have the capability to meet computer validation requirements? Where necessary, consider an alternative supplier.
3. Is there a need/capability to interface with legacy systems? Interfaced legacy systems should be validated where GxP data is passed to the BMS or to any independent monitoring system, including the network/data link.
4. Can the BMS application be subdivided such that a discrete part of the BMS can be applied to product critical areas?
5. Could a separate validated BMS be provided to product critical areas?
6. Is local support available at the level required for validation?
7. The degree of compliance in regard to the use of electronic records and signatures using technical and procedural controls (ref. 21 CFR Part 11, etc.).

The review should be formally documented and the system then made subject to ongoing change control. There must be no uncontrolled creep in the original quality assurance role of the BMS.

Independent Monitoring

GMP critical control input/output points typically in order of 5 to 10% of total input/output points. This has led many pharmaceutical manufacturers to consider the use of validated independent monitoring systems for the GMP critical control points and hence alleviate validation of the control system to a Good Engineering Practice (GEP) activity based on qualification.[4] Independent monitoring systems range in complexity:

- Chart Recorders are the simplest devices (0 to 30 input points), being industry standard (GAMP software Category 3). Validation requirements are based on recording model and version numbers, complemented by the necessary calibration and commissioning of alarm signals.
- Data Loggers are more complex than Chart Recorders (managing typically 30 to 300 input points) and while industry standard systems are available, there is usually considerable configuration and bespoke programming. Validation is based on the combination of GAMP software Categories 1, 4, and 5, covering operating systems, configurable software packages, and bespoke programming. A complete life-cycle approach is therefore required, including archiving the raw data.
- SCADA systems are more complex again than PC-based Data Loggers (typically managing in excess of 300 input points). They may directly monitor or supervise a number of monitoring PLCs. The software has a similar character to that associated with Data

Loggers and the validation approach should be the same. The scale of the validation work will be greater, however, than Data Loggers because of the increased complexity of the system.

Independent monitoring systems used to implement the key quality assurance controls must be validated (whether they are complex Supervisory Control and Data Acquisition (SCADA) systems, or simple chart recorders). For an independent system to be accepted as a validated alternative in the monitoring of critical parameters, the system must be able to manage key quality assurance functions. Such functions include, but are not necessarily limited to:

- Controls for maintaining set-points
- Functions for alarms and alarm logs
- Functions for trending over short and long term
- Preventative maintenance including calibration
- Access controls for security purposes
- Data interpretation and management

In those applications where all key quality assurance functions are managed through independent monitoring systems, the provision and maintenance of the controlling system can be managed through GEP. It is important to remember, however, that to avoid validating the control system, the independent monitoring systems need to keep records of all critical drug manufacturing parameters, e.g., air changes, temperatures, air exchanges per hour, and pressure differentials. This data may be used to support batch records, regulatory submissions, or QA investigations for out of specification incidents. Regulatory requirements for electronic records should also not be forgotten.

Data Interpretation and Management

Any difference in monitored values between the validated independent system and GEP/qualified control system would act as a trigger for investigation, in advance of any routine calibration or performance check of the validated system. Decisions about product quality must be driven by data generated from the validated independent system where an independent system philosophy is followed. The data used to support these decisions must be archived.

Good Engineering Practice (GEP)

GEP is defined as established engineering methods and standards that are applied throughout the development and operational life of a system to deliver appropriate cost-effective solutions. As such, GEP consists of the following:

- Professional and competent project management (processes, procedures, and staff)
- Professional and competent engineering design, procurement, construction, and commissioning
- Full consideration of applicable statutory safety, health, and environmental requirements
- Full consideration of operation and maintenance requirements
- Full consideration of recognized industry standards and guidance
- Appropriate documentation for ongoing operation and maintenance, and to demonstrate compliance with applicable regulations and codes
- A formal system of change control is adopted

The above definition is from ISPE Baseline Guide; for a more detailed explanation refer to the ISPE Baseline Guide.[4]

VALIDATION LIFE CYCLE

Typically, a BMS is a mixture of software categories; it is important, therefore, that the Validation Plan identifies what are the categories of software which make up the system, as well as incorporates the results of the system and component risk assessment into the overall validation strategy. Another important element that feeds into the validation plan is the result of the supplier audit. The validation life cycle presented here is consistent with GAMP Guidance and related case study material.[6–8]

VALIDATION PLAN

As stated above, the validation plan is a crucial document. From experience, the best method to create the plan is to set up a small team, consisting of the user, system expert, and quality assurance representative. The plan will include the results of the risk and software category assessments as well as any additional requirements determined by the supplier audit. The plan will state what documents are required, when they will be produced (i.e., in what order), and by whom. The validation plan will state what must be done in order to confirm that a system will be validated.

SUPPLIER AUDIT

As the BMS supplier generally supplies to the construction industry, they have little experience with cGMP and the resulting requirements for validation. Therefore, it is essential that a Supplier Audit is performed. The advantages of a Supplier Audit are:

- Defines the appropriate software life-cycle method to be followed
- Enables gaps in existing management system and documentation to be addressed early in the project life cycle
- Builds relationship between client and supplier
- Clarifies uncertainties
- Educates supplier in customer-specific validation requirements
- Identifies what follow-up activities may be necessary

The key to the process is to understand the system that is being proposed. It is good practice for the auditor to spend time reviewing the User Requirement Specification and the system descriptions and understanding of what software categories exist for the proposed system. This should be followed up, with the postal audit checklist. This will also provide valuable information to enable the auditor to plan the audit. Available information should be used to customize the audit checklist to address the specific issues that are relevant to both the supplier and proposed project. Consider, for example, a system that includes hardware and software, where some of the software is custom, other parts are configurable and yet others are part of a standard package. The auditor will need to establish how each part of the system will be developed, and how the build phase will be controlled. There may even be more than one supplier. The auditor would need to split up the main elements and examine how each part of the system will be built.

A flow chart indicating the activities of the auditor or audit team is given in Figure 30.4. The following sample supplier questionnaire can be used as the basis of the supplier audit, postal audit, and audit checklist.

- Summary of product/service under audit.
- Is the supplier registered to ISO 9000, or TickIT? If so, which parts and when?
- Product and service development.
- Use of subcontract suppliers, etc.
- Contract reviews.

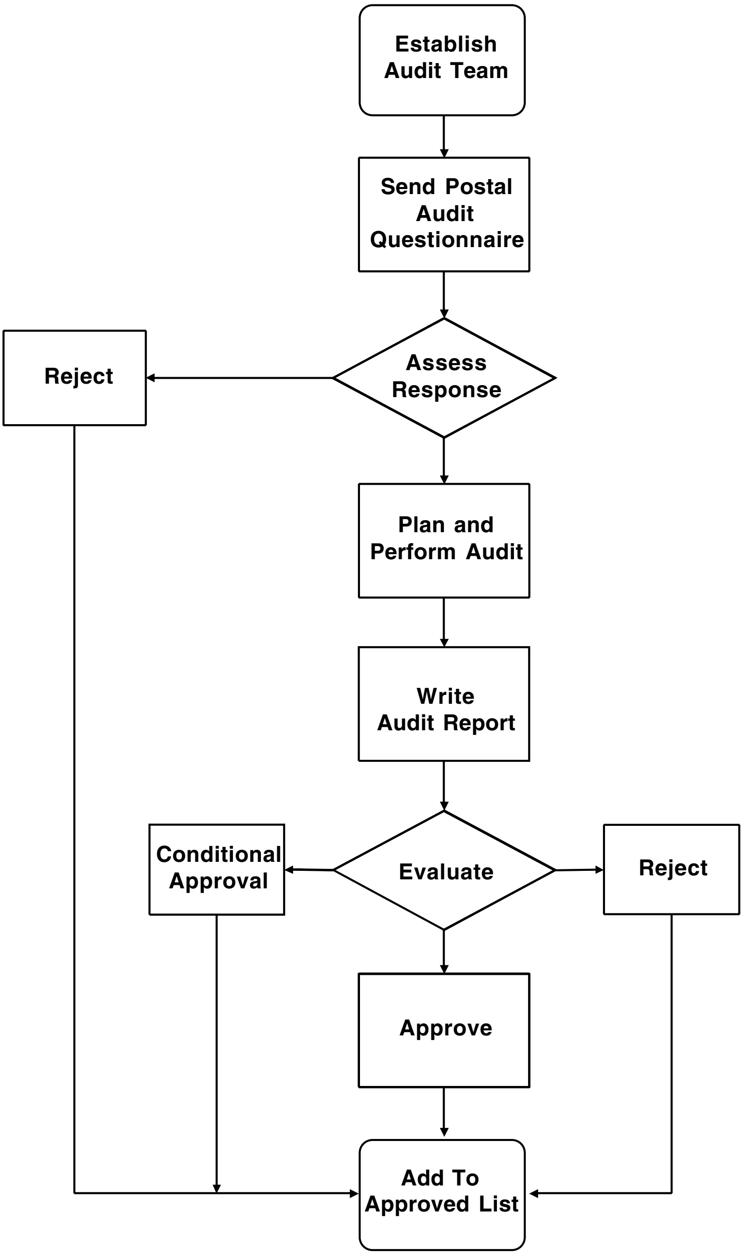


FIGURE 30.4 Audit Process.

- Specifications.
- Software life cycle method.
- Verification of purchased material.
- Testing, with deviation management.
- Change control for documents and software.
- Training.
- Support and maintenance.
- Support procedures and activities.
- Fault reporting.

The standard software packages and the custom elements of the system will require a similar review against the standard audit checklist. It is important to use the checklist as an aid to planning the audit, not to drive the audit. Remember, *if you fail to prepare — prepare to fail.*

When choosing a BMS supplier, look for experience in the validation process as a prerequisite. A close working relationship can save time and money beyond the initial cost of installation. A primary criterion for choosing a BMS supplier should be the ability to provide support for the life of the facility. Their attitude should not be one of walking away after commissioning.

System Specifications

System specification documents are required but often are not in the format that fits with the traditional GAMP "V-Model." In these circumstances, it is important that the user defines the required functionality in the URS, making sure to define the objectives and separation requirements for all quality-related areas, components, and associated functionality. However, do not be too specific with stating what the control limits are that must be achieved during the commissioning stage; better to specify some example limits and state that during the bedding-down period of the systems use, and during the subsequent qualification phases, these limits will be confirmed and documented. Continue to use the legacy system in parallel with the new system during the bedding-down period, if possible.

Remember, the user should get involved as early as possible and look at what the desired end result will be, not just the "correctness" of the specification. The URS is not always exactly what he wants, what he wants is not always what he gets, and what he asks for is not always what he needs.

It is not typical to receive a separate functional and design specification; these are often included as part of the building's environmental control package. If this proves to be the case, then it is not necessary to rewrite these specifications. The recommended approach is to create a matrix that references those parts of the environmental control package that are relevant to the computer system validation requirements for the critical functions. Also, ensure that this matrix is mapped back to the URS in order to ensure that the URS is met. This matrix will form the basis of the Requirement Traceability Matrix (RTM), which is intended to assure that all requirements have been addressed, that the functionality is appropriate, consistent, and meets predefined standards, and that the system is appropriately tested. The functional/design specification or the relevant sections of the environmental control package should typically address:

- Control and monitoring required for each specific area of the building; this should be in accordance with a predefined separation policy for critical areas vs. noncritical areas. It should also address the tolerances for control and monitoring accuracy.
- Field instrumentation requirements, which are impact vs. indirect and no-impact components covering:
    - Measurement range
    - Measurement accuracy
    - Control loop dynamics
    - Exposure to corrosion
    - Vibration
    - Hazardous area requirements, if any
    - Index of Protection (IP rating) for cleaning
    - Accessibility
    - Calibration requirements
    - Maintainability and spares
- Outstation locations and network requirements. These should be capable of overseeing the whole network with response times suitable for the process controlled. The presentation of data that takes account of the number of users on the system and the various levels of technical information required.

- Data and records production and how they are handled in meeting regulatory requirements (e.g., 21 CFR Part 11).
- Presentation of data may include the following:
  - Dynamic mimic displays
  - Graphical trend plots
  - Printed reports
  - Data processing and how this is handled
  - Alarm set points and handling

Design and Development

The BMS functionality is constructed mainly from GAMP 4 software Category 4, standard system modules configured to the user's specific requirements; therefore, little software development is necessary. Care must be taken to ensure that the configuration work and any code testing is independently reviewed and documented. Another aspect to consider is how the "standard modules" are to be configured. A sample recompiling of source code elements may be required. Also, if there is a need for custom coding, this should be treated, as GAMP 4 software Category 5 and full development validation will be needed.

Some aspects of validation are unique to HVAC control systems. Although the controls are one of the last things to be fitted, they must not be planned last. The user must make many decisions before the controls are installed and there should be qualification meetings early in the process. Quality cannot be tested into a process. It has to be designed into each system.

System Build

Build is normally part of the construction phase. It should be remembered that, generally speaking, standard components from the suppliers preferred range will be used so that no special build is required, except for purpose-built marshaling and display cabinets that will require design and build drawings. Special care must be exercised when using intelligent instrumentation and the associated bus-type communication networks. The desired functionality should be documented and Design Qualification (DQ) on these components should be undertaken.

The HVAC controls for critical (validated) areas should be grouped in specified field panels. One may want to label these panels "Critical Process Controls: Please Follow Change Control Procedures" or something similar. This will prevent the necessity of having to validate noncritical controls. Electrical supplies and other utilities must also be evaluated. One may need a UPS (Uninterrupted Power Supply) for critical field panels and PC workstations to continuously monitor critical equipment such as refrigerators, incubators, and particle counters with the BMS.

Factory Acceptance Testing (FAT)

It is recommended that as much factory testing as possible should be carried out before delivery of the control system to site. To a large extent this will be limited to simulation of the input and output elements (i.e., it is not connected to the building services and instrumentation yet). Full testing can occur once installation is complete. The extent of FAT is governed by how rigorous the simulation can be designed. Remember it is worth investing time at this stage to fully challenge the system in order to reduce the time and effort if faults are found, once the system is installed on site.

On-Site Testing

After the HVAC mechanical equipment and controls are installed, the process should begin with a point-to-point checkout of every component (i.e., verifying that every input and output device is connected to the proper terminals). If formalized, this method would reduce cost and time by utilization the commissioning documentation to support validation. For example, commissioning checklists can be referenced in the Installation Qualification (IQ). The alternative is to do them separately and duplicate a lot of paperwork. If calibration is required, the procedures and documentation must be referenced in the validation protocols. Once IQ is satisfactorily completed, start-up of the HVAC system can begin in accordance with the company's SOPs. The mechanical equipment must be up and running before Operational Qualification (OQ) can begin. This is where verification is done to ensure that the various mechanisms operate as intended (e.g., when the room thermostat calls for heat, does the hot water/steam valve open?).

Performance Qualification (PQ) must be carried out by the user. This is where verification is done to ensure that all systems work together under as-used conditions to meet the User Requirement Specification. Do room temperature, humidity, and pressure stay in spec with production underway and people entering and leaving the facility? All systems must be operational to complete PQ. Cooperation between the various contractors (mechanical, controls, etc.) is vital to completing PQ in a timely and cost-effective manner. As discussed earlier the user and the designer must sit down at the beginning of the project and determine critical (validated) and noncritical areas. Do not waste resources and money validating noncritical areas.

Maintenance

Change control procedures should address such issues as scheduling and documentation of maintenance and recertification of calibrated sensors. How will one ensure that a calibrated sensor is available if one fails or that the control program changes stick to standard formats? This is the nature of BMS change control. The following quote from the Proposed Changes file of the cGMP Web site emphasizes the FDA's viewpoint: "To preserve the validated status of a process, measures must be taken that will allow any significant process changes to be recognized and addressed promptly. Such change control measures can apply to equipment, SOPs, manufacturing instructions, environmental conditions, or any other aspect of the process system that has an effect on its state of control and therefore on the state of validation."

An auditor must be able to evaluate the current status of a facility based on the owner's documentation and compare it to the specifications, but the processes also have to work smoothly and allow improvement.

Reporting

The validation plan should require an assessment of the project success, and the validation report should present, or refer to, the evidence to support this. It is normally the case with a BMS project that the documentation is too large to attach to the report. It is, therefore, sensible to present the documentation in a list form with the location of each document referenced. This report must also clearly state that the system is validated and is approved by the user and Quality Assurance (QA).

BENEFITS DELIVERY

A major pharmaceutical company installed a building manufacturing system (BMS) in one of its sterile powder vial filling manufacturing suites. This would have replaced an old building service control system with independent monitoring via manual readings from fixed gauges. The BMS would control and monitor manufacturing suites and preparation areas including changing rooms, service areas, offices, corridor, and refreshment room. The implementation followed in the strategy is described in this study.

The first step was to assess the system for impact, i.e., which parts are direct impact, indirect impact, and no impact. The purpose of this strategy was to decide how to apply separation of the control system and air-handling equipment for the different areas. It was clear from the assessment that the impact areas were the manufacturing suites and changing rooms and the indirect area was the preparation area. The no-impact areas consisted of the service areas, offices, and corridor, and refreshment room.

The design, therefore, called for two control systems and associated hardware, which would require validation to be applied to one system with the other following the principles of GEP. The next step was to decide on the extent of the validation for the quality critical system with direct and indirect impact.

A further assessment of the quality critical system components and associated functionality was then required. The resulting list of direct impact components (and associated functionality) covered temperature control, humidity, and pressure differential between manufacturing areas to changing rooms and preparation areas to offices, restrooms, and corridors. This allowed for validation challenge testing of critical functions associated with the impact components and associated functionality. It was followed by validation confirmation testing of all other BMS control functions for indirect impact components (and associated functionality) on the BMS controlling the preparation areas, including changing rooms.

The principle of GEP for the BMS system controlling offices, restrooms, and corridors was adopted. The project milestones were then planned and auctioned in accordance with the combined (cGMP/GEP) plan. The supplier was audited and commissioned with the understanding that it must participate in the risk assessments. It was agreed that savings in project costs would be shared; however, the company's QA audit group would assess the whole project and fines could be applied if breaches in quality were detected.

The project saved 40% of the original estimated validation effort, and the whole project was completed under budget. Risk assessment delivered real benefit while maintaining compliance.

ACKNOWLEDGMENTS

The author would like to acknowledge comments and contributions of Dr. Guy Wingate, GlaxoSmithKline.

REFERENCES

1. U.S. Code of Federal Regulations Title 21, Part 210, *Current Good Manufacturing Practices in Manufacturing, Processing, Packaging, or Holding of Drugs*; Part 211, *Current Good Manufacturing Practice for Finished Goods.*
2. European Union Guide to Directive 2001/83/EC, Computerised Systems, Annex 11 of Rules and Guidance for Pharmaceutical Manufacturers and Distributors.
3. Pharmaceutical Inspection Co-operation Scheme (2003), *Good Practices for Computerised Systems in Regulated GxP Environments*, Pharmaceutical Inspection Convention, PI 011-1, Geneva, August.
4. ISPE (2001), *Qualification and Commissioning Baseline® Guide*, First Edition, International Society for Pharmaceutical Engineering (www.ispe.org).
5. Landis, *Validating Building Control Systems*, Siemens Building Technologies Inc.
6. Wingate, G.A.S. (1997), *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice*, Interpharm Press, Buffalo Grove, IL.
7. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), International Society for Pharmaceutical Engineering (www.ispe.org).
8. GAMP Forum (2003), Validation of Process Control Systems, Good Practice Guide, International Society for Pharmaceutical Engineering.

31 Case Study 13: Engineering Management Systems

Chris Reid, Integrity Solutions Limited
Tony Richards, AstraZeneca

CONTENTS

ENGINEERING MANAGEMENT SYSTEMS (EMS)

Effective and efficient utilization of assets by pharmaceutical research or manufacturing organizations is fundamental to the early delivery of new products to market and to satisfying customer demand once those products have been approved for release by the relevant regulatory authorities. A carefully designed strategy is essential for optimizing and maintaining system reliability, capability, and performance consistency; that is, assets must:

- Be available when needed and must not fail during use
- Function consistently to predefined performance criteria
- Meet performance criteria without undue stress, risk of failure, or reduced asset life

The continuous improvement of asset reliability, consistency, and capability, either mutually or simultaneously, is the basic objective of the engineering management strategy in order to reduce operation and maintenance costs and increase regulatory compliance. The foundation for continuous improvement is information, without which it is impossible to establish a rationale for change. This foundation must be established at the start of the project with the definition of the business need in measurable terms, i.e., performance criteria, without which there is no basis for design, testing, operation, maintenance, compliance, and consequently continuous improvement.

Figure 31.1 defines a simple asset development and operational life cycle depicting the creation of critical asset management information at each phase. The information generated must be managed in order to facilitate structured access and controlled maintenance. Considering the volume of information supporting even modest-sized organizations, it is essential that an Information System strategy be developed to manage critical information.

Business Need

Business need must be clearly defined and understood before initiating an asset development project that could require major financial investment and commitment of valuable resource. A business case must be developed that defines the strategic fit of the development within current and future business plans; the short, medium, and long-term benefits of the development; and the payback on investment.

Map Processes

Once the business case has been accepted and investment received, it is necessary to define the process and functional requirements of the development.

A coordinated team of users, engineers, safety inspectors, and quality representatives will map the operations required to meet the business need: that is, the scientific research and development operations, as well as the production process or the goods in process. The processes are often presented in flow-diagram form supported by descriptive narratives to expand process definition where required. Interaction between processes must be clearly defined.

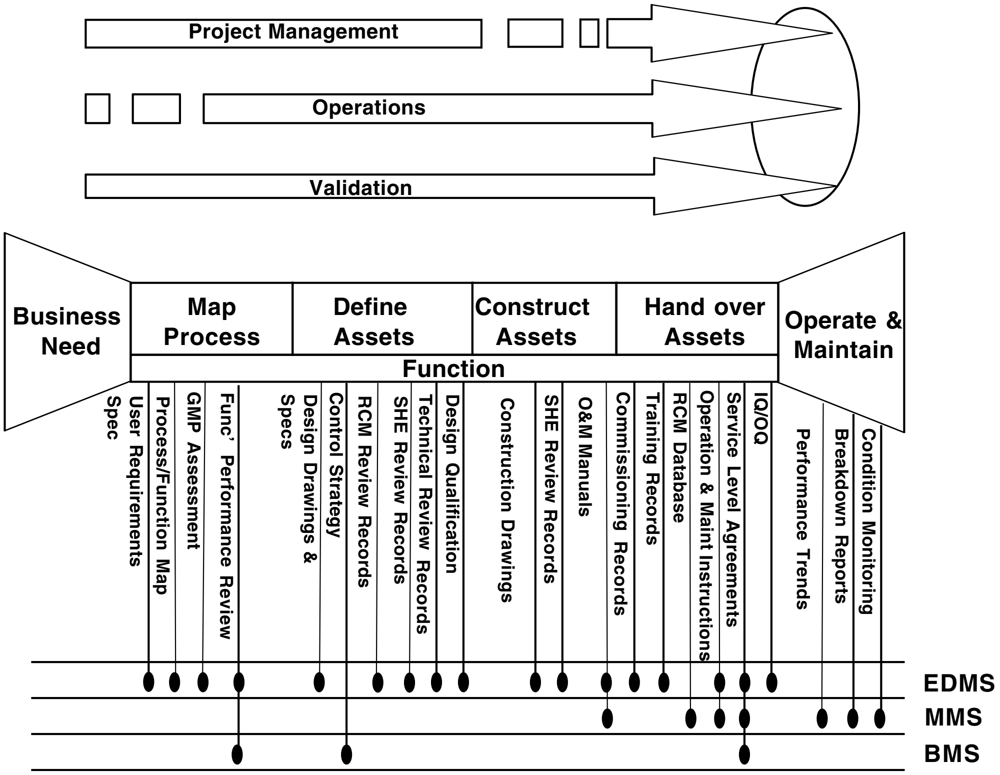


FIGURE 31.1 Asset Life Cycle.

Once processes have been established, the functions required to implement the processes shall be defined, e.g., equipment sterilization, environmental control such as temperature, differential pressure, particulate control, etc. It is at this stage that we must define the function performance criteria that will provide the basis for design, testing, operation, maintenance, and ultimately continuous improvement.

Function performance criteria must not be simply stated as discrete values that do not provide any degree of tolerance or that do not express the consequence of performance loss or interruption. Performance criteria stated as "Maintain room temperature at 18°C" are loose and ambiguous, and should be more accurately specified as:

Temperature Range:	18–22°C
Control Accuracy:	Set-point ± 1°C
Acceptable Excursions:	<5°C for less than 10 min

Processes and functional requirements must be reviewed before issue to the design consultant. The objective of the review, or in some cases multiple reviews, shall be to ensure that processes and functions have been completely and accurately defined, and that performance criteria are unambiguous. Reviews must also determine the consequence of function failure: that is, the risk to the research study, manufacturing process, and safety and regulatory compliance. These consequences must be documented so that the delivered solution is appropriate to the business risk, i.e., the design must be relevant to the operating context of the asset.

The output of this activity will be the User Requirement Specifications (URS) or Project Definition that will be issued to the design consultant and evolved into a detailed design defining the assets required to meet the requirements.

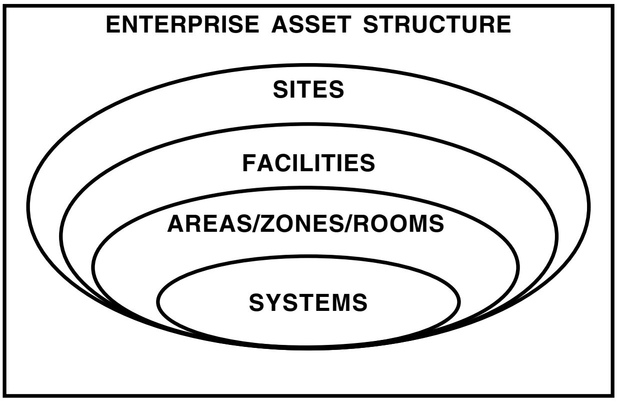


FIGURE 31.2 Asset Hierarchy.

Define Assets

Asset definition is a phased activity involving scheme, concept, and detailed design that will deliver the specifications, engineering drawings, databases, etc. that define the operational strategy and construction of the assets required to deliver the processes and functions defined within the URS. Typically, the assets of a pharmaceutical organization are managed as a hierarchy (see Figure 31.2) comprising:

- Sites
- Buildings
- Rooms/Areas/Zones
- Systems (Utilities, Building Services, Process Systems)

This hierarchical structure provides the foundation for information access: that is, the information search capabilities that enable rapid access of, say, the calibration records for the Heating, Ventilation, and Air Conditioning (HVAC) system controlling zone 1 within the biochemistry building.

This chapter focuses on systems as they are the most diverse and complex assets in the asset hierarchy and are the primary focus of continuous improvement strategies to improve system reliability, consistency, and capability, leading to operation and maintenance cost reduction and increased regulatory compliance.

System Concept

Engineers and users frequently refer to vessels, pumps, and valves which, although they are critical components, do not in isolation deliver the functionality required by the research or manufacturing process. It is the integration of such components into a system that enables the designed system performance to be delivered and maintained. The HVAC system delivers air to Class 10,000, temperature to 20 ± 2°C, and humidity to 50 ± 5% RH. The failure of a component, although important, becomes critical only if performance is lost. The design process must take account of the potential risk to the research or manufacturing process arising from the loss of performance and take remedial action to minimize such risk. Figure 31.3 differentiates between performance loss and system failure. GxP compliance is lost once the performance deviates from the predefined operating range, which is long before the system totally fails.

Essential information required for the design, operation, and maintenance of the system must, therefore, be specific to the system. For example, the provision of a master valve schedule listing all valves within a facility will provide the necessary information to maintain all valves for all

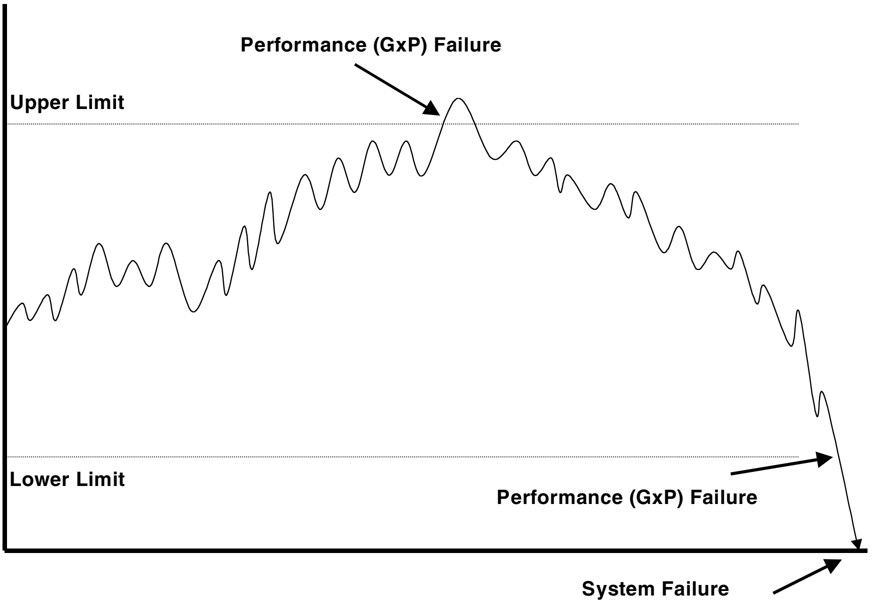


FIGURE 31.3 Performance Monitoring.

systems operating within the facility. However, retrieval of the information specific to a system that may have recently failed and that may be process or product critical will be cumbersome. The provision of system-specific valve schedules will enable more efficient information retrieval. A further advantage of focusing on systems is that information can be more easily provided that is relevant to the operating context of the system. For example, two systems may be providing similar functions; however, one may operate within a GxP environment and the other not. The level of information required to support the system operating within a GxP environment is significantly higher than is required for systems operating within a non-GxP-critical environment, e.g., materials specification for product contact parts, filter certificates, etc.

The relationship between the system and its component parts is synonymous with the relationship between information and data. Data in isolation is largely meaningless; however, when associated with other key data to create, say, asset failure reports that identify system function, function failure, failure mode, and consequence of failure, then a powerful basis for continuous improvement is established.

Systems and Functional Performance Criteria

Measurement of system performance is essential to asset management. If loose and ambiguous performance criteria are defined, it follows that the basis for design is poor and that the system is unlikely to meet the business need.

Design reviews will ensure that the proposed system can meet the performance criteria in a "reliable," "consistent," and "capable" manner. In addition to establishing a robust design that can be qualified against predefined performance criteria, it is essential to establish an asset management strategy that will effectively and efficiently maintain the system performance, reduce maintenance costs, and improve regulatory compliance.

Failure Mode and Effect Analysis (FMEA) is one tool that can be applied to challenge the design against the stated performance criteria and further to provide the foundation of the Asset Management Strategy to ensure that system performance is maintained (see Figure 31.4). The FMEA process defines:

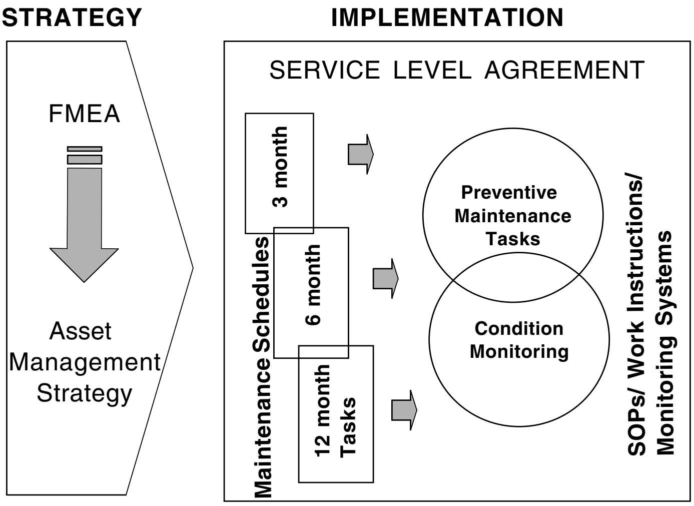


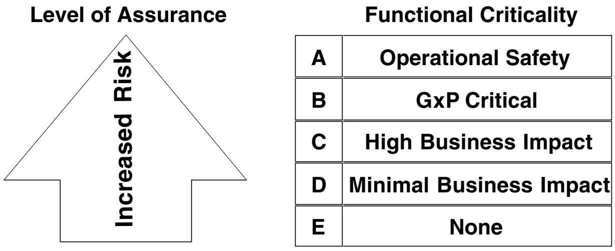
FIGURE 31.4 Asset Management Strategy.

- System functions (process requirements/objectives)
- Function failures (failure scenarios)
- Failure modes (reason for failure)
- Consequence of failures (impact on the business)

The output of the FMEA analysis is paramount to the determination of the business risk (consequence of function failure) presented by a system. This risk is used to determine the level of rigor applied to the validation, operational control, maintenance, and documentation/information needed to verify and maintain system performance as indicated by Figure 31.5. It follows that the documentation/information supporting system functions is as critical to the pharmaceutical organization as the system function itself.

The FMEA process is applied during design phase. The principles of FMEA may have already been used to review the processes and functions documented in the initial URS. At each phase, the outcome of the previous FMEA is refined. Each function of the system is challenged and assigned a criticality based on the consequence of functional failure. Table 31.1 provides a generic view of FMEA objectives at each stage of application.

The objective of this chapter is not to provide a detailed description of FMEA; however, it is clear that FMEA is a key ally of the validation process. Figure 31.6 identifies some of the key outputs of the FMEA process that support the validation.



Functional Criticality	
A	Operational Safety
B	GxP Critical
C	High Business Impact
D	Minimal Business Impact
E	None

FIGURE 31.5 Functional Criticality.

TABLE 31.1
Generic View of FMEA Objectives

Stage	System Evolution	Objectives
User Requirements	Functional performance criteria known. System options understood.	Ensure that the platform for system design is clearly understood and defined by users.
Functional Design	Tailored design evolving. Performance criteria clear, system construction evolving, i.e., specific components not known. System relationships/interfaces are known. Engineering line diagrams available.	Confirm system performance. Ensure the future maintainability of the system. Ensure that the evolving design addresses the consequences of potential functional failure.
Detailed Design	System construction details are largely defined.	The consequence of functional failure and the potential causes of functional failure are known. Maintenance tasks defined, Service Level Agreements (SLAs) and Standard Operating Procedures (SOPs) can be developed.

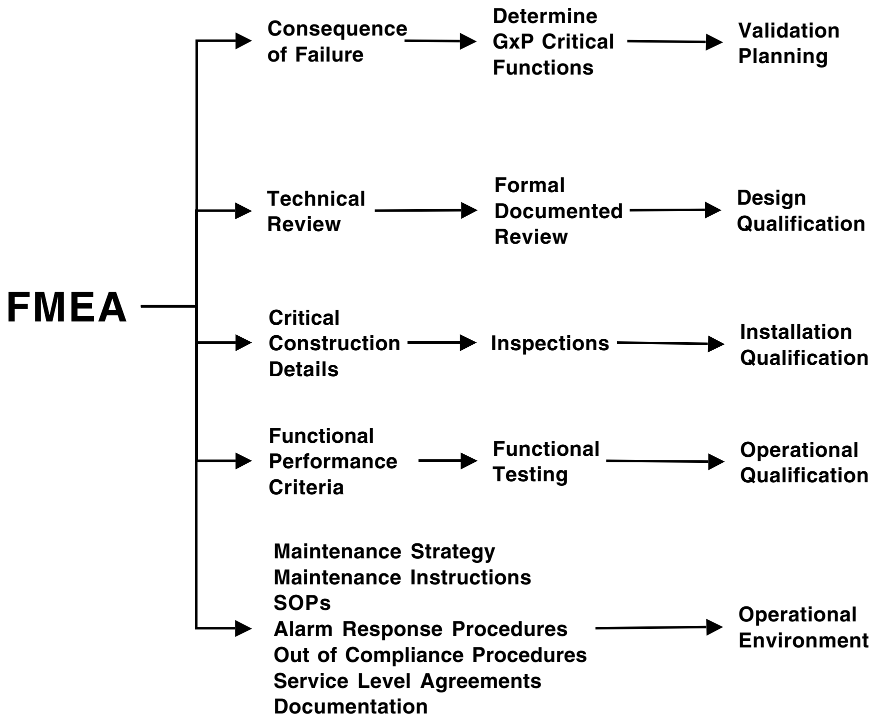


FIGURE 31.6 Relationships between FMEA and Asset Validation.

Construct Assets

Assets are constructed in accordance with the detailed design that provides the definitive description of the systems and components required to build facilities and assemble systems in a manner that will meet the business need.

During this period, construction documentation shall be established that provides an accurate basis for system commissioning and validation. Further, operation and maintenance plans, instructions, SOPs, and SLAs shall be developed in readiness for the handover of the assets to the operational environment.

Handover

Handover activities comprise commissioning and validation, operational and maintenance take-up, and user take-up. Collectively, these activities ensure the following:

- Facilities and systems are qualified against design and meet their predefined performance criteria.
- Engineering and user training have been successfully delivered.
- Operation and maintenance strategies, plans, SOPs, and SLAs have been developed and issued.
- Critical documentation and information supporting system design, validation, and operation has been imported into the relevant system within the EMS.

Operation and Maintenance

Operation and maintenance of assets delivered by the development project are managed in accordance with the asset management strategy defined by the FMEA process discussed earlier in the chapter. Figure 31.7 shows a typical operation and maintenance process that is a key component of engineering management strategy. The process comprises three primary phases: “Work Order Generation,” “Work Environment,” and “Reporting and Feedback.” Work Order Generation controls the creation of work requests and the identification of the work instructions, SOPs, documentation, and service level agreements required to conduct the work. Work Environment controls the application of the work instructions and SOPs in order to confirm current system performance, carry out the defined maintenance tasks, and reestablish performance prior to releasing the system back into the operational environment. It is essential to confirm system performance prior to carrying out the maintenance tasks in order to detect performance deviations requiring investigation and to assess the effectiveness of the engineering management strategy in preventing such performance deviations.

The Reporting and Feedback phase of the process establishes information such as:

- Maintenance records
- Performance deviation
- Condition monitoring
- Failure reporting
- Out of compliance records
- Maintenance costs

This information is used to establish trends which provide the basis for continuous improvement of the engineering management strategy including:

- Improved system design
- System replacement
- Improved operation and maintenance strategy

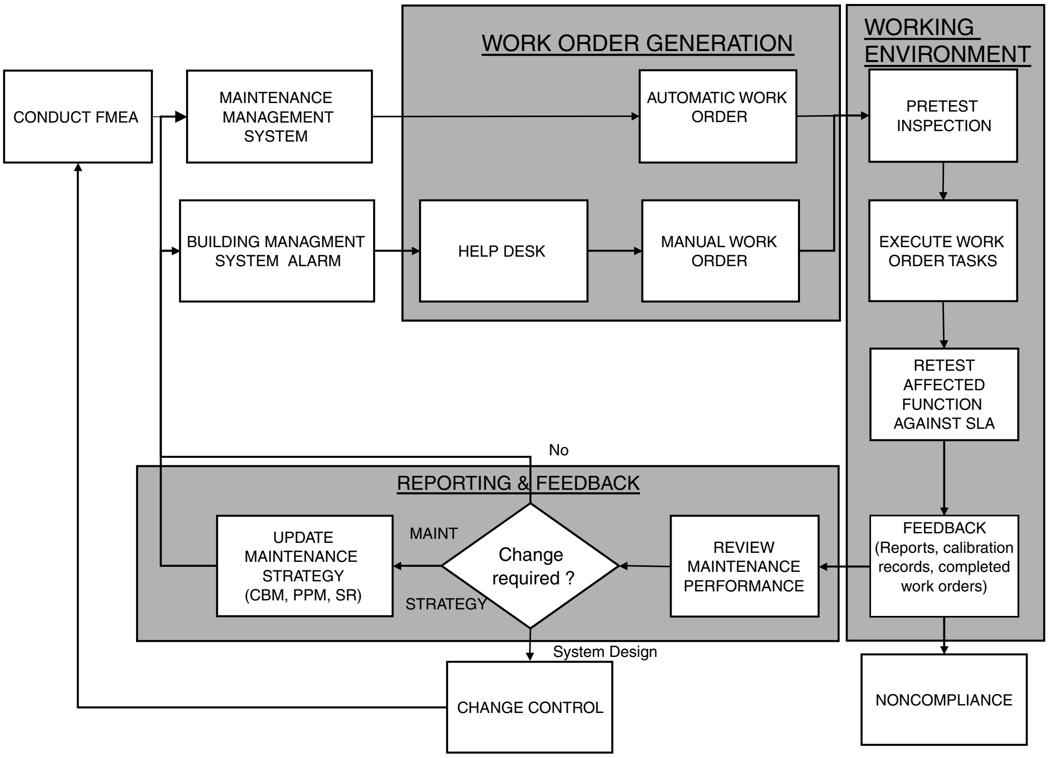


FIGURE 31.7 Operation and Maintenance Process.

- Improved operation and maintenance instructions and SOPs
- Improved technical documentation/information
- Improved maintenance reporting and feedback

It is clear that a failure to adequately define system function and performance criteria would make such continuous improvement difficult if not impossible.

DEFINING INFORMATION NEEDS TO SUPPORT THE ENGINEERING STRATEGY

Asset management information provides the foundation for all other GxP documentation used in the research and manufacture of drugs as shown in Figure 31.8. The results of a scientific study or a manufacturing campaign are worthless if the systems used to control drug-stability testing or the manufacturing process were not operated and maintained in accordance with their design and predetermined performance criteria.

Defining the information requirements to support asset management in a consistent manner that is understood by both the pharmaceutical organization and their suppliers is a considerable task. The pharmaceutical company must establish internal standards that define:

- System/asset numbering
- Asset management documentation needs
- Engineering drawing requirements
- EMS information structures

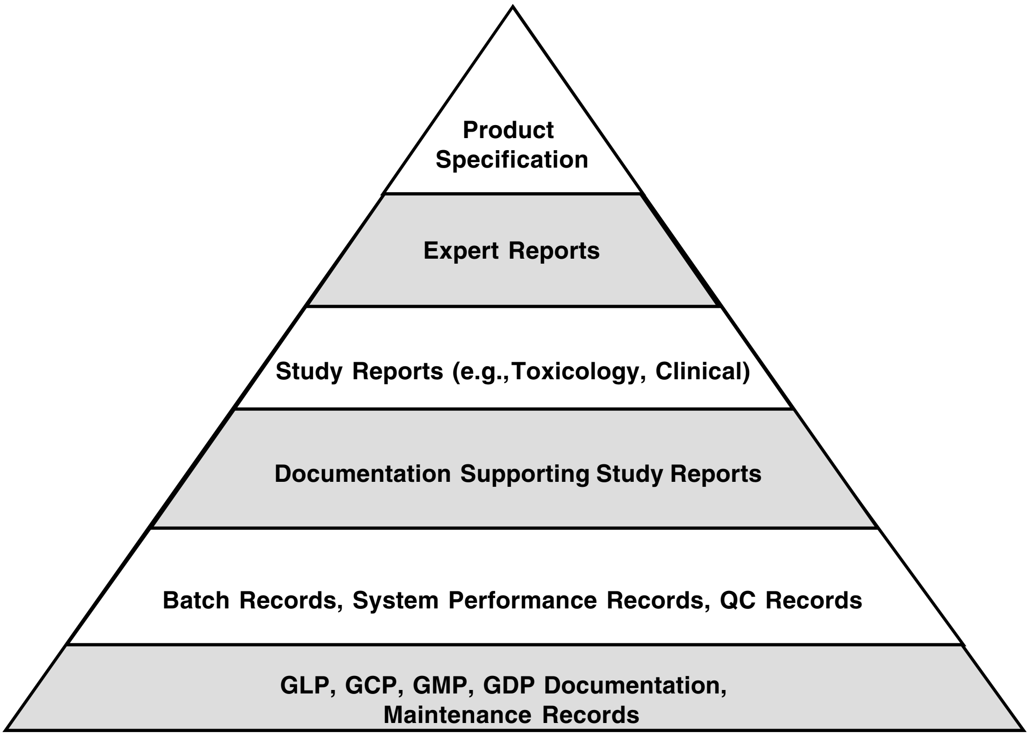

Product Specification
Expert Reports
Study Reports (e.g.,Toxicology, Clinical)
Documentation Supporting Study Reports
Batch Records, System Performance Records, QC Records
GLP, GCP, GMP, GDP Documentation, Maintenance Records

FIGURE 31.8 Documentation Pyramid.

- Engineering database structure (information templates for system types/system component types)
- Record requirements

Before establishing such standards, the pharmaceutical organization must assess the value of the information to the asset management process and consider the different requirements for information against the varied operating contexts of the business. The standards must also guard against information and documentation overload, which can be as detrimental as insufficient or unstructured information. The key objective of the standards is to ensure that information delivered by suppliers and maintained by the pharmaceutical organization is:

- Of the correct scope and depth
- System specific
- Relevant
- Accurate
- Single instantiation (or as few instantiations as possible)
- Easily retrievable and maintainable
- Controlled in accordance with risk

Ensuring the above must not be underestimated. Delivering information to the required technical standard in a consistent format and in a timely manner is a considerable project management task, especially when multiple suppliers are involved and when those suppliers have traditionally delivered information to their own and often unsatisfactory standard. Further, the import of such information into the information systems comprising the EMS is a challenging process. In some instances, there may be value in adopting the standards used by principal suppliers in order to minimize information translation and the associated risks.

Initiatives such as STEP (ISO 10303) attempt to define a standard to enable sharing and exchange of technical engineering data, independent of applications and organizations. Prominent industries involved in the development of STEP include aerospace, automotive, ship building, and process manufacturing, and there is an increasing awareness of such standards within the pharmaceutical industry.

Electronic Records and Electronic Signatures

Throughout this chapter we will highlight the importance of the information held within the Information Systems comprising the EMS in terms of supporting regulatory compliance. It is essential that all information created, maintained, and distributed within the EMS is evaluated against regulation relating to electronic records and electronic signatures management, the most prominent regulation being U.S. FDA 21 CFR Part 11, Electronic Records; Electronic Signatures, effective from August 20, 1997.

Table 31.2 provides examples of electronic records that may be maintained within the Information Systems comprising the EMS.

All electronic records need to be evaluated in order to determine their criticality and the appropriate controls that need to be established in order to assure integrity, authenticity, and where appropriate, confidentiality of the information contained within.

TABLE 31.2
Example Electronic Records

System	Typical Records
Engineering Database	Data sheets Calibration data and records Materials of construction data Asset configuration data Asset performance data Engineering drawing data
Building Management System/ Environmental Monitoring System	Asset performance data Process deviation alarms Environmental control deviation alarms Trends of critical process and environmental parameters
Maintenance Management System	Maintenance schedules Maintenance instructions Maintenance activity reports Supplier records Noncompliance and breakdown reports
Electronic Document Management System	Maintenance procedures and work instructions Engineering drawings Design and construction specifications Validation documentation Maintenance strategy records (e.g., reliability centered maintenance records) Calibration procedures Training records Operation and maintenance manuals

The need for authorization, review, and approval of each electronic record in response to regulatory or internal quality requirements needs to be determined. Where electronic records are signed electronically, the technical controls required by regulations such as U.S. FDA 21 CFR Part 11 need to be implemented within the system or a secure hybrid (signed printout of the electronic record) solution applied.

Role of the EMS

Pharmaceutical organizations are increasingly looking to information systems to maintain the vast volumes of information that support modern research and manufacturing operations.

Physical Architecture of the EMS

The EMS is not a single information system, but rather a collection of integrated systems providing information management, engineering control, and monitoring functions. Figure 31.9 presents a high-level representation of a typical EMS architecture. The engineering database is the hub of the architecture, providing a repository for information related to each asset (site, building, room/area/zone, system, etc.). The nuclear power and oil and gas industries have taken the lead in the development and utilization of "intelligent" databases. Such databases provide automated links to the Electronic Document Management System (EDMS) and Computer Aided Design (CAD) system in order to provide single point access to data. For example, a Functional Design Specification (FDS), CAD drawing, and SLA will contain automated links to the temperature performance criteria for an aseptic suite. If the engineer corrects an error on the drawing, the database information is automatically updated and propagates through the automated links to the FDS and the SLA. The information is therefore consistent (although not always correct) within all specifications, procedures, drawings, etc. The general principle that documents and information must be structured to minimize the number of occurrences of data is paramount to reducing the burden of information maintenance and the risk of regulatory noncompliance.

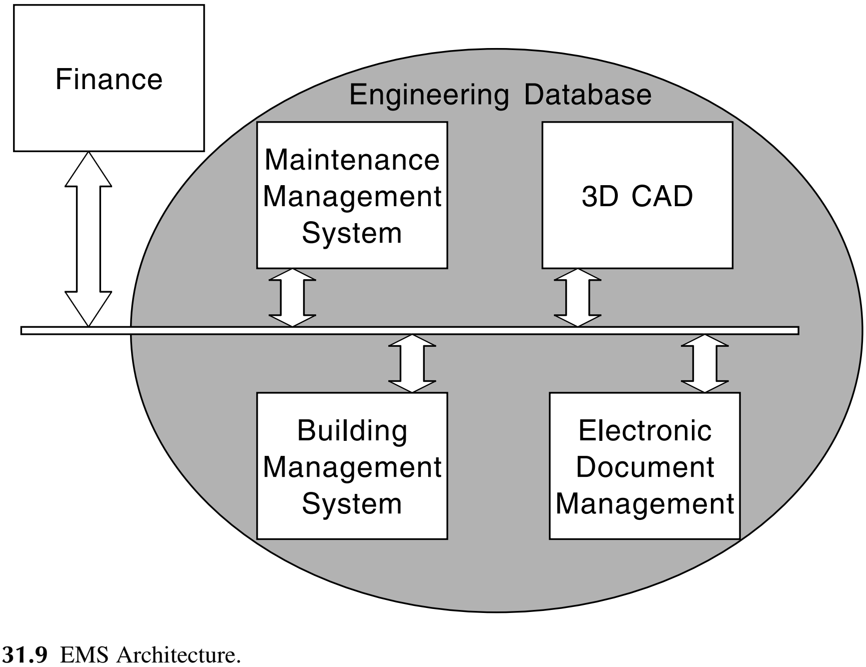

Finance
Engineering Database
Maintenance Management System
3D CAD
Building Management System
Electronic Document Management

FIGURE 31.9 EMS Architecture.

Access to information is as equally important as the controlled maintenance of information. For example, when a critical alarm is reported by the Building Management System (BMS), the engineer needs to promptly respond to instructions, SLAs, SOPs, and Engineering Drawings. The integration of the EDMS, CAD, and Maintenance Management System (MMS) with the BMS enables the documents and databases providing this essential information to be automatically available when the alarm is triggered.

Functional Architecture of the EMS

There is obvious functional overlap between the systems described in Figure 31.9. For example, the MMS will provide some degree of document control which obviously overlaps with the EDMS. We shall, therefore, describe the generic functionality associated with the EMS architecture.

Maintenance Planning

Maintenance tasks are either implemented proactively in order to prevent or minimize the chance of failure or reactively to correct a situation following failure. Preventive maintenance plans are derived from the FMEA process that will define the tasks and task frequencies required in order to maintain system reliability, consistency, and capability. Having applied the FMEA process in order to determine the maintenance strategy, it is essential that the system

- Schedules planned maintenance at defined intervals
- Identifies relevant maintenance task schedule (consistent with FMEA requirements)
- References the SLA defining system performance criteria and system criticality
- References documents, drawings, and databases supporting the maintenance tasks
- References instructions and SOPs to ensure controlled and consistent execution of maintenance tasks

Corrective maintenance conducted following a failure must be carried out in a similarly controlled manner. However, the engineer must manually construct the maintenance task schedule following investigation of the problem. The risk to GxP compliance, therefore, increases due to the manual intervention. It is essential that the organization of the EDMS and MMS, in particular, is structured to ensure that SOPs, documentation, drawings, and information are easily accessible through the relationship to a specific system.

Operation and Maintenance Implementation

Work instructions and SOPs define the operations required to start-up, operate, monitor, shutdown, and maintain systems. These work instructions and SOPs must be controlled in order to prevent inadvertent and unauthorized modification and to ensure access to only the latest revision of the document. Work instructions and SOPs must, therefore, be held in accessible but secure areas that are periodically backed up and archived. Information system access must be controlled by a hierarchical security system that constrains system operations in accordance with the role, responsibilities, and competency of the user. Access to and modification of the information supporting such work instructions and SOPs — for example, engineering drawings and specifications — must be controlled equally.

System Control and Performance Monitoring

The BMS and similar systems integrated into the EMS architecture provide control and performance monitoring functionality to ensure that performance criteria are met and performance deviations detected. Functional performance deviations will inevitably affect product quality and consequently GxP compliance. It is essential that the design process builds in quality to ensure that the system is "reliable," "consistent," and "capable" of meeting the predetermined performance criteria. Monitoring functions, although GxP critical, should only provide the fail-safe mechanism for detection and reporting of failures. Monitoring of process variables using available technologies such as BMS, ultrasonic devices, and vibration analysis may often be deployed in order to predict pending failures, enabling corrective action to be taken before performance and hence GxP compliance is lost.

Maintenance Reporting

Maintenance history is an essential component of GxP compliance. Work instructions and SOPs controlling maintenance operations should ensure that the maintenance engineer records all performance measures, observations, and maintenance tasks in a consistent manner with calibration records, for example, containing calibration parameters, with calibration procedure, reference to calibration equipment, name of engineer, date of calibration, next due date, etc. Where automated condition and performance monitoring is employed, the integrity of the recorded data is obviously a GxP issue.

Records generated by the asset management process shall be used to bring about continuous improvement in order to increase the effectiveness and efficiency of the engineering management strategy. All changes arising from the review must be controlled and documented.

TECHNICAL ISSUES

The review of the asset management process can be broken down into the technical issues that could potentially impact GxP. Table 31.3 is by no means an exhaustive list of issues that could potentially impact GxP compliance; however, they are a good indication of the criticality of the information held with the EMS.

TABLE 31.3
Technical Issues with Asset Management Functions

Asset Management Function	Technical Issue
Maintenance scheduling	Accuracy and consistency of interval between preventive maintenance work order issue
Assignment of maintenance tasks	Referential integrity between work order and maintenance task schedule/plan
Accuracy of instructions and procedures	Templates, print controls
Accuracy of information	Screen input formatting and input verification, data recovery, referential integrity, data transfers, print controls
Automated condition and performance monitoring	Scanning frequency, data communication integrity, records retention, time and date stamping
Manual condition and performance monitoring	Screen input formatting and input verification, data recovery, time and date stamping
Work order assignment	Robust relationship between tasks and trades, e.g., do not allow assignment of a mechanical installation to an electrician
Work order traceability	Event sequencing, e.g., "accept," "work in progress," "wait for parts," "approve," "complete"
Maintenance record traceability	Record "key" management and assignment of record to correct system
Archiving of maintenance records	Accurate retrieval
Change	Record locking to prevent parallel access, security to prevent inadvertent or malicious modification, disaster recovery, maintenance of referential integrity

REGULATORY IMPACT

Earlier discussions have provided a strong indication of the GxP criticality of the information and functionality of systems comprising the EMS. However, the GxP impact can only be determined in the context of the operating environment. Two identical mechanical systems may provide similar functionality; however, the fact that one system operates within a GxP environment whereas the other operates within an office block is fundamental and is inextricably linked to the consequence of system function failure. It, therefore, follows that the regulatory impact of information and the information systems that manage such information is inextricably linked to the functional criticality of the mechanical system.

GxP ASSESSMENT

As previously discussed, the FMEA process can be used to determine the consequence of functional failure. This process can, in turn, be applied to the functionality provided by the information systems comprising the EMS. For example, if an information system fails to generate preventive maintenance plans that are required for the periodic calibration of critical temperature control loops, the functionality of the information system responsible for the generation of preventive maintenance plans must be deemed GxP-critical. Once again we can see that operating context must be considered, as it is the environment within which the preventive maintenance plans are applied that determines GxP criticality.

The GxP Assessment is conducted in accordance with FMEA principles. Figure 31.10 provides a high-level representation of the GxP Assessment process. The flowchart is supported by standardized questions that challenge the impact of the EMS function on GxP compliance. Typical challenges make us ask, will the total or partial failure of the information system lead to

- Loss of, or interruption to, process system performance?
- Failure to conduct critical maintenance activities in accordance with a predetermined schedule?
- Use of superseded or wrong maintenance procedures?
- Incorrect maintenance/failure/performance reporting?
- Incorrect chronological reporting of operation and maintenance tasks?
- Loss or corruption of operational/maintenance data?
- Loss of database referential integrity?
- System security violation?
- Failure to recover following system failure?

The above is not an exhaustive list but provides an insight into the extent to which EMS functions can impact GxP.

EMS INFORMATION SYSTEM VALIDATION STRATEGY

Earlier discussions have referred to systems in terms of mechanical assets. The following discussions refer to the systems in terms of the information systems comprising the EMS.

PRINCIPLES OF CRITICALITY-BASED VALIDATION

The cost of validation is much publicized as is the debate regarding the extent to which information systems should be validated. Validation is essentially the term adopted by the pharmaceutical industry and their regulators to define the additional rigor required to confirm GxP critical aspects of information systems throughout the development and operational life of those systems. Given that pharmaceutical regulators have the power to withhold, suspend, or withdraw product licenses,

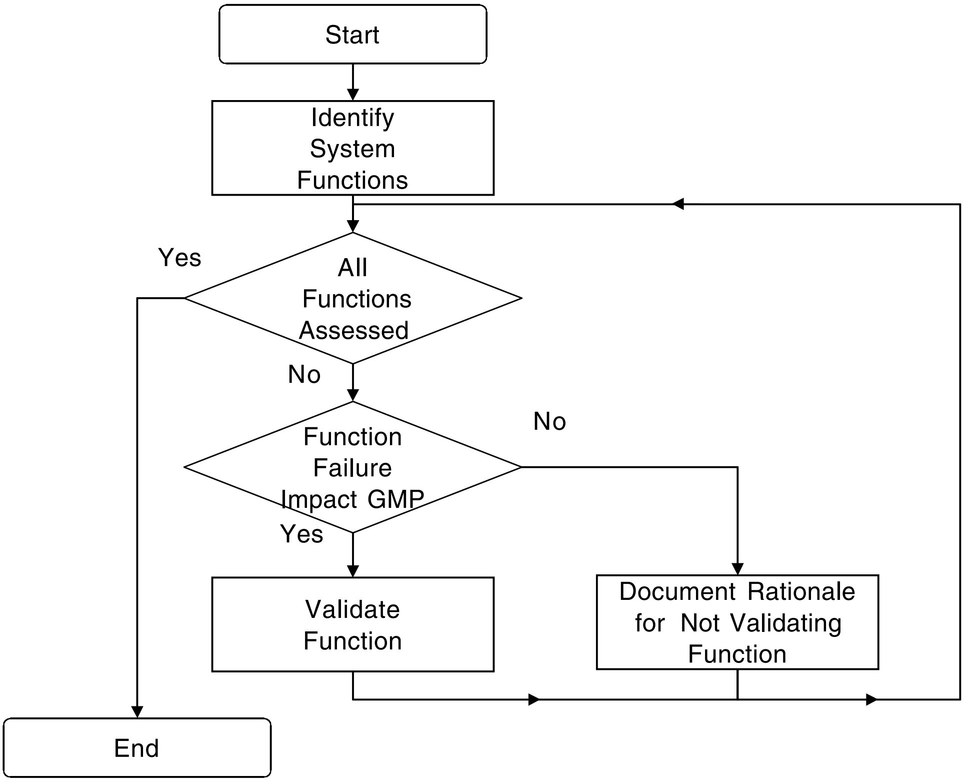

Start
Identify System Functions
Yes
All Functions Assessed
No
No
Function Failure Impact GMP
Yes
Validate Function
Document Rationale for Not Validating Function
End

FIGURE 31.10 GxP Assessment.

it is essential that pharmaceutical organizations validate GxP critical functions. In order to maximize business efficiency and minimize cost, it is also essential that pharmaceutical organizations differentiate those functions of the information systems that are GxP critical and those that are not, and hence focus valuable and limited resource where it is most warranted.

Processes used to determine functional criticality, such as FMEA, have already been discussed within this chapter. Similarly, FMEA and other risk assessment tools can be used to determine the scope of validation. The risk of failure increases as information systems supporting the EMS strategy deviates from a standardized solution; that is, the level of tailored development increases. In addition, the extent to which a product is utilized within industry, in particular pharmaceuticals, must be taken into consideration when determining the scope of validation.

GAMP 4[1] provides guidance for the extent to which operating systems, third-party packages, and applications utilized by the EMS must be validated from the simple recording of the version number for extensively used, industry standard operating systems to full life-cycle validation for tailored applications. There will, however, always be gray areas where the level of validation documentation required is under debate. A general principle is "If in doubt, err on the side of caution," as the cost of the additional effort may not always exceed the cost of the debate (see Figure 31.11).

VALIDATION LIFE CYCLE

Figure 31.12 depicts a typical validation life cycle from validation planning to validation reporting and ongoing support. Responsibility for each phase of the validation life cycle will switch from the pharmaceutical organization to the supplier at certain key phases. It is, however, the fundamental

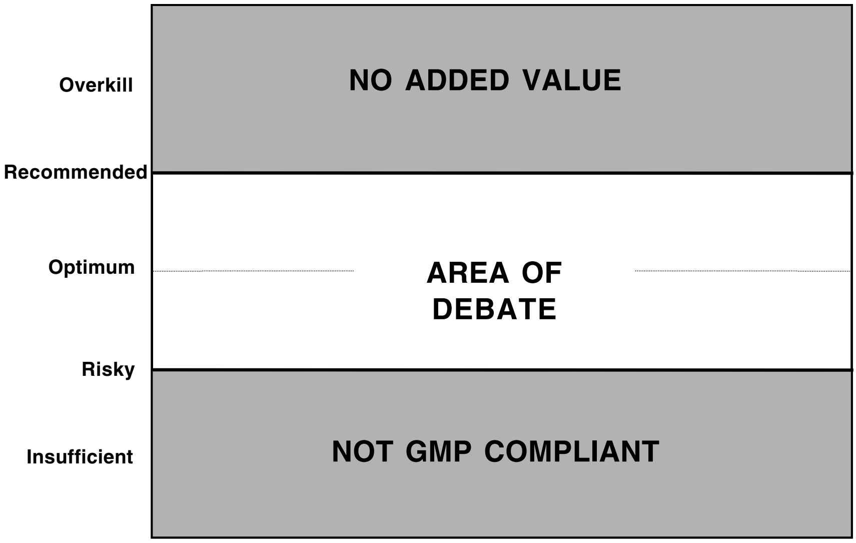


FIGURE 31.11 Documentation Levels.

responsibility of the pharmaceutical organization to assure the supplier that all phases of the life cycle have been conducted in a quality manner, consistent with the expectations of the pharmaceutical regulators.[2–4] A detailed discussion on validation life cycles can be found in "Validating Automated Manufacturing and Laboratory Applications."[5]

Validation Master Plan (VMP)

The pharmaceutical organization shall develop a VMP to define the validation strategy for the implementation of the EMS. The VMP should address the process by which the pharmaceutical organization shall assure itself of the quality of the products being procured and the strategy for validation of its specific implementation of those products.

Regulators consider the VMP to be the first fundamental commitment of the pharmaceutical organization to the validation process. In particular, the VMP must recognize that different approaches may be required to meet the differing capabilities of the various suppliers contributing to the EMS architecture. The VMP must be reviewed and approved by the project sponsor, GxP process owners, and quality assurance representative. Table 31.4 lists the typical contents of the VMP.

User Requirement Specification (URS)

The URS is developed by the pharmaceutical organization and forms the foundation of the project. It should convey the asset management processes, functional requirements, and performance criteria to the supplier. The objective of the URS is to convey business needs rather than technical solutions (other than where corporate standards apply). The URS should be written in a clear, concise, and unambiguous manner that facilitates traceability throughout the design and testing phases. GAMP 4[1] provides a guideline for the development of URSs. Additional guidance was also produced by the GAMP's Special Interest Group for Suppliers in 1999. Table 31.5 provides the typical contents of the URS.

Supplier Audit

All suppliers, system integrators, and consultants contributing to the supply of systems and advice that may impact GxP regulations must be audited. Often the supplier development organization is

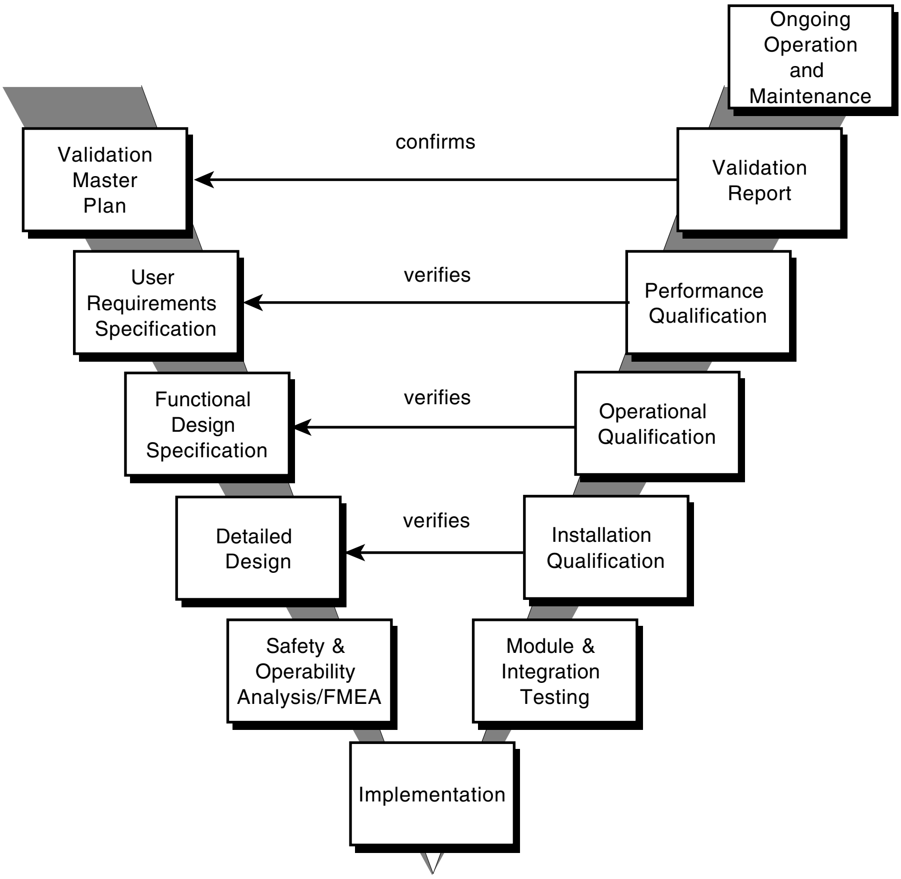

Ongoing Operation and Maintenance
Validation Master Plan
confirms
Validation Report
User Requirements Specification
verifies
Performance Qualification
Functional Design Specification
verifies
Operational Qualification
Detailed Design
verifies
Installation Qualification
Safety & Operability Analysis/FMEA
Module & Integration Testing
Implementation

FIGURE 31.12 Validation Life Cycle.

TABLE 31.4
Validation Master Plan Contents

Validation scope
Validation strategy (core products and applications)
Validation organization
Roles and responsibilities
Validation documentation requirements
Document approval authorities
Key phasing and milestones

logistically separate from the support organization and will utilize different Quality Management Systems (QMSs) in the execution of its services, further complicating the audit plan. Where the EMS architecture is complex and comprises several systems from a variety of organizations, the cost of the audit phase can be considerable. The increasing use of postal audits by pharmaceutical organizations has, however, had a significant and positive impact on the efficiency and cost of auditing. Site audits are only conducted when there is considerable risk arising from the use of the application or where the results of the postal audit are dubious or indicate serious weaknesses that need to be investigated in greater detail.

TABLE 31.5
Typical Contents of the URS

- Asset management processes
  - Maintenance strategy development (FMEA, performance monitoring, etc.)
  - Maintenance strategy implementation
  - Document management
  - Drawing management
  - Change control, etc.
- Functionality
  - Process step definition
  - Functional performance criteria
  - Functional failure mode analysis
  - Failure mode recovery requirements
- Informational requirements
  - Information structures
  - Legacy system interfaces
  - Data entry range
  - Data retention requirements
- Human/machine interface requirements
  - Screen specifications
  - Data entry modes
  - Refresh rates
- Data migration
  - Legacy system data structures
  - Manual process data
  - Data transformation requirements
  - Data cleansing
  - Data archive and restoration
- Security requirements
  - Access levels
  - Security mechanism
- Communication interfaces
  - Information transfers
  - Transfer frequencies
  - Legacy system protocols
- Client/server infrastructure
- Standards
  - Corporate hardware standards
  - Current installations
  - Environmental conditions
  - Hazardous, static, dust, etc.

The Supplier Audit shall establish whether the controls applied to the development of the core product and application configuration are consistent with GxP requirements and whether the organization is technically, organizationally, and commercially capable of supporting the application for its anticipated life. The Supplier Audit will collate information for review and, where required, corrective action, and determine whether a follow-up audit is required. An example postal audit questionnaire is presented in Table 31.6. The questionnaire is not exhaustive. However, it clearly demonstrates the objective and scope of the postal audit.

TABLE 31.6
Example Postal Audit Questionnaire

Question	Yes/No Comment	Evidence Attached Yes/No
Organization		
Is the company organization documented?		
Does the organization include specific responsibilities for quality?		
Are roles and responsibilities of the organization documented?		
Is there a project management structure?		
Quality Systems		
Is there a documented QMS?		
Has accreditation/registration been achieved for:		
a. BS 5750 Pt. 1 or 2 or ISO 9000?		
b. TickIT?		
Please detail any other quality accreditations/registrations held.		
Is the QMS based on a life-cycle approach?		
Does the QMS cover post-delivery support and maintenance?		
Have procedures been established to control each phase of the development and operational life cycle?		
Are internal quality audits/inspections conducted on a regular basis?		
Planning		
Do you develop detailed project plans (e.g., Gantt charts)?		
Do project plans include task dependencies?		
Do project plans identify resource assigned to each task?		
Do project plans identify critical paths?		
Do you develop project quality plans?		
Are plans reviewed and/or approved by your customers?		
Specification		
Do you insist that your customers issue a URS?		
Do you develop an FDS in response to the URS?		
Do you develop a cross-reference matrix to relate all URS clauses to the FDS?		
Design		
Are detailed SDS produced?		
Are SMDS produced?		
Are detailed HDS produced?		
Are customers invited to attend design review meetings?		
Are design reviews minuted?		
Are design changes:		
a. Proposed?		
b. Approved?		
c. Implemented?		
d. Controlled?		
Do controls extend to subcontractors?		
Are subcontractors audited?		
Implementation		
Do software coding standards exist?		
Are SCRs conducted?		

TABLE 31.6 (Continued)
Example Postal Audit Questionnaire

	Yes/No	Evidence Attached
Question	Comment	Yes/No
Are SCRs documented?		
Are changes traceable from the code?		
Is there a procedure for data migration?		
Is there a procedure to control the inadvertent or malicious modification of software?		
Testing		
Is software module testing conducted?		
Is software integration testing conducted?		
Is hardware testing conducted?		
Is factory acceptance testing conducted?		
Are test specifications produced to cover all testing?		
Are test acceptance criteria defined?		
Are test results recorded?		
Is test evidence retained (e.g., printouts, screen dumps, etc.)?		
Installation		
Is installation carried out by subcontractors?		
Is installation controlled by procedures?		
Are installation inspections conducted?		
Is site acceptance testing conducted?		
Are customers invited to witness tests?		
Are installation reports produced?		
Support and Maintenance		
Is an SLA established with the customer?		
Is a help desk facility provided?		
Are customers notified of faults/defects/anomalies within your product?		
Do you have procedures to control bug fixes?		
Do you have procedures to control hardware and software upgrades?		
How long do you support:		
a. Hardware		
b. Software		
What is the minimum notice period before support is withdrawn for a hardware or software product?		
Is there a change control procedure?		
Are change control records maintained?		
Personnel Development		
Is personnel development planned?		
Are records of personnel development maintained?		
Are records of personnel experience maintained?		
Document Management		
Are procedures and documents reviewed and approved prior to issue.		
Are procedure and document changes managed through formal change control?		
Are obsolete documents withdrawn?		
Are subcontract documentation standards audited?		

TABLE 31.7
Project Quality Plan

Project background and scope
Organization, roles, and responsibilities (internal and external)
Approach (life-cycle activities, tools, and methodologies)
Quality standards and procedures (internal and external)
Review points
Communication channels
Key deliverables (including documentation)
Milestones

Where a site audit is required, the audit can focus on the main areas of concern raised by the postal audit, thus reducing the duration of the audit and enabling a more in-depth review of the main areas of risk.

Auditors must assure themselves that there is sufficient documentary evidence available to demonstrate that quality controls appropriate to the pharmaceutical industry are in place and are being routinely applied. The supplier should have produced quality plans for all projects, similar in objective to the VMP produced by the pharmaceutical organization.

The following sections define the typical areas to be challenged by the audit team and the scope and content of documents produced at each phase of the project life cycle.

Quality Plan

Where a quality plan exists, it will often be used as a guide to the audit. Specific activities and documentation stated in the quality plan should be reviewed against their controlling procedures. The quality plan shall also reference the project plan, usually presented as a Gantt chart that defines:

- Project tasks
- Task start, end, and duration
- Task dependencies
- Resource allocation
- Critical path

The project plan should be reviewed to ensure that all tasks have been defined, and that the estimates are realistic and will not compromise the delivery and quality of the information system. Ongoing monitoring of the project plan is essential as project slippage and the associated commercial implications generally lead to shortcuts in key quality controls such as application of procedures, detailed design, reviews, and testing. The earlier that project slippage can be detected, the earlier corrective action can be taken to minimize the risk to project delivery and quality. Table 31.7 provides the typical content of the project quality plan.

Functional Design Specification (FDS)

The supplier should be able to demonstrate that an FDS is produced in response to the pharmaceutical organization's URS. Typically, the FDS will be a standard document for the core product application with deviations from the standard offering documented in an addendum or separate project-based document. Deviations usually warrant some degree of tailored software development and should, therefore, be of prime consideration during the validation exercise. Table 31.8 provides typical content of the FDS.

TABLE 31.8
Typical Contents of the Functional Design Specification

- System architecture
  - Diagrammatic representation of the major hardware components of the system
  - Client/server architecture
  - Hardware component interfaces
  - Geographical location of major hardware components
- Software module architecture
  - Hierarchical structure of software modules and packages
- Process definitions
  - Diagrammatic representation of business processes implemented within information system, e.g., maintenance management, change control, document management, etc.
  - Work flows
- Functional definition (flowcharts, narratives)
  - Function inputs
  - Function objectives
  - Functional performance
  - Function outputs
  - Functional failure modes and reporting
  - Functional failure mode recovery
  - Function synchronization (events, relationships)
- Information structures
  - Database schema
- Data migration strategy
  - Migration of information from legacy systems
  - Import of information from manual systems
- Data storage
  - Folder/directory structures
  - File structures
  - File sizes
  - File properties
  - File access
  - Storage media
  - Storage capacity
  - Data retrieval rates
- User interfaces
  - Menu/display hierarchy
  - Display structure
  - Screen formats
  - Screen access security
  - Toolbar options
  - Message bars
  - Data entry field configuration
  - Window configuration
  - Refresh rates
  - Input devices (mouse, touch screen, keyboard)
- Communication interfaces
  - Wide area networks to other sites
  - Local area networks/intranet
  - Serial interface protocols
  - Message packet formats
  - File transfer protocols

TABLE 31.8 (Continued)
Typical Contents of the Functional Design Specification

- Transfer frequency
- Transfer rate

- System reliability and performance
  - Server performance
  - Network bandwidth
  - Serial communication performance
  - Printer performance
  - Backup and restoration
- Expansion/enhancement capability
  - Redundancy
  - Hardware upgrade paths

TABLE 31.9
Hardware Design Specification

- System architecture diagrams
- Layout and wiring diagrams and drawings
- Main component specifications
- System interface specifications
- Performance (CPU, bus, cache, clock, etc.)
- Capacities (RAM, hard disk, floppy disk, DAT, CD Rom, etc.)
- Peripherals (HMI, printers, keyboards, mouse, barcodes, etc.)
- Interfaces (communications cards, network connections, cabling, speed)
- Settings (switch settings, firmware configuration)
- Environment (temperature, humidity, RFI, UV, electromagnetic)
- Electrical supplies (UPS, earth requirements, filters, etc.)
- Define relevant standards (safety, electrical, etc.)

The FDS is the first critical technical document produced by the supplier and will demonstrate the supplier's understanding of the user requirements and forms the foundation of the final technical solution. If either of the above is inaccurate, inconsistent, or ambiguous, the likelihood of project failure is high.

The FDS must be fully traceable to the URS, clearly demonstrating that all URS clauses have been met. Where the FDS deviates from the URS, a rationale for the potential impact of the deviation must be provided.

HARDWARE DESIGN SPECIFICATION (HDS)

The HDS will define the hardware platform to support the EMS architecture. It is likely that the pharmaceutical organization will impose corporate standards to ensure compatibility with other installations on the site. In many instances the supplier will simply state the hardware requirements and allow the pharmaceutical manufacturer to procure the hardware, especially when the EMS architecture is comprised of a number of systems from a variety of suppliers. Table 31.9 provides the typical contents of the HDS. The Installation Qualification (IQ) shall be developed to verify the major critical components stated in the HDS.

TABLE 31.10
Software Design Specification

Module architecture
Module descriptions
Module interfaces
Module relationships (events, timers, handshaking)
Database schema
File structures
System interfaces

TABLE 31.11
Software Module Design Specification

Module input parameter definition (integer, real, char)
Global data definitions
Local data definitions
Parameter passing mechanism (pass by value, pass by reference)
Detailed functional definition
Returned values
Programming standards
Test harnesses

Software Design Specification (SDS)

The SDS provides a detailed decomposition of the processes and functions defined in the FDS. The audit should establish that appropriate design methodologies have been applied leading to a structured modular and logical design. The SDS should provide sufficient detail to enable unambiguous implementation of the software. Table 31.10 provides the typical content of the SDS.

Specific module specifications should be produced for complex software. Table 31.11 provides the typical contents of a Software Module Specification.

Good Programming Practice and Source Code Reviews (SCR)

Suppliers should have established standards to govern the development of software. The objective of such standards is to ensure a consistent and structured approach to the development of software, thus minimizing the risk of software failure and enhancing software maintainability, avoiding personal style while trying not to suppress creativity.

Suppliers should conduct SCRs on all critical software modules in order to capture deviations from programming standards, identify logic errors, and ensure software modularity. Tailored software developed to satisfy user requirements not catered for within the standard product offering should be a particular focus of attention as the risk of software failure increases for new software developments. SCRs should be documented in order to record observations raised against the software and resultant corrective actions. Further, documented evidence of the implementation of corrective actions should be available for inspection. Where software modules present a major risk to GxP compliance or evidence of internal SCRs is limited, the pharmaceutical organization should consider additional independent reviews. Table 31.12 details the scope and content of programming standards.

TABLE 31.12
Good Programming Standards

Data scoping
Module size
Module layout
Module cohesion and coupling
Naming conventions
Use of control blocks
Module structure
Commenting

TABLE 31.13
Configuration Management

Development is controlled within a project-specific area on the development system
The development area is organized into a meaningful folder structure to facilitate controlled access to files
Tested files are held in secure read-only areas
Files are "booked out" of secure areas before modification can take place
Files are "booked in" to secure areas once tested
Simultaneous file access is prohibited
File access is restricted to authorized users
Version control is applied to track file modification
Command files are controlled in the same manner as software files and configuration files
Records of the development environment are maintained to enable reconstruction of the development environment for subsequent modification (e.g., compilers, linkers, assemblers, operating system versions)

Configuration Management

Configuration management ensures that hardware and software is controlled during the development and operational phases. Configuration management extends beyond the control of software modules during development to the control of the development environment (development system configuration, operating systems, command files, compilers, linkers, etc.) and documentation. Table 31.13 lists the requirements of configuration management.

The whole project development environment must be routinely backed up during development phase and archived at the end of the development to enable reconstruction of the development environment so that it facilitates subsequent maintenance activities and disaster recovery.

Configuration management must also consider hardware configuration, both of the development environment and the test environment, especially where the supplier maintains systems for a number of years.

Supplier audits must establish whether such configuration management requirements are implemented by the supplier organization. Experience shows that rarely are all controls applied. However, there is an increasing use of standard configuration management products that provide access control and modification of software, configuration, and documentation files.

Application Configuration Specification

The Application Configuration Specification (ACS) documents the application configuration required to meet the URS. The ACS shall record system set-up parameters, process configuration, database configuration, file structures, etc. required to implement the specific business implementation of the system. Where a standard implementation of the core product is adopted, this will be the only custom specification delivered to the pharmaceutical organization, accompanied by standard technical and user manuals.

Supplier Testing

Testing is conducted in several phases depending on the complexity of the software design. The supplier shall be responsible for developing module, integration, factory, and site acceptance test specifications to demonstrate that the design has been fully and accurately implemented.

Module tests are conducted against the software module design specifications in order to verify the discrete functionality of the module and simulate the input and output interfaces of the module. Module testing often requires the development of "test harnesses" simulation software specifically written to supply inputs to the module and interpret the returned result. The test harnesses themselves further pose a risk to GxP compliance as they must be appropriately developed and controlled.

Integration tests are conducted to challenge the integration of new modules into the system. Table 31.14 provides the typical scope of the integration test specification.

One of the greatest challenges to system suppliers is demonstrating that the integration of new software modules has not had a detrimental impact on the existing software. Regression tests must, therefore, be conducted in order to provide reasonable assurance that existing modules have not been affected. In extreme circumstances, it may be necessary to compare the image of the new software release with the image of the previous version in order to determine which modules have been affected and then provide a documented rationale for the potential impact on each module.

TABLE 31.14
Integration Testing

Module interfaces	
	Number of parameters compare
	Parameter types compare
	Parameter passing mechanism is correct, e.g., by name, by value, by reference
Module synchronization	
	Handshakes, e.g., events, interlocks
	Sequencing
System performance	
	Functional performance
	Information storage and retrieval (e.g., database queries)
	Screen refresh
	Data entry response times
	Serial communication interfaces
	Network performance
	Data storage capacity
	Multiple user access
File and data integrity	
	Shared file access
	System failure during read/write operations
	Data retention following system failure
	Cyclical file management
	Prevention of duplicate record creation
	Referential integrity
Impact on existing modules	
	Regression testing

Factory Acceptance Testing (FAT) is conducted within a simulated environment to demonstrate that the system meets the URS and FDS. The FAT will only be conducted if the software is a new development or major adaptation of the standard product. The FAT is the first opportunity for the pharmaceutical manufacturer to test the system in its entirety. The scope of the testing should be wide enough to ensure that most problems can be identified and rectified within the development environment. Where tests are not dependent on the operating environment as, for example, in data entry validation, the FAT may serve as the validation record for that function. However, in order to adopt this approach, formal test specifications must be developed and approved prior to executing the tests, and results must be clearly recorded.

Supplier Audit Reports

An audit report will be produced for each supplier, documenting the positive and negative observations made during the assessment of the response to the postal audit questionnaire and/or the site audit. All corrective actions must be followed up, possibly requiring further site visits, in order to ensure that nonconformance issues have been appropriately addressed in a timely manner to minimize the impact on project success.

Audit reports must be factual and not subjective, clearly stating the basis of observations and the criticality of the observation (major or minor). Suppliers must be allowed to review the audit report prior to issue to enable observations to be agreed upon or, where a misunderstanding has occurred, to enable additional mitigating information to be provided.

Audit reports should avoid the detailed specification of corrective actions. It is the supplier's responsibility to define how observations will be addressed within the context of the supplier organization and quality systems.

In certain circumstances, the supplier may consider corrective actions contrary to their business objectives. For example, a supplier who is dependent on the pharmaceutical industry for only 10% of its business may be reluctant to implement corrective actions specific to pharmaceutical regulatory requirements. The pharmaceutical organization must then determine whether to seek an alternative supplier or implement corrective actions within their own organization in order to address the issues.

Installation Qualification (IQ)

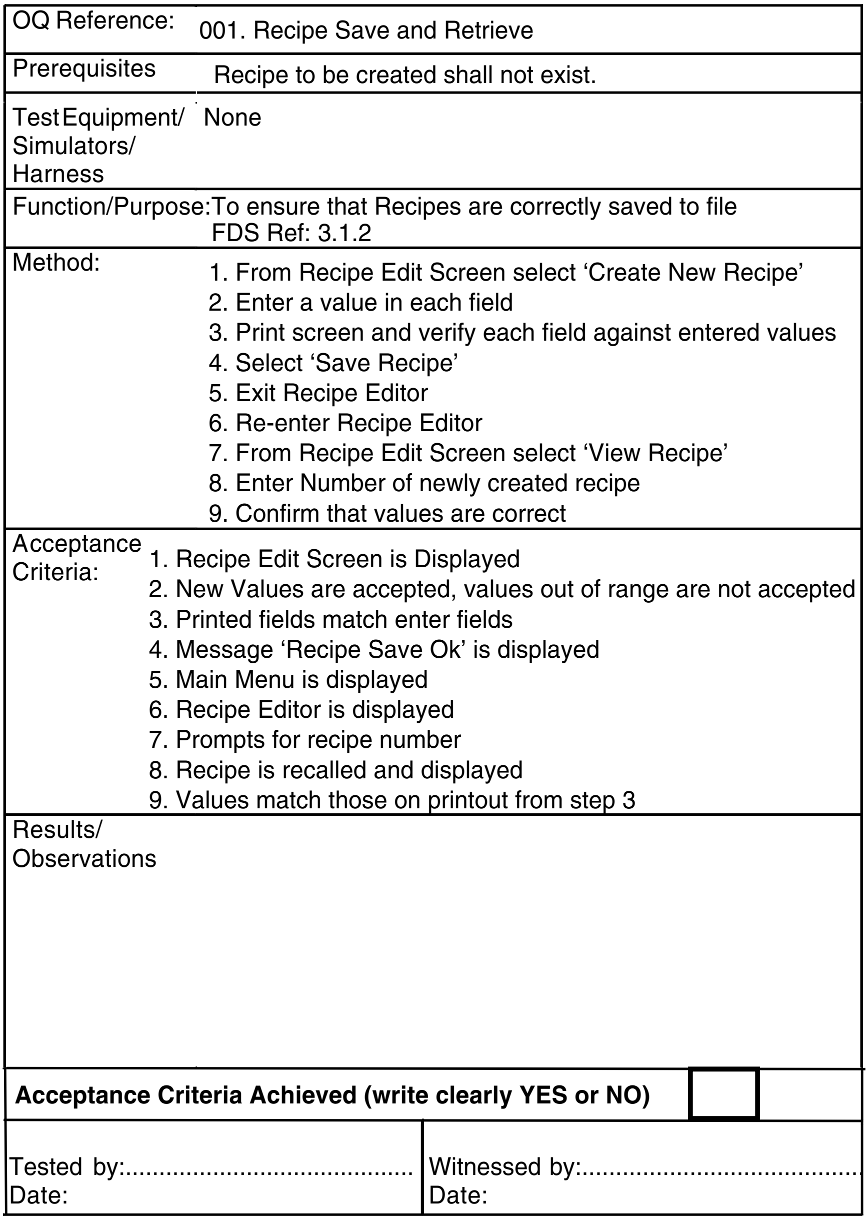
The IQ is the responsibility of the pharmaceutical organization. The IQ shall define methodical inspections that verify the hardware and software installation against the design. Each inspection is conducted in accordance with a detailed inspection method and the outcome verified against unambiguous acceptance criteria. The results of the inspection must be recorded on a test result sheet, referenced to or contained within the IQ protocol. An inspection will be deemed to have passed if all the acceptance criteria set forward have been satisfied. Table 31.15 provides the typical contents of an IQ.

Operational Qualification (OQ)

The Operational Qualification (OQ) should be integrated with the Site Acceptance Testing (SAT) normally conducted by the supplier. The OQ verifies the functionality of the system within its normal operating environment. An OQ protocol shall be developed which clearly defines the methodology by which the tests shall be conducted and the acceptance criteria that shall determine the success or failure of the test. Figure 31.13 provides an example of a typical OQ test script. The OQ must reasonably challenge the operating boundaries of each function (although never to destruction). For example, data input functions should be challenged by entering:

TABLE 31.15
Typical Contents of an Installation Qualification

- Installation plans/procedures
  - Satisfactory execution of installation procedures
- Software installation
  - Correct executable images installed (including versions)
  - Correct third-party software packages installed (including versions)
  - Correct folder/directory structure created and files installed within folders/directories
- Software configuration completed satisfactorily
  - Site and system identification
  - User access groups
  - Security configuration
  - Menu/display access configuration
  - Logical device connections
- Inspection of critical hardware components
  - Servers in correct locations
  - Processor speed
  - Cache size
  - ROM bios
  - Memory capacity
  - Peripherals
  - Storage devices
  - Input devices
  - Network interface cards, addressing, and connections
  - Printers
  - Connection
  - Bit switch settings
  - Printer driver installation
- Networks
  - Connection
  - Network addressing
  - Server synchronization (e.g., date and time)
  - Network conflicts
- Electrical installation
  - Cable connections
  - Electrical testing
  - Uninterrupted power supplies
- Input/output
  - Outstation connection and configuration
  - Field device connection and calibration
  - Diagnostic checks
  - System performance
- Documentation
  - User manuals
  - Technical documentation
  - Availability of user SOPs
  - Availability of disaster recovery recovery plans
- Training
  - Availability of training plans



OQ Reference:	001. Recipe Save and Retrieve
Prerequisites	Recipe to be created shall not exist.
Test Equipment/ Simulators/ Harness	None
Function/Purpose:	To ensure that Recipes are correctly saved to file FDS Ref: 3.1.2
Method:	1. From Recipe Edit Screen select ‘Create New Recipe’ 2. Enter a value in each field 3. Print screen and verify each field against entered values 4. Select ‘Save Recipe’ 5. Exit Recipe Editor 6. Re-enter Recipe Editor 7. From Recipe Edit Screen select ‘View Recipe’ 8. Enter Number of newly created recipe 9. Confirm that values are correct
Acceptance Criteria:	1. Recipe Edit Screen is Displayed 2. New Values are accepted, values out of range are not accepted 3. Printed fields match enter fields 4. Message ‘Recipe Save Ok’ is displayed 5. Main Menu is displayed 6. Recipe Editor is displayed 7. Prompts for recipe number 8. Recipe is recalled and displayed 9. Values match those on printout from step 3
Results/ Observations	
Acceptance Criteria Achieved (write clearly YES or NO)	□
Tested by:.. Date:	Witnessed by:....................................... Date:

FIGURE 31.13 Test Script.

- Nonnumeric characters in numeric fields
- Values on and slightly outside operating ranges
- Field lengths that exceed the permitted number of characters for the field
- Negative values in positive value fields
- Decimal values in integer configured fields

Table 31.16 provides the typical content of the OQ.

As with the IQ, all protocols must be preapproved and postapproved, and tests independently witnessed or reviewed. When writing OQ tests, it is important that the acceptance criteria is clear

TABLE 31.16
Typical Content of the Operational Qualification

Process flows
Functional operation and performance
Failure processing, reporting, and recovery
Multisite challenge
Operating boundaries (e.g., data entry)
Network interfaces
Serial communication interfaces
Start-up and shutdown
Security and access
Data storage and retrieval
Error reporting
Backup and restoration SOPs
User SOPs
Contingency plans
User and system administrator training (delivery)

and not embedded in the test method where it may be overlooked. Tolerances on the acceptance criteria must be in line with design and should not be so wide as to guarantee success.

Performance Qualification (PQ)

The PQ will demonstrate the consistent operation of the system once released for operational use. The scope of the PQ for an information system is described in Table 31.17. During the PQ period, it is preferable that legacy systems and manual systems are operated in parallel. This approach is not without difficulty and the strategy for parallel operation must be carefully planned. The PQ will require the collation of records such as change control, operator logs, etc., across many sites. The procedures and responsibilities for the collection of records in support of the PQ must be established in advance of system cut-over. Further, the PQ review team must be determined to ensure that personnel are available to conduct the review, considering the pressures that will be exerted by the business to divert resources onto the next project.

The outcome of the PQ may indicate that there are areas of weakness in the design or implementation of the application, in particular if high levels of design changes or functional failures are observed within a localized area. In such instances, it will be necessary to reconsider the adequacy of the OQ, and further testing may be required to determine the root cause of the failures. The outcome of the PQ shall further define the need to extend the PQ duration.

Validation Report

The Validation Report responds to the VMP, providing a summary of the actual approach taken and the documentation produced. Any deviations from the approach prescribed by the VMP must be justified and the consequence of the deviation assessed. Where the deviation is not acceptable, a corrective action plan must be formulated to address the issue. It may be possible that by implementing manual procedures to overcome the issue in the short term, the system will be enabled to move into the operational environment while the issue is being addressed.

The Validation Report should clearly demonstrate that a suitable operating environment has been established including procedures to control documents, changes, backup, archive, restoration, and security, and that appropriate service contracts have been established.

Where there are no issues or they can be overcome by manual procedures, the system may be recommended for operational use.

Operational Environment

The validated status of information systems must be maintained during the operational life of the system. It is, therefore, essential that procedures are developed to control operation and maintenance of the system, post-cutover. Such controls shall include

- System operation
- Change control
- Backup and restoration
- System upgrade
- Contingency plans
- Service Level Agreements (SLAs)

All procedures must be developed, reviewed, and approved prior to OQ. The effectiveness of the operation and maintenance procedures and plans shall, where possible, be verified during the OQ and PQ phases.

ASSET MANAGEMENT ORGANIZATION

As a final note, the culture of the asset management organization is fundamental to establishing and maintaining GxP compliance of the process and information systems comprising the asset management infrastructure.

Senior management must assign high priority to GxP and not abdicate responsibility to users because they believe they are in the front line for regulatory inspection. As was depicted by Figure 31.7, asset information and validation documentation is the foundation of regulatory submissions. If the foundation crumbles, so does the credibility of those submissions and subsequently the reputation and profitability of the pharmaceutical organization.

Effort must be expended to raise GxP awareness and to ensure that personnel view the rigors of producing GxP documentation as an integral part of the asset management process. Documentation should be seen as the definition and recording of essential processes and not as a paperwork exercise to satisfy the regulators. Further, continuous improvement of engineering strategies, processes, information, and information systems is paramount to the evolution and success of the pharmaceutical company and to satisfying increasing regulatory expectations. Establishing such a culture is not easy in an area that has traditionally not recognized the need for such control and documentation. Investment in change management programs may be needed to bring about such cultural change; ultimately, though, the senior management must lead by example.

REFERENCES

1. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (www.ispe.org).
2. European Union Guide to Directive 91/356/EEC (1991), *Directive Laying Down the Principles of Good Manufacturing Practice for Medicinal Products for Human Use, European Commission.*
3. Rules and Guidance for Pharmaceuticals Manufacturers (2002), Her Majesty's Standards Office (U.K.).

4. U.S. Code of Federal Regulations Title 21, Part 210: *Current Good Manufacturing Practice in Manufacturing, Processing, Packaging, or Holding of Drugs (General)*; Part 211: *Current Good Manufacturing Practice for Finished Pharmaceuticals*, The Office of the Federal Register, National Archives and Records Administration.
5. Wingate, G.A.S (1997), *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice*, Interpharm Press, Buffalo Grove, IL.
6. Owen, P. and Gough, P. (1997), Validating Engineering Management Systems, in *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice* (Ed. G.A.S. Wingate), Interpharm Press, Buffalo Grove, IL.

32 Case Study 14: Spreadsheets

Peter Bosshard, F. Hoffmann-La Roche

CONTENTS

Spreadsheets provide a means of organizing and manipulating data. Tremendous functionality and ease of use have made spreadsheets very popular and they are widely used in the pharmaceutical and healthcare industry.

Spreadsheets are often used to maintain recipes or analytical methods that directly include calculations. This may be very helpful to overcome the disabilities of LIMS systems or the poor functionality of an ERP. Analytical method validation, calculations of cleaning validation, and calibration lists are other areas where spreadsheets may be used within the pharmaceutical industry. This category of spreadsheet typically takes the form of preconfigured templates. Before spreadsheets were so widely used, programmable pocket calculators were used to perform such functions.

The powerful macro language capabilities of spreadsheets support a second category of application: the control of simple analytical devices or production equipment. Such applications are again typically based on preconfigured templates and functions.

Spreadsheets are also sometimes used as simple databases. Although their functionality is not comparable with relational databases the ease of use for simple data structures makes it a viable alternative.

Finally, spreadsheets may also be used in an *ad hoc* fashion for preliminary analysis of data and investigations. Pivot table reports are a feature of some spreadsheet software that facilitates data mining.

DIFFERENT SPREADSHEET APPLICATIONS

There is a broad variety of spreadsheets available for use with different operating systems.

LINUX APPLICATIONS

The operating system Linux has a reputation for robust server operation. A choice of various office applications are available, including GNUMERIC for Gnome, Hancom Office, KOffice, OpenOffice, and the new StarOffice 6.[1]

GNUMERIC is a comprehensive product. Hancom Office, a South Korean product, is similar to StarOffice 6.0. The spreadsheet and word processing are not well aligned. Some commands have different functions, e.g., the tools menu in the word processor is initiated with ALT+K while in the spreadsheet ALT+T invokes the same function. KOffice has the least functionality but here the word processing and the spreadsheet align. StarOffice has been developed from OpenOffice by the open source community to include ADEBAS database and some English, Spanish, French, and German dictionaries.

WINDOWS APPLICATIONS

Six popular spreadsheet applications are Lotus 1-2-3 of Lotus SmartSuite, Excel from Microsoft, PlanMaker of SoftMaker Office, Quattro Pro of Corel WordPerfect Suite, Spreadsheets of Applixware, and StarCalc of StarOffice, see Table 32.1. These applications were tested for how they cope with calculations, scientific graphics, calculations with the time, and usability as simple databases.[2] Testing did not uncover any calculation errors. Calculations using floating-point algorithms had a precision of more than 14 digits. These algorithms had minimal rounding errors that are well accepted in the scientific community.

Excel 97 had some problems with automatic recalculation, which were corrected with the service release SR-2. However, it still behaves erratically if a number is formatted as text. To change text to a number, the content of the cell has to be deleted, the cell is then formatted as a number, and then the content has to be reentered into the affected cell again. Testing also revealed a recalculation error in PlanMaker. If a number in cell A1 is changed, a formula containing the value of cell A1 is correctly updated, after "undo." However, only the value of A1 is changed back and not the calculated value. This also implies that consecutive calculations are not corrected after such a change.

Another problem might be caused by the fact that the displayed value and the effective value of a cell can be different. For instance, comparing an analytical specification with the result, the outcome may be misleading. The comparison IF(A1> = 90%;Pass;Fail) gives for A1 = 89.9 a "fail" even though the screen and the printout show 90%, which would indicate that the result was within the specification. To overcome these problems Excel and StarCalc offer to calculate with the values as displayed or printed out (see Figure 32.1).

As spreadsheet applications are usually in a bundle with the corresponding word processor, most users do not compare the features of the spreadsheet application but rather consider the overall offering of the complete office package. For the rest of this chapter all the samples are based on Excel 97 (Excel Version 8.0h).

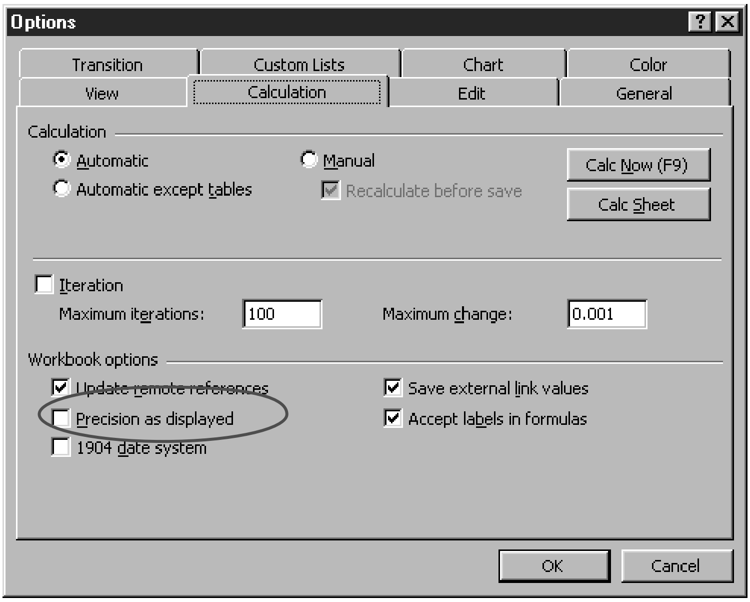
TABLE 32.1
Comparison of Six Spreadsheet Applications

Product	Applixware Spreadsheets 4	Quattro Pro 8	1-2-3 Version 9	Microsoft Excel 97	PlanMaker 97	StarCalc 5.0 Personal Edition
Producer	Applix	Corel	Lotus	Microsoft	SoftMaker	Star Division
Version	4.41	8.0.470	N9.0.9805.2800	SR-2	December 98	Unavailable
			General			
Multiple Undo	Unavailable	Unavailable	Unavailable	Available	Available	Available
Autofill with Mouse	Unavailable	Available	Available	Available	Available	Available
			Finance			
Calculation 160 × 5,98 – 956,80 =	1.1×10^{-13}	1.1×10^{-13}	1.1×10^{-13}	0	0	0
Decimal Rounding	Function ROUND	Function ROUND	Function ROUND	Choice	Function ROUND	Choice
Precision digits	16	16	16	15	19	14
Euro Currency	1	User defined[1]	Available	User defined	Unavailable	User defined
Update data from the Web	Unavailable	Macro	Available	Available	Unavailable	Macro
Scenario, Versions, What-If-Analysis	1	Available	Available	Available	+	Available
Goal seeking	1	Available	1	Available	1	1
Iteration	Unavailable	Unavailable	Available	Available	1	Available
Notification for circular references	Error message	No error message, dependency check with arrow diagram	Unavailable	Warning message, dependency check with arrow diagram	Error message	Warning message, dependency check with arrow diagram
Excel-Import	Up to Version 97[1]	Up to Version 97[1]	Up to Version 97	Up to Version 97	Up to Version 4.01	Up to Version 97[1]
Excel-Export	Up to Version 95[1]	Up to Version 97	Up to Version 97[1]	Up to Version 97	Unavailable	Up to Version 97
			Curves			
Halve logarithmic x-y-Plot	+	Available	Available	Available	Unavailable	Available
Equalizer lines/curves	+/+1	only formela/ Unavailable	Available/Available	Available/Available	Available/ Unavailable	Available/Available 1
Diagram Clipboard format (Bitmap/Vector)	1/unavailable also export to file	Available/ Available 1	Available/Available	Available/Available 1	Unavailable/ Available 1	Available/Available 1

TABLE 32.1 (Continued)
Comparison of Six Spreadsheet Applications

Product	Applixware Spreadsheets 4	Quattro Pro 8	1-2-3 Version 9	Microsoft Excel 97	PlanMaker 97	StarCalc 5.0 Personal Edition
Time Calculations						
Times outside 0…24 h	Unavailable	Unavailable	Unavailable	Available	Available	Available
Entry of two-digit years	Flexible	1951…2050	1900…1999 or flexible	1930…2029	1900…1999	Flexible
DATUM-Function with two-digit date	Flexible	1900…1999	1900…19991	1900…1999	1900…1999	Flexible
Databank						
ASCII-Import	Separator characters, preview	Separator characters	Separator characters or fixed with	Separator characters, preview	Separator characters	Separator characters
Calculation of formulas an formats with ASCII-import	Available/ Unavailable	Unavailable	Unavailable	Flexible	Available/Available	Flexible
ASCII-Export	Available	Available	1	Available	Available	1
HTML-Import/Export	Unavailable	1	Available/Available 1	Available 1/Available	Unavailable	Available/Available
Entry form	Unavailable	Unavailable	(Approach)	Available	Available	Unavailable
Validity check/rules	Unavailable	1	Unavailable	Available	Unavailable	Available
Filters	1	Auto or manual	(Approach)	Auto or manual	Auto or manual	Auto or manual
1 for errors and constraints (see text)						
2 largest *n*, for which $(1 - 10 - n)^{-1}$ is not equal to 0						

1 = Reference 1.



Calculation

● Automatic ○ Manual

○ Automatic except tables ☑ Recalculate before save

□ Iteration

Maximum iterations: 100 Maximum change: 0.001

Workbook options

☑ Update remote references ☑ Save external link values

□ Precision as displayed ☑ Accept labels in formulas

□ 1904 date system

FIGURE 32.1 "Precision as Displayed" Feature.

TABLE 32.2
Classification Strategy of Different Spreadsheet Applications

GAMP Software Category	Spreadsheet Complexity	Restrictions in Use of Spreadsheet
3	Simple	Used only once for ad hoc reports One-time graphical evaluation of data Manual review of the outcome No direct influence on the product quality
4	Simple	Using only standard functionality Not more than two bracelets No VBA-Macros are used
5	Complex	No restrictions

VALIDATION APPROACH — SIMPLIFIED LIFE CYCLE, SPECIAL CONSIDERATIONS

Classification of Spreadsheets

Spreadsheet applications can be classified based on their complexity and functionality. Depending on the classification, different approaches to validation can be taken. Table 32.2 specifies restrictions in use of a spreadsheet that should exist for validation to be based on a particular GAMP category of software[3] to apply.

General Requirements for the Validation

Spreadsheet applications need to be protected from inadvertent or unauthorized changes. Applications without database functionality should be preserved in a write-protected directory, where only a very limited number of people have access and all changes are documented.

- Applications with database functionality are to be classified as complex applications.
- All applications for the generation of electronic records need to have the audit trail switched on.
- Templates should be considered to be used for repeated jobs.
- Templates should not contain any data; this prevents incorrect entries that result when the user forgets to replace the template data.
- After the validation and release, the spreadsheets need to be brought to a restricted folder.
- The retired spreadsheets need to be archived so that the user cannot erroneously take a retired one.
- Calculations are to be performed on the sheet and not in VBA routines.
- Entries should be tested for plausibility.
- Every worksheet and printout must list the Spreadsheet and OS version that was used to calculate the results.
- All calculations need to be designed in a clear and traceable way.

To help address these points, the following recommendations are made:

- Entry fields should be marked with a specific background color to facilitate the localization. Names of the spreadsheets should be clear and unique in an area to prevent the users from taking the wrong ones.
- The spreadsheet could provide lists for the entry of data that might be entered with variations, e.g., Capsules, Capsule, Caps, CPS, Kapseln, Kap, etc.
- The file names could include a version number.
- Entries and results that are out of specification should be highlighted using conditional formatting.
- Unused worksheets could be deleted.
- If it is impossible to design entries as self-explanatory, other measures such as detailed SOP and Help File should be generated.
- Consider taking names to increase the readability of the formulas.

Design Principles

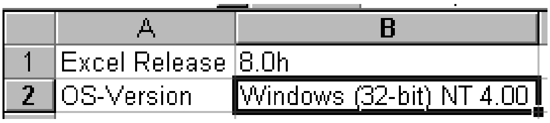
Design is key to successful spreadsheet compliance.

Indicate Release and Operating System Version

Shortly after the release of Excel 97, several errors were reported by the user community. These errors included problems with calculation and updating of calculation fields. Since that time, users are well advised to indicate on a worksheet with which version something was calculated (see Figure 32.2). The formula = INFO("release") and = INFO("OsVersion") give the result.

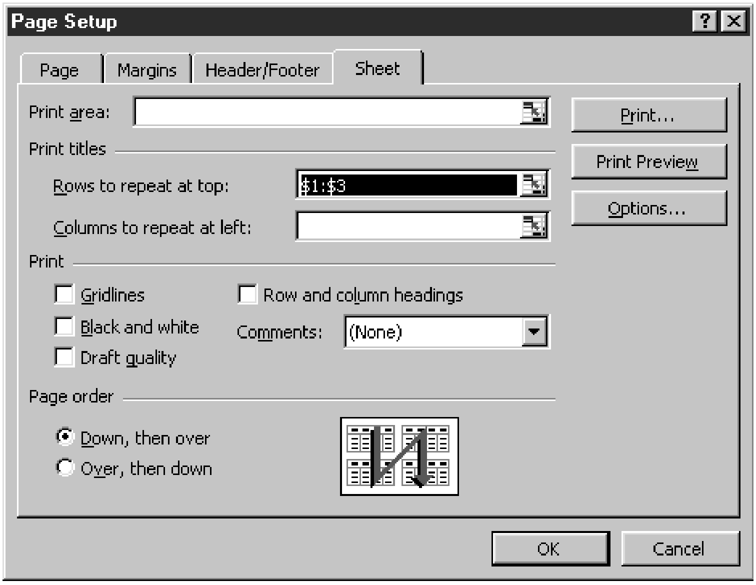
Print Headings on Every Page without Macros

With macros practically everything can be done in a spreadsheet. However, because macros catapult the class of the spreadsheet from 3 to 5 and lead to much more validation and maintenance work,



	A	B
1	Excel Release	8.0h
2	OS-Version	Windows (32-bit) NT 4.00

FIGURE 32.2 Display Release and OS Version.



Print area:

Print titles

Rows to repeat at top: $1:$3

Columns to repeat at left:

Print

□ Gridlines □ Row and column headings

□ Black and white Comments: (None)

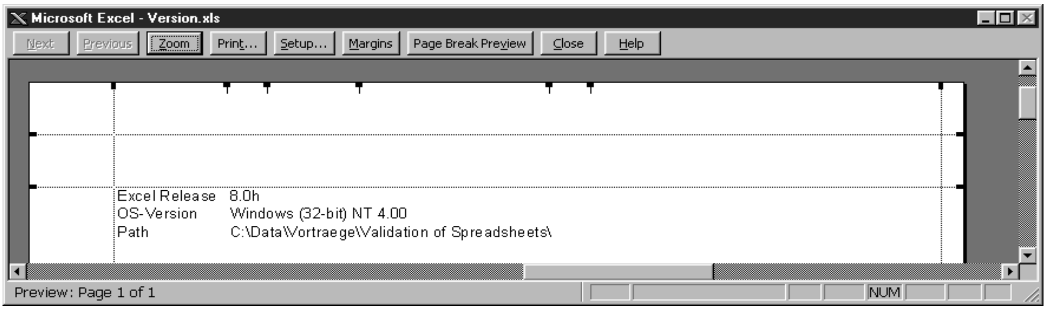
□ Draft quality

Page order

● Down, then over

○ Over, then down

FIGURE 32.3 Repeat a Part of the Spreadsheet on Every Page.



```
Excel Release   8.0h
OS-Version      Windows (32-bit) NT 4.00
Path            C:\Data\Vortraege\Validation of Spreadsheets\
```

Preview: Page 1 of 1

FIGURE 32.4 Printout of Repeated Information.

there should be a way to have this information available without programming. To repeat the printout of the version information and the file path on every page in the footer or header, the following command might be useful: = *Info("directory")*. Choose menu File, Page Set up, on the Tab Sheet; one may enter the number of rows that need to be repeated at the top of each page (see Figure 32.3). The command results in the printout shown in Figure 32.4.

Validation Approach for Different Classes

The validation measures that need to be taken for the different classes of spreadsheet are shown in Table 32.3.

Procedure for Class 3 Spreadsheets

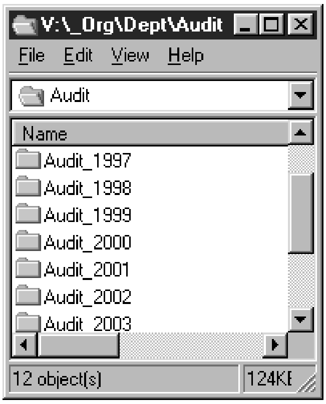
For this class the spreadsheet version should be documented on the printout. The printout is separately reviewed and if GMP or SOP foresees a signature is placed on the printout. Examples include CSV-Inventories and task lists of open complaints.

Electronic Records should be handled carefully as if they were paper. Spreadsheets are electronic records if they are used directly for a GMP-related task. This includes analytical worksheets,

TABLE 32.3
Validation Measures for Different Classes of Spreadsheet

	GAMP Software Category		
Measure	**3**	**4**	**5**
Procedure for usage	x	x	x
Training of the user	x	x	x
File cabinet restrictions	x	x	x
Switch on Audit Trail	x	x	x
Indicate OS-Version and Spreadsheet Program	x	x	x
Print heading on every page	x	x	x
Describe general user requirements. Input, Calculation, Output		x	x
Establish Validation Plan and get it approved		x	x
Classify system		x	x
Inventorize system		x	x
Develop Workbook		x	x
Design of the fields		x	x
Traceable calculation		x	x
Protection		x	x
Testing		x	x
Print formulas and review against URS		x	x
Enter Test-Data and review against expected results		x	x
Check field protection		x	x
Check every plausibility testing		x	x
Check all conditional formatting		x	x
Check against implausible entries		x	x
Establish Validation Report and get it approved — system released		x	x
Move to protected area and start Change Control		x	x
Approval by QA		x	x
Developer training			x
Vendor assessment (Supplier Audit)			x
Functional Specification			x
Interfaces			x
System borders			x
Algorithms			x
Follow guidelines and programming standards for development			x
Perform Source Code review			x
Test plan created from the system responsible against the functional specification			x
Review of the test plan of the system owner			x
Approval by QA			x
Testing and Review, QA approval			x
Inspection of the validation documentation			x
Establish validation report, QA approval			x
Move application to productive environment			x

calculations that are performed ad hoc to investigate problems or any other record that is directly related to a product. Spreadsheets are also electronic records if they are used to support a GMP-regulated task or if they are part of an SOP. Examples would be calibration lists and inventories of equipment. Further, spreadsheets are electronic records if an SOP requires that a spreadsheet be established: for example if an SOP demands that internal audits should be planned based on a spreadsheet then this spreadsheet becomes an electronic record.



Audit_1997
Audit_1998
Audit_1999
Audit_2000
Audit_2001
Audit_2002
Audit_2003

12 object(s) 124KI

FIGURE 32.5 Annual Organization of Data Files.

Electronic records should be saved in a well-designed directory structure that is strictly controlled. This includes measures such as backup and access restrictions. An example organization of data is shown in Figure 32.5. General security and administration are discussed in Chapter 12 of this book.

Procedure for Class 4 Spreadsheets

Validation requirements in addition to Class 3 spreadsheets exist for Class 4 spreadsheets. These additional activities can be divided into planning, design, testing, and reporting.

A Validation Plan should be established and approved. A User Requirements Specification should be prepared describing the general user requirements, input, calculations, and output from the spreadsheet. The classification of the spreadsheet could be defined in the Validation Plan, URS, or separately according to a defined SOP. The use of the spreadsheet as a GxP application should be logged in the site or facility system register.

A workbook capturing the design of the spreadsheet then needs to be prepared. This should include the design of fields, traceable calculations, and data protection mechanisms.

Testing should be planned in a protocol and a subsequent report produced after testing is complete. Formulas should be printed and compared to the URS. Test data should be entered to check calculations and data manipulations against expected results. Field protection setup should be confirmed. Plausibility checks on conditional formatting for cell entries should be conducted.

Finally a Validation Report should be prepared in response to the Validation Plan to summarize the outcome of validation. Approval of the Validation Report authorizes release of the spreadsheet for use. The spreadsheet should be moved to a protected area to prevent unauthorized or unintentional changes being implemented, and a formal change control process initiated.

Procedure for Class 5 Spreadsheets

Validation requirements in addition to Class 4 spreadsheets exist for Class 5 spreadsheets. These additional activities can be divided into supplier assessment, detailed design, testing traceability, and training.

A supplier assessment may be necessary if a third party is being used to develop the spreadsheet application. Supplier Audits for COTS software packages are not required.

A detailed design is required. A Functional Specification should be written covering interfaces, system boarders, algorithms, macros, and data structures. The definition of macros should include a description of all the variables used, template design, VBA-Architecture, and VBA-Codes. The definition of data structures should include Entity Relationship Diagrams and data field descriptions.

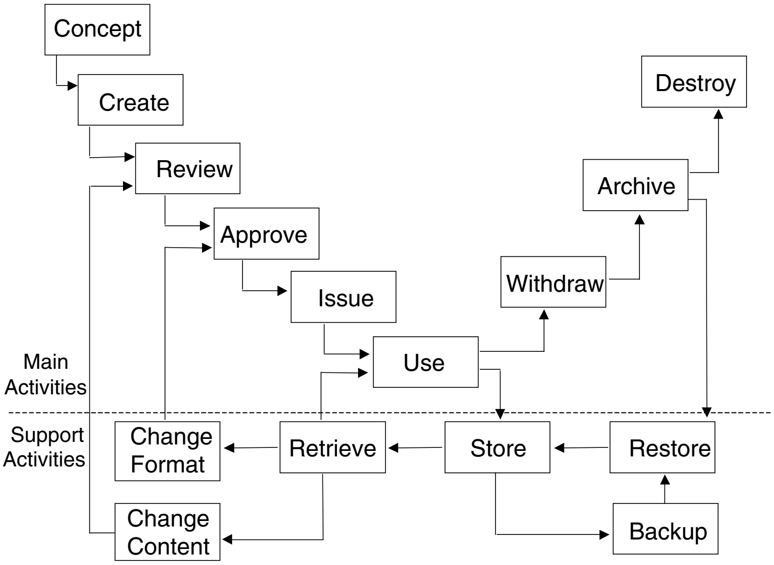


FIGURE 32.6 A Typical Life Cycle of a Spreadsheet as Defined by GAMP.

The development of templates and macros should follow predefined guidelines and programming standards. A Source Code Review will be appropriate for macros.

With the increased complexity of the spreadsheet application, and the hierarchy of specification and design documentation, it becomes very important to ensure there is good traceability between requirements and testing. For this reason the use of a Requirements Traceability Matrix (RTM) is recommended.

Finally, since this class of spreadsheet is more complex, it is expected that some formal training will be required for particular applications. Training materials to accompany user SOPs will be required and should be approved prior to use. Training records should log any training received.

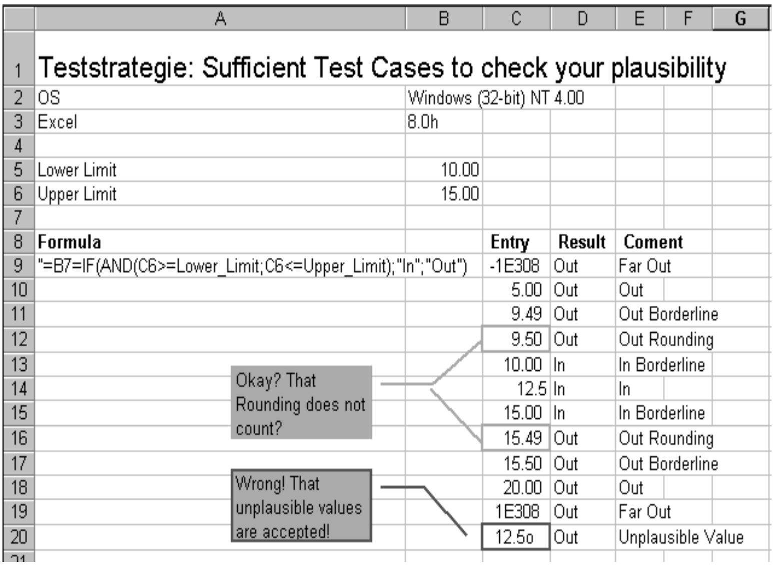
Development Standards for Spreadsheets and Macros

Knowledgeable analysts are often developing spreadsheets in the lab where they are used. This helps ensure that applications are user friendly and as required by Annex 11 of the EU-GMP have the necessary user involvement.[4] Nevertheless, there should be a development standard defined for spreadsheet applications. Such a standard could basically follow the life cycle presented in Figure 32.6.[5] External developers should also follow equivalent standards.

For legacy spreadsheets, a review of the documentation and any additional documentation and testing should be considered.

Protection of Workbooks

Users should only have read access to spreadsheet applications. All templates should be password protected. The password should not be identical with the NT password because the spreadsheet password does not age. The system owner needs to administer the passwords. If there is no authorization via the operating system (insular workstations), the system owner should define and administer the specific access authorization. Group access should not be allowed. Only cells receiving an entry should be unprotected (unlocked). Macros need to be hidden and protected from user manipulation.



	A	B	C	D	E	F	G
1	Teststrategie: Sufficient Test Cases to check your plausibility						
2	OS	Windows (32-bit) NT 4.00					
3	Excel	8.0h					
4							
5	Lower Limit	10.00					
6	Upper Limit	15.00					
7							
8	**Formula**		**Entry**	**Result**	**Coment**		
9	"=B7=IF(AND(C6>=Lower_Limit;C6<=Upper_Limit);"In";"Out")		-1E308	Out	Far Out		
10			5.00	Out	Out		
11			9.49	Out	Out Borderline		
12			9.50	Out	Out Rounding		
13			10.00	In	In Borderline		
14			12.5	In	In		
15			15.00	In	In Borderline		
16			15.49	Out	Out Rounding		
17			15.50	Out	Out Borderline		
18			20.00	Out	Out		
19			1E308	Out	Far Out		
20			12.5o	Out	Unplausible Value		

Okay? That Rounding does not count?

Wrong! That unplausible values are accepted!

FIGURE 32.7 Plausibility Checking.

Testing of Applications

Test Strategy

Wherever possible, a copy of real-life data should be used for testing. Tests should be performed on the finished saved worksheet. The file date and size do not change afterward. All the formulas need to be checked. All the calculations should be verified using a pocket calculator. All ranges (plausibility checking) need to be checked (e.g., a range of 10 to 15 needs to be checked with the following values: 1E308, 5, 9.49, 9.50, 10, 12.50, 15, 15.49, 15.50, 20, 1E308, 12.50). Figure 32.7 illustrates that implausible values are not rejected and the spreadsheet needs to be improved.

Document the Cell Protection

Do a printout on a standard printer to check the cell format. Print out all the formulas. Sign and date all prints and ensure cross references to test protocol.

ELECTRONIC RECORD/SIGNATURE CONSIDERATIONS

Twelve basic requirements for electronic records and signatures are defined in U.S. regulation 21 CFR Part 11 concerning electronic records and electronic signatures. Table 32.4 suggests necessary actions to comply with spreadsheet applications.

Switch on the Audit Trail

Appendix 32A outlines a process for switching on audit trails to record all the changes to a worksheet in Excel spreadsheets. If the audit trail is already switched on during the development it records also the history of the XLT-Template file.

Example Regulatory Observations

FDA inspection findings can be found at www.fda.gov if they have been escalated to a Warning Letter. Such observations are sometimes difficult to understand and to set into the correct context.

TABLE 32.4
Twelve Basic ERES Requirements

Requirement	Actions to Comply
1. Validation	Only for classes 4 and 5
2. Audit trail	
3. Discerning of changed records	Changed records are flagged with a blue triangle as soon as the audit trail is switched on
4. Authorization	To be done over the file access rights (see Chapter 11)
5. Authority check	To be handled over system access
6. Equipment authority check	Not relevant
7. True and complete copy	Is given by the spreadsheet application itself
8. Storage during document retention	To be handled over file system
9. Enforce permitted sequence	Only relevant with macro programming
10. Measures against falsification	File protection
11. Training of user and developer	Procedural
12. Control over development documentation	Procedural

Observations may occur because there is a misunderstanding between the inspector and the person explaining the matter. Observations may also occur because the auditee guides the inspector to fields that are easier to correct just to keep the inspector busy in an area where no major observations are expected. Table 32.5 lists some recent FDA Warning Letters and suggests suitable remedial actions.

TABLE 32.5 Example Warning Letters

Warning Letter	Observation	Proposed Remedy
Earlham College[6] July 29, 2002	(FORM FDA-483 Item #14) There is no provision in the test method procedure for the use of an [redacted] spreadsheet for entering test parameters and calculation of test results that was used to calculate the results of blend samples of Vitaroca and Vinatal from 5/6-13/02.	Use spreadsheets only in a controlled way. Inventorize spreadsheets and reference them in the method.
Cardinal Health, Inc.[7] July 10, 2001	Failure to have an adequate validation procedure for computerized spreadsheets used for in-process and finished product analytical calculations. The current validation procedure uses only the values that result in within specification findings, aberrant high findings, and aberrant low findings [21 CFR 211.165(e)]. For example, SOP 644.00, QA/QC Spreadsheet Validation, is deficient in that only a small range of values are being used to challenge computerized spreadsheet mathematical calculations.	Have an appropriate procedure available to validate spreadsheets. Do not filter out OOS values. Design the spreadsheets in a logical fashion. Challenge the capability to calculate with OOS values during validation testing.
	Failure to use fully validated computer spreadsheets to calculate analytical results for in-process and finished product testing [21 CFR 211.165(e)]. For example, the computer spreadsheets used to calculate analytical results for [redacted] have not been validated.	Validate spreadsheet.
	Regarding the validation of computerized spreadsheets used for in process and finished product analytical calculations (FDA-483, #4), your response states that current spreadsheets were challenged using the proposed revisions to SOP 644, QA/QC Computer Spreadsheet Validation. However, your response does not indicate if computerized spreadsheets for all products which use the spreadsheets were challenged using the proposed revisions to SOP 644.	Inventorize spreadsheets and test all spreadsheets in a controlled manner.
	Regarding the failure to use fully validated computer spreadsheets to calculate analytical results for in-process and finished product testing (FDA-483, item #5), your response states that old spreadsheets will be revalidated according to the proposed revisions to SOP 644 prior to being implemented into use. You identify that SOP 644 will not be revised until July 20, 2001.	Give priority.
	This response is not acceptable. Any validation studies performed must be performed using an approved revision to your SOP, validating using a proposed SOP revision is not an acceptable practice.	

TABLE 32.5 (Continued)
Example Warning Letters

Warning Letter	Observation	Proposed Remedy
EP MedSystems[8] July 10, 2001	Your firm failed to validate several computer databases that are used for quality functions including your Access database, your … software, and your MS Excel spreadsheet program as required by 21 CFR 820.70(i).	Validate spreadsheets, access databases, and other COTS.
Hospital for Special Surgery[9] March 21, 2002	On 10/24/01, a spreadsheet of subjects was provided by your office in response to FDA's request. The document was unlabeled and provided a subject/medical record/device trail but not the reverse. The accountability records expected as part of an investigator's study file include records of the receipt and use or disposition of all investigational devices received from the sponsor.	Follow a procedure like that for paper when giving electronic records to the FDA.
B. Braun Medical Ltd.[10] November 7, 2002	… software validation plan does not address the user requirements of inputting data into the … spreadsheet used as a tool for trending.	
Drager Medizintechnik GmbH[11] August 6, 1999	Failure to validate computer software used as part of the quality system for its intended use according to an established protocol as required by 21 CFR 820.70(i). For example, the data in the Excel spreadsheet identified as a "hit list" of top nonconforming components contains 16 record counts for part number 8601618 DC converter failures compared to 18 record counts for part number 860168 DC converter failures in the dbase database. The spreadsheet is used for management review of component suppliers for all components.	There is no obvious Excel problem. The redundant entry into dbase and Excel seems to be inconsistent. Have only one source of data and ensure that it is reliable.
B. Braun Medical, Inc.[12] April 29, 1999	Your response indicated that Braun is currently changing the complaint handling system from tracking complaint information on a … spreadsheet to using an off-the-shelf database system … tracker. As required by 21 CFR 820.70(i), Automated Processes, this off-the-shelf software shall be validated for its intended use if Braun has not already done so.	Validate spreadsheets; if going to alternative solutions validate these, too.

Purepac Pharmaceutical[13] November 26, 1997	There are insufficient controls of the integrity of calculated data generated by the software in the Quality Control Laboratory, in that: • There is no audit trail to track the number of templates accessed to generate data calculations. • Password protection can be bypassed in the system. • Data files are automatically deleted after a hardcopy is generated. There is no requirement to identify the analyst or time/date stamping of spreadsheet hardcopies.	Switch on the audit trail. Use a secure system. Difficult. To my understanding 21CFR211.68 allows such automated deletion. "… A backup file of data entered into the computer or related system shall be maintained except where certain data, such as calculations performed in connection with laboratory analysis, are eliminated by computerization or other automated processes." By the way, automated deletion is standard for many temporary files such as printer spooler files.
Willi Eye Associates[14] July 7, 1998	You failed to investigate the failure of the … when operating in MS Access. The system locks up at random and it is unknown whether the software which controls the… during …which operates off of MS Excel, could be similarly affected. Disruption of the … or an incorrect … pattern could result from such an occurrence.	Validate the application and review the user requirements.
Medical Industrial Equipment Ltd.[15] June 8, 2000	Failure to validate computer software used as part of the quality system for its intended use according to an established protocol as required by 21 CFR 820.70(i). For example: Software such as Excel, Access, and Word used to create and maintain data bases (rejects, complaints, and concessions) and electronic documents, is not validated.	Validate Excel Spreadsheets.

REFERENCES

1. Brors, D., Diedrich, O., and Büro, T. (2002), Fünf Office-Pakete für Linux im Vergleich, c’t 8, pp. 188 ff.
2. Loviscach, J. (1999), Zahlenspiele-Tabellenkalkulationen im Praxistest, c’t 3, pp. 98 ff.
3. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP4), published by International Society for Pharmaceutical Engineering (www.ispe.org).
4. EU-GMP, Annex 11, “Computerised Systems,” Principle http://pharmacos.eudra.org/F2/eudralex/vol-4/pdfs-en/anx11en.pdf.
5. ISPE/PDA (2001), “Good Practice and Compliance for Electronic Records and Signatures,” Part 2, International Society for Pharmaceutical Engineering.
6. FDA (2002), Warning Letter to Douglas Bennett, Earlham College, 801 National Road, West Richmond, IN 47374, July 29, http://www.fda.gov/foi/warning_letters/g3419d.pdf.
7. FDA (2001), Warning Letter to Robert D. Walter, Cardinal Health, Inc. 7000 Cardinal Place, Dublin, Ohio 43107, July 10, http://www.fda.gov/foi/warning_letters/g1485d.pdf.
8. FDA (2001), Warning Letter to Mr. David A. Jenkins, EP MedSystems, Bldg. D, 575 Route 73 N., West Berlin, New Jersey 08091, July 10, http://www.fda.gov/foi/warning_letters/g1483d.pdf.
9. FDA (2002), Warning Letter to Mark P. Figgie, Hospital for Special Surgery, 535 East 70th Street, Suite 328, New York 10021, March 21, http://www.fda.gov/foi/warning_letters/g3160d.pdf.
10. FDA (2002), Warning Letter to Mr. Peter Mitchell, B. Braun Medical Ltd., Thorncliffe Park, Sheffield S35 2PW, U.K., November 7, http://www.fda.gov/foi/warning_letters/g3667d.htm.
11. FDA (1999), Warning Letter to Dr. Peter Gebhardt, Drager Medizintechnik GmbH, Moislinger Allee 53-55, 23542 Lubeck, Germany, August 6, http://www.fda.gov/foi/warning_letters/m2848n.pdf.
12. FDA (1999), Warning Letter to Mr. Gale White, B. Braun Medical, Inc. 1601 Wallace Drive, Suite 150, Carrollton, Texas 75006, April 29, http://www.fda.gov/foi/warning_letters/m2579n.pdf.
13. FDA (1997), Warning Letter to Mr. Richard F. Moldin, Purepac Pharmaceutical Co., 200 Elmora Avenue, Elizabeth, New Jersey 07207, http://www.fda.gov/foi/warning_letters/m699n.pdf.
14. FDA (1998), Warning Letter to Amos J. Willis, Willis Eye Associates, Rappahannock Eye Center, 10 White Oak Road, Fredericksburg, Virginia 22405, July 7, http://www.fda.gov/foi/warning_letters/d1902b.pdf.
15. FDA (2000), Warning Letter to Mr. Paul Roderique, Medical Industrial Equipment Ltd., Liverton Business Park, Salteron Road, Exmouth, Devon EX82NR, U.K., June 8, http://www.fda.gov/foi/warning_letters/m3851n.pdf.

APPENDIX 32A
PROCESS TO ENABLE SPREADSHEET AUDIT TRAILS

Many users are unaware that a standard audit trail facility is available in some spreadsheet products. This appendix explains how such audit trail facilities can be used and notes their limitations.

The following steps can be used to configure the audit trail facility in Microsoft Excel spreadsheets:

1. Select Menu Tools/Share Workbook.
2. Select Share Workbook.
3. On the Tab "Advanced" you can now enter the time, how long the audit trail can be saved. The field accepts entries from 0 to $2^{15}-1$, which would be approximately 91 years.
4. Now the activated audit trail needs to be switched on over the menu Tools, Track Changes, Highlight Changes.
5. Select visibility of the audit trail in the new form that appears. There are different filter options that can be easily selected from the corresponding drop down.
   If the option "Highlight changes on the screen" is selected, all the changed records are discerned with a blue label. Additionally, the last change can be seen when the mouse is placed over the cell.
6. An additional possibility is to create a list of the audit trail. In this instance, you must save the workbook first.

This audit trail facility does have some limitations. For instance, color changes are not recorded; this might be quite important if color changes are used to alert users to certain alarm conditions. Where such dependencies would be an issue, they should be designed out. Another limitation is that if entries are changed to blank and then back to another entry the blank status is not mentioned in the audit trail. This is not important if the transition through the blank entry is all part of a single data change transaction. Another potential issue is that the user name is not captured automatically; rather, the user needs to enter his or her name manually via the menu tools option. This means a user could potentially input someone else's name (mistakenly or maliciously). This can be controlled to some extent by inserting the user name (e.g., the name of the Lab Analyst) as a data item in the spreadsheets to indicate the responsible person who will verify the audit trails as part of final approval. The user names in the audit trail should correspond to the user names defined in the spreadsheet. In simple spreadsheets like the one shown in Figure 32.14 only one user name will appear in the audit trail.

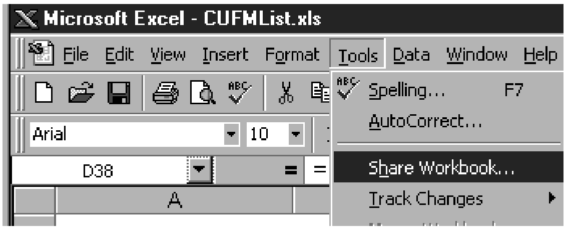
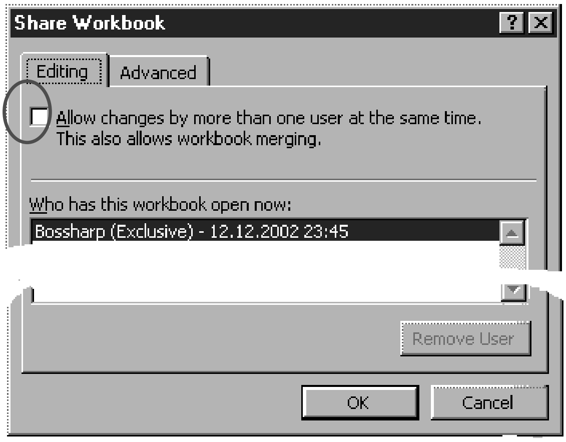

Microsoft Excel - CUFMList.xls
File Edit View Insert Format Tools Data Window Help
Spelling... F7
AutoCorrect...
Share Workbook...
Track Changes
Arial
10
D38
A

FIGURE 32.8 Selecting Tools.

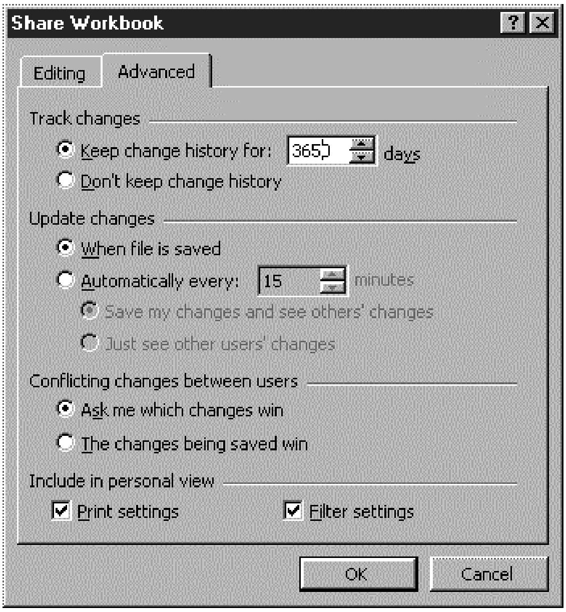


□ Allow changes by more than one user at the same time. This also allows workbook merging.

Who has this workbook open now:

Bossharp (Exclusive) - 12.12.2002 23:45

FIGURE 32.9 Configure Sharing.



Track changes

● Keep change history for: 365 days

○ Don't keep change history

Update changes

● When file is saved

○ Automatically every: 15 minutes

● Save my changes and see others' changes

○ Just see other users' changes

Conflicting changes between users

● Ask me which changes win

○ The changes being saved win

Include in personal view

☑ Print settings ☑ Filter settings

FIGURE 32.10 Advanced Tab Settings.

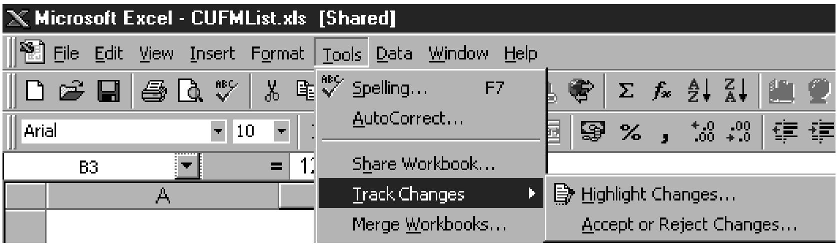
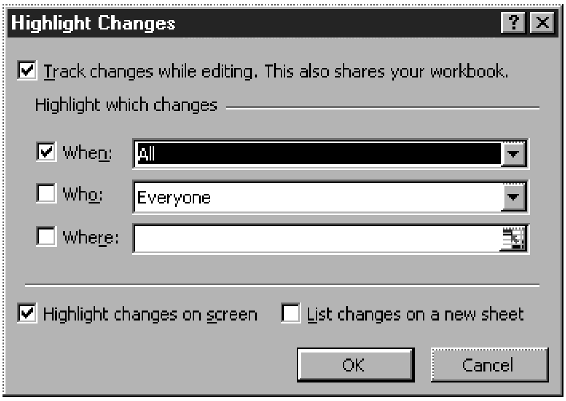

Microsoft Excel - CUFMList.xls [Shared]
File Edit View Insert Format Tools Data Window Help
Spelling... F7
AutoCorrect...
Share Workbook...
Track Changes
Merge Workbooks...
Highlight Changes...
Accept or Reject Changes...
Arial
10
B3
=
A

FIGURE 32.11 Track Changes.


Highlight Changes

☑ Track changes while editing. This also shares your workbook.

Highlight which changes

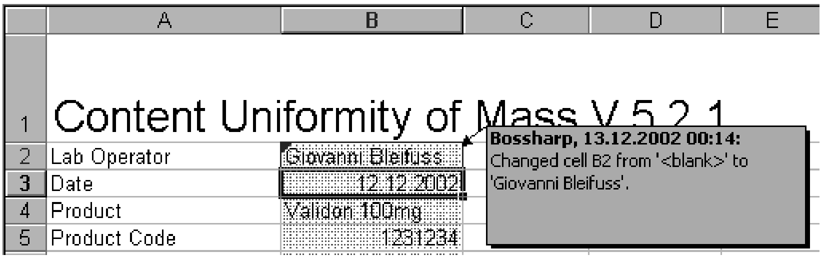
☑ When: All

☐ Who: Everyone

☐ Where:

☑ Highlight changes on screen ☐ List changes on a new sheet

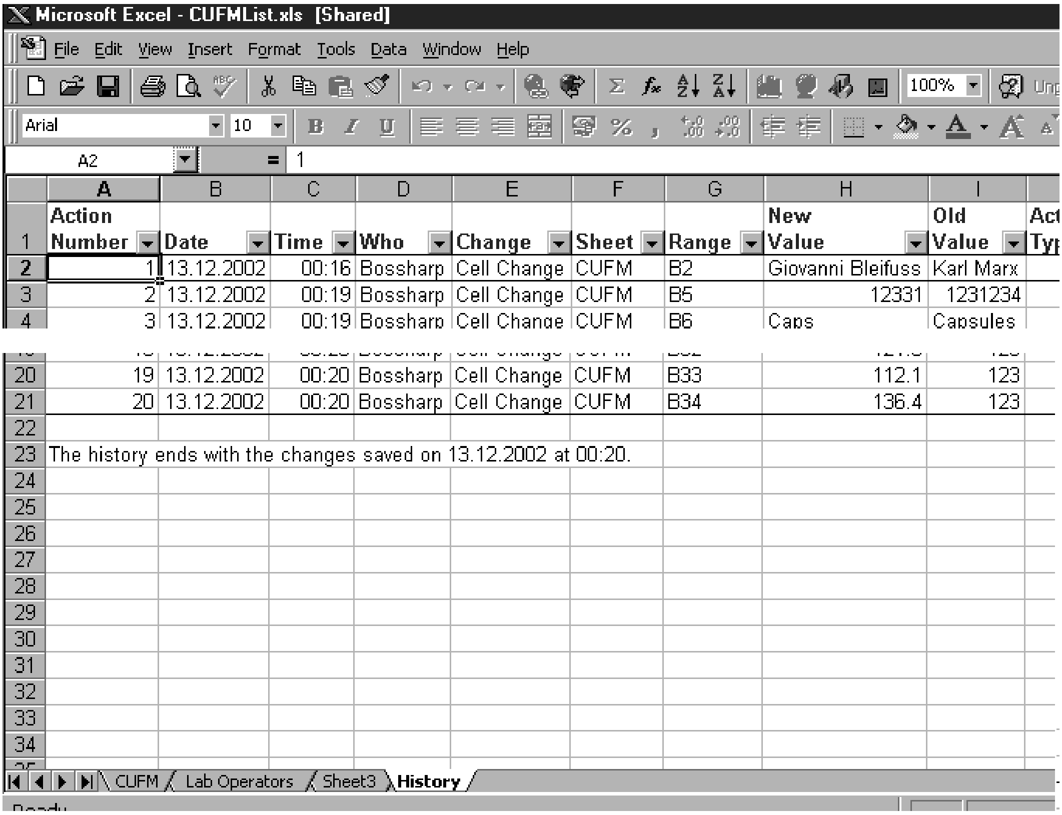
FIGURE 32.12 Highlight Changes.



	A	B	C	D	E
1	Content Uniformity of Mass V 5.2.1				
2	Lab Operator	Giovanni Bleifuss			
3	Date	12.12.2002			
4	Product	Validon 100mg			
5	Product Code	1231234			

Bossharp, 13.12.2002 00:14:
Changed cell B2 from '<blank>' to 'Giovanni Bleifuss'.

FIGURE 32.13 Example Audit Trail.


Microsoft Excel - CUFMList.xls [Shared]
File Edit View Insert Format Tools Data Window Help
100%
Arial
10
A2
CUFM
Lab Operators
Sheet3
History
Ready

A2 = 1

	A	B	C	D	E	F	G	H	I	
1	Action Number	Date	Time	Who	Change	Sheet	Range	New Value	Old Value	Act Typ
2	1	13.12.2002	00:16	Bossharp	Cell Change	CUFM	B2	Giovanni Bleifuss	Karl Marx	
3	2	13.12.2002	00:19	Bossharp	Cell Change	CUFM	B5	12331	1231234	
4	3	13.12.2002	00:19	Bossharp	Cell Change	CUFM	B6	Caps	Capsules	
20	19	13.12.2002	00:20	Bossharp	Cell Change	CUFM	B33	112.1	123	
21	20	13.12.2002	00:20	Bossharp	Cell Change	CUFM	B34	136.4	123	
22										
23	The history ends with the changes saved on 13.12.2002 at 00:20.									
24										
25										
26										
27										
28										
29										
30										
31										
32										
33										
34										

FIGURE 32.14 Audit Trail File.

33 Case Study 15: Databases

Arthur D. Perez, Novartis

CONTENTS

The *FDA Glossary of Computerized System and Software Development Terminology* published by the FDA in 1995 quotes an American National Standards Institute (ANSI) definition: a database is "a collection of interrelated data, often with controlled redundancy, organized according to a schema to serve one or more applications. The data are stored so that they can be used by different programs without concern for the data structure or organization. A common approach is used to add new data and to modify and retrieve existing data." Perhaps a more prosaic definition, and one that is probably much more readily understood by a large majority of computer system users, quality assurance organizations, and regulatory authorities, is simply that a database is a compilation of related data that is needed to support some activity.

The pervasiveness of electronic tools like the Microsoft Office Suite's Access®, Word®, and Excel® have placed the ability to build databases at the fingertips of many people who would never dream of trying to build an application using a more sophisticated tool like Oracle®. Users of such applications can make the mistake of not considering regulatory compliance. A perusal of FDA Warning Letters and 483s from the last few years, however, shows that the FDA, like other regulatory authorities, does inspect such applications.

The following list summarizes regulatory issues that can impact databases. Many of these are specifically addressed to the FDA's Final Rule on Electronic Records; Electronic Signatures (21 CFR 11).

- Verification of data load process.
- Limiting computer access to authorized individuals.
- Protecting data from unauthorized modification and destruction.
- Use of authority checks to determine if the identified individual has been authorized to use the system or device, or to access or perform a particular operation.
- Changing passwords periodically.
- Use of time-outs of terminals to prevent their unauthorized use while unattended.
- Use of security measures to protect against natural system failures.
- Use of time-stamped audit trails. The audit trail provides the capability to reconstruct the data that has been modified in order to prevent the previously entered data from being obscured.
- Use of record revision and change control to maintain configuration management.
- Use of operational checks to enforce permitted operational parameters such as functional sequencing.
- Use of device (location) checks to determine whether the physical source of the data or e-signature is valid.
- Facilities for electronic signatures where required by application.

Examples of recent FDA citations for noncompliance of database applications include comments regarding the deviation database currently maintained by the quality unit as an Excel spreadsheet file for monitoring the status of deviations and investigations:[1]

- The firm failed to put in place procedures defining or controlling the use of this database.
- The firm has failed to validate this database. [FDA 483, 2002]

Software such as Excel, Access, and Word used to create and maintain databases (rejects, complaints, and concessions) and electronic documents is not validated. [FDA Warning Letter, 2000]

No security system to prevent unauthorized changes to computer database used to print labels. [FDA 483, 2001]

DATABASE ARCHITECTURE

RECORDS AND FIELDS

The common characteristic of even these simplest of databases and of the most complex of their cousins is the concept of records. A record comprises the smallest collection of related data elements that is typically retrieved by a search. Again quoting the *FDA Glossary*, a record is "a group of related data elements treated as a unit. [A data element (field) is a component of a record, a record is a component of a file (database)]."

A good visualization is to think of a record as a line in a table, although it is not that simplistic in even moderately complex relational databases. In the Excel database cited above, a record might consist of a name for an investigation, a product batch number, a date the table entry was created, to whom the investigation was assigned, and a status such as "in progress," "under review," or "completed." None of these data elements means much in isolation, but when considered together they constitute an important collection of information.

As noted in the definition above, the individual data elements that make up records are typically referred to as fields. In the example above, the fields are the name, the date, and the status of the investigation that is the subject of each record.

Database Management Systems

When discussing validating databases, it is important to distinguish between a database and a Database Management System (DBMS). The DBMS is the layered software that provides the tools to build and use a database. For example, Oracle® and Microsoft Access® are two examples of a DBMS often used in pharmaceutical companies.

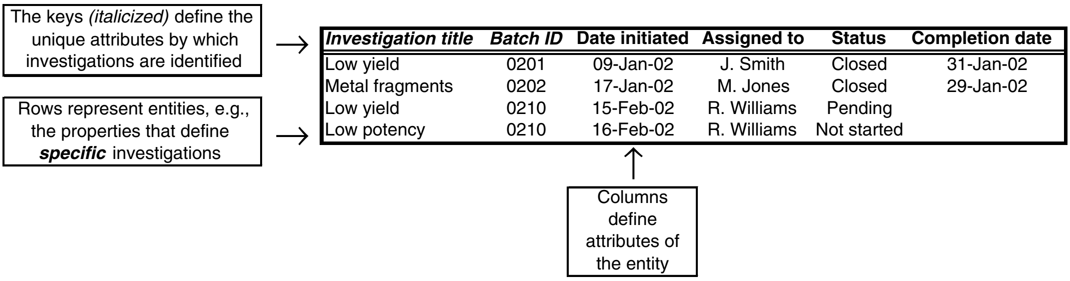
Types of Databases

The simplest kind of database is a flat file. (*FDA Glossary*: A flat file is "a data file that does not physically interconnect with or point to other files. Any relationship between two flat files is logical; e.g., matching account numbers.") Searching for records in a flat file is essentially a brute force task. In effect, the computer looks at all of the stored information in order to determine what records fit the criteria of the query. This is not very efficient, and it is a highly impractical search means for large amounts of data. Performing a sort on an Excel® spreadsheet or using the Edit/Find function are ways of doing this sort of search.

A more sophisticated database design is relational. This design makes use of defined relationships between data to vastly increase the efficiency of data retrieval. Its popularity is largely attributed to its relatively simple data model:

- Data is presented as a set of relations.
- Each relation is a data table.
- Columns within a table are data attributes.
- Rows represent entities possessing attributes.
- Tables have a set of attributes that when taken together uniquely identify each entity (a key).[2]

For example, consider the Excel table noted in the 483 referenced earlier. While it is not part of a relational database, it serves as a model for a table that could be part of one. Figure 33.1 shows how the table is organized. In this table, each investigation is an entity. There are six attributes that


The keys *(italicized)* define the unique attributes by which investigations are identified
Rows represent entities, e.g., the properties that define ***specific*** investigations
Columns define attributes of the entity

Investigation title	***Batch ID***	**Date initiated**	**Assigned to**	**Status**	**Completion date**
Low yield	0201	09-Jan-02	J. Smith	Closed	31-Jan-02
Metal fragments	0202	17-Jan-02	M. Jones	Closed	29-Jan-02
Low yield	0210	15-Feb-02	R. Williams	Pending	
Low potency	0210	16-Feb-02	R. Williams	Not started	

FIGURE 33.1 A Database Defining Status of Investigations into Manufacturing Discrepancies.

define the entity, but only two of these are needed to define the key by which unique investigations are identified. There may be duplicate values for attributes, but it is not possible for independent investigations to have duplicate values for all of the keys. As can be seen in this example, there are cases where the Title is duplicated and cases where the Batch ID is duplicated, but in no case can the Title *and* ID both be duplicated.

During a search, keys can be used to combine data from this table and others, as long as all of the keys are identified and values are supplied. For example, another table in a relational database might contain information concerning the product that was being made. This might have Batch ID as the sole key, so a search on investigation title and batch ID could provide all data shown in Figure 33.1 plus the identity of the product by finding the information in the two tables based on values provided for the keys.

Oracle® and Access® are two examples of relational database management systems. SAP®, which has thousands of tables, is an example of a large application that makes use of relational database technology. There are two other types of databases — hierarchical and network — but these are less common and generally limited to the mainframe world and will not be discussed here. This chapter will focus primarily on relational databases.

The kind of information managed, whether it is sales data, electronic documents, clinical trial data, or recipes for a manufacturing execution system, is fairly independent of the database type (although no one would build a flat file database for any of these). The choice of relational vs. hierarchical vs. network is primarily dependent on business needs.

APPLICATION DEVELOPMENT ISSUES

VALIDATION APPROACH

When we speak of validating a database such as Oracle®, we are talking about validating the database *application*, not the DBMS. The actual DBMS should be qualified as a Commercial Off-The-Shelf (COTS) software package. GAMP 4 recommends the following activities for this category of software:[3]

- Record version (and configuration of environment)
- Verify operation against user requirements

Validation of the database application will typically be based on custom/bespoke software. GAMP 4 recommends the following activities for this category of software:[3]

- Audit supplier
- Conduct full validation life cycle
- Record version (and configuration of environment)
- Verify operation against user requirements

USER REQUIREMENTS

As with any other system that must be validated, the starting point for a database is with a strong User Requirements Specification (URS). It is very important for the users to truly understand the data they want in the database, and the relationships between various data elements, including the keys they want to be able to use to relate the data. Users do not need to understand the underlying design or even the theory behind relational databases. That can be left to the designers, but users do need to be able to explain how their data is related.

It is crucial in this step to thoroughly understand and document the nature of the data, whether the database is being designed and built by an individual or by an internal IT organization, or whether a third-party package is being purchased to meet the need. While the latter case is unlikely to involve creating the relationships between various data elements, there will undoubtedly be some level of configuration involved, such as naming and defining fields.

Database design is very dependent upon being able to define hierarchies and relationships of data. For example, if you were designing a database to document GMP training, a logical way to define records would include assigning certain attributes to employees, such as name, employee number, department, and date of hire. None of these data mean particularly much if they are separated from the employee's identity (the name or, conceivably, the employee number); however, when you put them all together, you can draw certain conclusions about the training required.

The mode of data input must be specified. Will the system have to accept input from another system such as a laboratory instrument data system, or will it have to take direct input from sensors, e.g., thermocouples? Will it be interfaced to another database, such as an ERP system that must determine a batch status residing in a LIMS system before allowing a raw material to be allocated to a manufacturing step? Will there be direct entry of data via keyboard or barcode scanner?

Required automation features of the application must be specified in the URS. Does the database need logic or arithmetic functions to populate automated fields? For example, when controlling product release, a database may need to keep a status field as "Quarantine" until five other fields have acceptable values, in which case the status switches to "Released," *or* until any one field has an unacceptable value, which switches the status to "Rejected."

In cases where not all of the data required for a record will be available when the record is created, the users must know whether this should affect the search and reporting capabilities of the system. For example, in view of FDA Rule 21 CFR Part 11, hypothesize a scenario wherein only part of the record is entered initially. When data entry is finally completed this should show up as the initial committed entry of the record, or shown as a change to the whole record in the audit trail, or should the audit trail be granular enough to show that this is the initial entry in those particular fields?

The URS needs to record any special requirements for searching. For example, do the users want to be able to search on partial strings? Should searching on the name John retrieve John Smith, Edward John, and Mary Johnson, or only one of the possible subsets of this? Do they want to be able to use wildcard characters? Do they need to be able to search ranges between numerical values? Do they want to be able to refine earlier searches by applying new criteria to a prior search result?

If there are common reports that the system will be asked to generate, these should be specified in the URS. If reports need to be generated without user intervention, that needs to be noted as well.

For GxP databases data integrity issues will always be viewed as critical by regulatory authorities. It is important for those who will build the database based on this URS to fully understand the implications of regulations such as Part 11. Developing the design approach for audit trail will be dependent upon a number of factors upon which the users can shed light. This might include considerations such as how often data will be changed after the initial record is committed, how they envision audit trails should be available on-line and printed, or whether there is a requirement for electronic signatures. It is important that Quality Assurance/Regulatory Compliance be involved in the assessment of such issues, both to ensure that the database will be designed and built in a compliant fashion, and to ensure that this is done in a manner consistent with corporate compliance standards.

Functional Specifications

In the Functional Specification (FS) the database architect converts the information gleaned from the user requirements into a definition of what the application will actually do (but not how it will do it). A good URS is prioritized because users tend to ask for the world, while management is only willing to fund a rocky islet. The FS must be based on, and directly traceable to, the URS.

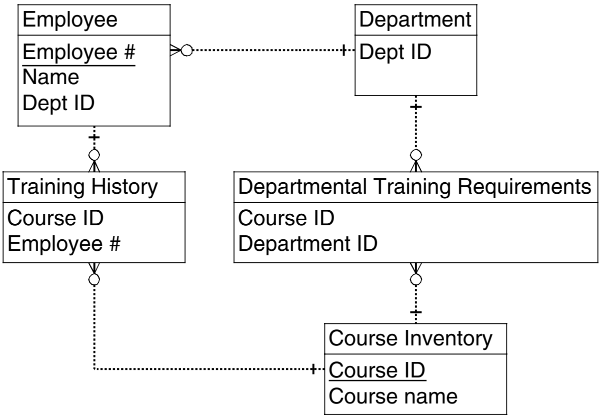


FIGURE 33.2 Entity Relationship Diagram for a Simple Training Database (Crow's Foot Technique).

Prioritization is removed at this stage because the FS represents what will be designed. This definition of the functionality defines the majority of the basis for Operational Qualification (OQ).

Design Specifications

Entity Relationship Diagrams

Relational database design is certainly unlike other forms of programming, but there are tools available to help with the task. Since the relationship between data elements is a key consideration, the Entity Relationship Diagram (ERD) is an important one of these tools. There are many conventions for documenting ERDs; the one depicted below is one of the simpler examples to understand.

The ERD is a basis for developing the database tables that define the relationships between data elements and, as such, it is an important factor in determining how to challenge a database during validation testing. Each of the entities is defined by one or more associated tables. Figure 33.2 shows an ERD for a training database. In this scheme, each department has a one-to-many relationship with training requirements and a one-to-many relationship with employees, i.e., there are many employees in a department. The employees, in turn, have a one-to-many relationship with courses taken (training history). Similarly, the training department's course inventory has a one-to-many relationship with both the departmental requirements (the training department offers many courses for each client department) and each employee's training history (each employee takes many courses). The database tables will be set up to make optimum use of these relationships. For example, the names of the courses an employee needs to take can be determined by taking the intersection of the department and the inventory, which gives the courses required by the department, less the courses already taken by the employee. As can be seen from the example, this database is extremely simple in that none of the tables will require more than three columns, yet the database will be able to track and report a reasonably complex (and important) regulatory activity.

When compiling validation documentation, it is also worth noting that visual aids are quite valuable tools during regulatory inspections. Confucius may have been thinking of this when he noted that one picture is worth a thousand words.

Field Definition

It is important that database fields be defined properly to fit the type of data expected to be entered in that field. A common point of failure for poorly designed databases is an inability to handle unexpected data. One of the aims of database validation is to demonstrate that this is not a problem. It is obviously best for the design to preclude such problems, thus avoiding heavy reliance on less reproducible factors like training and administrative processes.

An example of where field definition can improve the database integrity is simply date format, which is especially critical for data shared between organizations in the United States and other parts of the world. If a database query is entered for records between 1/4/02 and 7/4/02, should the database interpret this as January 4 to July 4 or April 1 to April 7? If the field is defined to display dates in a dd-mm-yy format, 04-Jul-02 will be unambiguously interpreted.

Similarly, it is much more difficult to validate databases that store important data in free text fields. Searches on free text often have to have more precise parameters than is reasonable to expect, and programmers cannot be expected to anticipate every free-text query. For example, a free text search for "blue and green" probably would not find "green and blue," or even "blue and green" unless the database developers were prescient. However if there are two fields for color, it is fairly easy to build logic that will search for both colors regardless of order. Defining fields properly can improve database performance too. Free text fields are generally slower to search.

Prudent requirements in planning and design will help to preclude user errors. If a field should accept only positive integers below 9, specifying and constructing the field in that manner may prevent a large portion of inadvertent data entry errors. Care should be taken when specifying boundary values, as this has often been a regulatory hot button. Validation efforts must include challenging this type of design feature because systems often fail at boundary values. An example of where this is critical might be a LIMS database where batch release is dependent upon an assay of 98 to 100%. It is critical that the acceptance criteria be set knowing whether 98 and 100 are values that should be passing or failing. It is possible that only values of 99 would allow batch release if the programmer used ">" and "<" instead of "≥" and "≤." Boundary testing is one important fashion in which validators can demonstrate that a system does not fail (as opposed to demonstrating that it works, which is much easier).

Finally, it is important that the design recognizes those fields without which the record is meaningless, and that these fields should be mandatory. This clearly includes the keys that are used to relate the tables, but that may not be the limit of critical data. In the training database above, if you do not accurately store the courses taken, our model would obviously be pretty useless, but that relates to data entry and not design; it is possible that the employee record will be set up before he has taken any training. However, the records would have no meaning without the identity of the employee (what is being tracked without this?) or without the department (there would be no record of what is required). These can be made mandatory when setting up the record. A record should not be allowed to be saved missing such critical information. Validation testing must verify that there are no meaningless records in the database.

Data Input Interfaces

The design must account for the origin of the data going into the database. There are few issues for systems whose only input source is keyboard, but as systems grow more electronically integrated this becomes less common. An important consideration when designing a database that is to be validated is ensuring that data input is not received directly from an unvalidated application. The GIGO philosophy (Garbage In — Garbage Out) applies in the world of validation as well.

Interface design should have some checks to ensure that data transmitted electronically meets expectations for completeness. For example, if one were using some electronic device to record attendance at a training session, e.g., an ID card reader, it is imperative that the incoming record of attendance be associated with a course ID. Validation must verify that such transfers work properly.

Audit Trails

In view of 21 CFR Part 11, audit trails have become a standard part of many database designs. When specifying any database that will include GxP data, it is important to recognize that specifications for Part 11 compliance must go beyond a statement that the application "must be Part 11 compliant." Not all designers or suppliers of commercially available solutions will have intimate familiarity with the regulation, and even those who think they do may have a different concept of compliance from the user company. Ergo users must be specific in their expectations as to what they deem to be a compliant solution. Key considerations regarding a compliant audit trail should include requirements to:

- Ensure that all data entries, modifications, or deletions are identified with the user, date, and time of the action. This must be based on a good understanding of the underlying predicate rules. For example, there may be a requirement for a motivation field in addition to the information noted above (GLP regulations require this reason for change[4]).
- Ensure that modifications or deletions do not obscure any of the previous values for the changed fields.
- Ensure that the audit trail is irrevocably tied to the record and will be retained for as long as the record.
- Ensure that reports can be generated for regulatory review, both electronically (to screen and file) and on paper.

Validation testing must demonstrate that the audit trail as implemented successfully meets the documented Part 11–based audit trail criteria.

Electronic Signatures

If a database is to employ electronic signatures, Part 11 is again a guide for design. Key considerations that should be recorded as requirements and challenged in validation testing are:

- In all displayed manifestations of the signature (both on-line and printed) the signature must display the name of the signer (not just a user-ID), the unambiguous date and time of the signing, and the reason for signature (e.g., approval, executed the task, etc.).
- The signature must be irrevocably linked to the signed record. It cannot be excised and applied elsewhere, and it must be invalidated if the record is changed subsequent to the signing.

Validation planning must ensure that these points are documented and challenged during testing.

Security

Databases comprise a class of application that often requires multiple levels of security. For example, it is possible that a business process may necessitate keeping the ability to modify existing records distinct from the ability to enter original records. Large database applications like ERP systems have many roles defined, and virtually no one should be able to enter, manipulate, or delete data across the whole system. Role-based security schemes are required, where appropriate, by Part 11. In any case, general users should never have the same level of access and edit/delete privileges that a database administrator would have. All access levels need to be challenged in validation testing. Role-based security may be built upon the ability to access certain tables and views in the database, and this may be a reasonably complex mechanism, so understanding the database design is quite valuable in developing a validation strategy for security.

Testing and Qualification

Test Planning

Validation test planning for a database application has the same two principal foundation blocks as for any other type of application: the traceability matrix and a risk assessment process. The traceability matrix is a tool that both ensures that the test plan challenges everything that needs to be challenged, and properly maintained, enables a firm to demonstrate to regulators that each current specification has a corresponding successfully executed test.

A judiciously applied risk assessment process is an important tool that can provide essential guidance at a number of key project junctures. It may be appropriate to use a variety of risk assessment techniques in one project. For a good example of one of these techniques see GAMP 4.[3]

The first risk assessment, and generally a very easy one to execute, is an assessment in conjunction with user requirements analysis that determines whether the database has any GxP bearing and thus requires validation.

If packaged solutions are being considered, another assessment should be conducted prior to selection of the supplier. Much of this assessment will be based on results from a supplier audit. In addition to a critical look at supplier quality systems, it pays to understand the database design process. Sometimes, in an effort to cut costs or meet tight timelines, a supplier may move from one DBMS to a newer one (e.g., DB2® to the most recent version of Oracle®) but not update the design. Such a practice can even go back more than one generation of the application. This may manifest in problems of incomplete compatibility and lead to such troubling problems as orphan data after deletion, etc. Especially in view of 21 CFR Part 11, this can lead to questionable data integrity. Such risks, and the potential associated increase in the complexity and the amount of work required for the validation, should be carefully considered. The rigor and extent of validation testing is one lever that can be applied to the problem of a poor supplier quality system. (The same issues would of course apply if similarly dubious practices were employed in an internal database design project.)

A third level of risk assessment enables validation planners to justify the extent of testing. This assessment should look at each of the principle functions of the database and assess them for likelihood of failure (based on a combination of design quality and complexity, user ability, and frequency of use); failure consequence (based on patient or worker safety and regulatory and/or business impact); and the likelihood of detection before serious consequences arise. Mitigation strategies, which often result in increased or decreased depth of testing, can then be developed. Other strategies that might be considered include (but are by no means limited to):

- Enhancing automated error checking processes for critical database entries, such as by requiring a data entry confirmation or building acceptance criteria logic into critical fields
- Requiring a confirmation of critical entries by a second operator
- Developing procedural checks

Any of the above strategies will have an impact on test planning, as they all are intended to reduce unacceptable risk to a tolerable level. Regulators have consistently demonstrated that they believe that there needs to be demonstrable evidence that everything that should have been tested has been, and that the depth of testing needs to be justified. The traceability matrix provides the former, and a sound risk assessment practice satisfies the latter.

These tools should not be considered limited in application to the initial validation effort. By keeping the traceability matrix up to date it becomes (and will remain) an important tool for assessing the impact of changes to the database, and both it and the risk assessment process are still important test planning tools as part of change control. Of course, sound change control practices are absolutely imperative for keeping an application validated.

Qualification

Database applications can be installed on stand-alone PCs, host computers, or may have client server architecture. The hardware platform supporting the application requires qualification. Hardware testing (Installation Qualification) should include:

- System diagnostics
- Power failure
- Communications failure
- Environmental controls (EMI, RFI, etc.)
- Inventory of resident software with versions
- Hardware layout diagram
- Physical access
- Logical access to the network (or operating system on a stand-alone implementation)

User acceptance testing (often equated with Operational Qualification) of the application is also required against the functionality of the application's specification.

- Reports
- Calculations
- Data entry processes
- Search processes
- Logical access to the application
- Role-based security challenges
- Backup and recovery proceses
- Archive and restore processes
- Interfaces to other systems

The role of Performance Qualification, which entails challenging the application within the scope of business processes, is harder to distinguish. As a consequence PQ for databases may be combined with User Acceptance testing. Items for consideration within a PQ include verification of data management within the application (actually checking manipulated data sets to determine they are correct), and examining the role of the application within the wider "process" flow.

Traceability

As with any other type of application, one of the keys to developing adequate testing will be thoroughly traceable specification documentation. Database design must be built from the functional specifications, and the functional specifications must be predicated on the user requirements. Beyond general adherence to this fundamental concept, significant attention must be paid to the details as to how each element of design relates to both the previous and the next level of specification. Figure 33.3 shows an example of how the design of the training database module discussed previously might be derived from a simple requirement to report on outstanding training needs. This single requirement leads to a set of four functional specifications, which in turn are elaborated into many more design elements. As noted earlier, understanding the design is important for developing test challenges, and a good traceability matrix is an aid in this. The matrix should provide transparent traceability from URS through FS and design specification to actual test cases.

Documenting traceability requires significant rigor in order for a traceability matrix to be useful. Further, the traceability matrix must be maintained throughout the life of the system to support change control. Using manual methods is barely manageable for a small application like this training database; for larger systems it becomes nearly impossible. Fortunately, a number of automated solutions to this problem are available.

It should also be noted that the ERD is an important part of the design specifications. It would be conceivable that the design specification referencing URS-1 and FS-1 through FS-4 would be the ERD and some field definition information such as defining the employee number as a positive six-digit integer. The important consideration is that a programmer be able to unambiguously interpret whatever design specification is provided to build the right database.

Specification ID	Specification
URS-1	System must be able to document training required by each employee
FS-1	System will track all departmental training requirements
FS-2	System will track employee department
FS-3	System will track courses taken
FS-4	System will be able to report difference between courses taken and courses required
DS-1	Employee table defines name and employee number
DS-2	Department table defines Department ID
DS-3	History table includes employee number and Course ID
DS-4	Requirements table lists Department ID and Course ID
DS-5	Training inventory lists Course ID and Course name
DS-6	Department to employee one-to-many relationship
DS-7	… and so on

FIGURE 33.3 Traceability Example: All of the Functional Specifications and Design Specifications in This Table are Derived from the User Requirement Labeled URS-1.

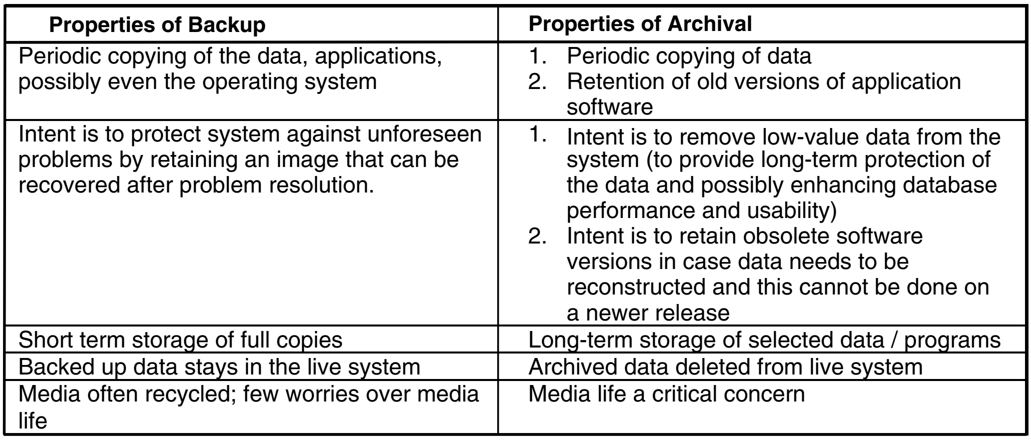
The key to developing and maintaining traceability is making certain that there is a process to ensure that changes to any of the referenced documentation is evaluated for potential propagation to other levels of specification. Developing and adhering to this process is an important validation activity and a regulatory expectation. The activity of planning validation testing should be built around such a matrix.

OPERATION AND MAINTENANCE

BACKUP/RECOVERY

Given the orientation of Part 11 toward data integrity, the FDA has stressed in many recent Warning Letters that it considers backup to be an integral part of protecting data integrity. For databases, the principal concern is protecting them against corruption, which can result from a variety of causes. When validating databases, two points need to be kept in mind:

1. The backup process needs to be defined, and most importantly it needs to fit the business scenario. For example, if a database is used very infrequently — let us propose, hypothetically, only in the month of January to close out year-end activities — it is inappropriate to use a process of monthly backups. This is because data centers typically retain only about four backup copies, overwriting the oldest copy when it is time to do the fifth backup. In such a scenario, if this database became corrupted in March, the last good backup copy would be overwritten in July but the corruption would not be discovered until the following January, at which point the data would be irretrievably lost. A much better strategy is to do annual backups in February, keeping them for several years. It saves the IT department work and provides a much more reliable data protection scheme. Similarly, if a database were extremely heavily used daily, monthly backup would be inadvisable because failure in the third week would force the restoration of a copy missing a large amount of data. There are backup strategies that can adapt to such scenarios while minimizing burdens on the IT group such as daily incremental backup (backing up only what has changed from the previous day) with weekly or monthly full backups.
   Even scheduling needs to fit business processes. For example, if the business process calls for large batch jobs to be run overnight, and IT is counting on nights to run backups, this issue needs to be addressed.
2. Backup and restore must be tested, and this testing, too, needs to be within the scope of the business process. Quite frequently validation teams simply note that their application and data are on machines that are supported by the data center, and that these are already



Properties of Backup	Properties of Archival
Periodic copying of the data, applications, possibly even the operating system	1. Periodic copying of data 2. Retention of old versions of application software
Intent is to protect system against unforeseen problems by retaining an image that can be recovered after problem resolution.	1. Intent is to remove low-value data from the system (to provide long-term protection of the data and possibly enhancing database performance and usability) 2. Intent is to retain obsolete software versions in case data needs to be reconstructed and this cannot be done on a newer release
Short term storage of full copies	Long-term storage of selected data / programs
Backed up data stays in the live system	Archived data deleted from live system
Media often recycled; few worries over media life	Media life a critical concern

FIGURE 33.4 Differences between Archival and Backup.

> within the scope of existing backup processes. By not testing the restore process they are exposing themselves to potentially unpleasant surprises when a recovery of backups becomes necessary after the system is placed in production and the process does not work as expected.

It is important to remember that by their very nature databases are constantly evolving. This gives a slightly different flavor to decisions regarding backup strategies than would be the case with a system like a chromatography data manager in which the data is, for the most part, static.

Archive and Restoration

While they are often lumped together in a discussion of system management issues, it is important to understand the difference between backup and archival processes in order to understand how each relates to validation and maintaining the validated state. While some companies do it, it is the wrong concept to retain backup copies for the length of the archive period. It is also terribly inefficient since the backup tapes will have the application and operating system in addition to any data that needs to be archived. Figure 33.4 shows a comparison of the properties of backup vs. archival.

As discussed in Chapter 15, 21 CFR Part 11 has made archival problematic, especially in the world of preclinical and clinical data management where retention requirements may amount to decades. This means that it is virtually impossible to avoid archiving data, if only because it is impractical to keep obsolete hardware and applications running *ad infinitum.*

In this light it is imperative that the archival strategy and validation effort for a database consider metadata. If metadata is incompletely copied, records restored from archive will not be properly retrievable and/or reported. Ergo, it is imperative that the database functionality be challenged again with restored records after it has been found acceptable with “normal” data.

Change Control

The principal issues for keeping a database validated are in essence the same as for any other type of applications. Change control procedures need to ensure that all changes are assessed for impact on the database (and interfacing systems). Decisions regarding the extent of testing should be based on a risk assessment.

INFRASTRUCTURE AND LAYERED SOFTWARE

To be considered validated, a database (or any other application) must be running on a qualified infrastructure. This includes servers, and as applicable, network(s) and workstations. For an infrastructure to be considered qualified the support organization must have current, approved documentation describing its configuration, and test evidence demonstrating that it has been appropriately challenged. There must be a reasonable, documented, and approved mechanism for handling change and problem resolution. Compliance requirements for IT infrastructure are discussed in a later case study.

SECURITY

In addition to the role-based security described earlier, it is necessary that operating system–level security be enforced. It does little good to have sophisticated application-level security if a user can access a controlled directory through the operating system and employ standard tools to modify or delete data. This means that this level of security should be included in planning for the security-oriented validation testing.

Especially in electronic signature databases, passwords must be controlled, enforcing periodic renewal. This process must not be too frequent, however, or else users may try to simplify the process by using inappropriately simple passwords, or worse, writing them down. User-IDs should also be controlled in that IDs should not be recycled after a user leaves the company. This helps to ensure that all ID/password combinations are unique. If password aging is managed by the application, this needs to be verified during testing. If it is handled administratively, one of the activities the validation team needs to plan for is verifying that the procedures have been developed and properly implemented.

DECOMMISSIONING

Decommissioning databases usually entails some decisions regarding the fate of the data within the database. Regulatory, legal, or business concerns often require retaining the data past the time when the cost-benefit ratio justifies keeping a database active. This may mean expending considerable effort migrating the data to another database or to a format that can be handled by a generalized archiving tool.

Once the decision has been made regarding what to do with the data, a formal decommisioning notfication should be prepared and signed by IT, the System Owner, and by Quality Assurance.

Two potential scenarios merit further discussion:

1. If the data is, indeed, no longer needed, the decommissioning letter should note this. Once the documentation is complete, IT should delete all instances of the application, all copies of the data, (including archives), and all supporting documentation (including validation documents). Users should always destroy relevant documentation. Firms must remember the costs and implications of legal discovery processes if they are tempted to retain data that should really be destroyed. The decommissioning letter should be retained in accordance with appropriate legal and regulatory expectations.
2. If the data is not being destroyed because there are business regulatory or legal reasons to retain it, where data will reside and any special tools or procedures required to access it should be noted in the decommissioning letter. The letter should be retained with the validation docmentation until such time as the data can be destroyed in accordance with the guidelines in the preceding paragraph.

MIGRATING DATABASES

Especially in the light of the 21 CFR Part 11 requirement to retain data in electronic form for as long as the predicate rule requires the data be retained, migrating databases becomes a major hurdle for companies using electronic record systems. Given the rapidity with which technology has become obsolete in the past 15 years, it is naïve to assume that database systems that are state of the art today will exhibit any greater longevity than their older siblings. Under this assumption, there are two logical routes to retaining electronic data. The concept of retaining obsolescent hardware and software can be rejected out of hand because of the expense of retaining it. The only realistic alternative, often unattractive in its own right because of the complexities involved, is migrating old data to new database applications.

It is important to preserve or translate as much of the original form and format of the data as possible, and that includes metadata. A database can contain a tremendous amount of metadata, such as audit trail information, electronic signatures, relationships between database tables, definition of field characteristics, etc. It is necessary to consider this metadata as part of the data set; failure to do so could inhibit searchability and reporting after migration, make modification of old records problematic, or possibly even result in loss of the integrity of the records. Validation tasks associated with data migration must be geared to demonstrate that neither of these circumstances prevails.

Validation testing for migrated records should include testing where similar examples of migrated and freshly entered data are challenged in a similar fashion and the test results compared. It can be the case that metadata is lost or otherwise affected during the migration, in which case otherwise identical records may behave differently in such tests.

Finally, firms intending to migrate data must remember to include already-archived data in their migration plans. It is possible that complications may arise with this archived data. For example, data archived from earlier software releases of the database might be readable through the current version, but it could be that minor differences in metadata could render this older data unreadable after migration if these differences are not specifically addressed in the migration process.

CONCLUSION

The principles of validating databases are essentially the same as they are for any other computer system. Key issues are having good user requirements, developing traceability while generating the functional and design specifications, test planning based on risk assessment, documenting everything thoroughly, and maintaining that documentation to reflect the current state of the system after it goes live.

However, understanding how the database is designed and keeping it in mind throughout the project will have a significant impact on the effectiveness of testing and on the ease with which the system can be supported after implementation. Extending this understanding to how metadata affects the records will make maintenance and eventual retirement a smoother operation.

ACKNOWLEDGMENTS

I would like to thank the following people for their contributions to the preparation of this chapter: my wife Jeanne for proofreading and suggestions for improving readability; Philip “Flip” Rutledge for help with the finer points of entity relationship diagramming; and Guy Wingate for final review and comments.

REFERENCES

1. Warning Letters issued by the FDA since 1997 can be found on the FDA Web site at http://www.fda.gov/ foi/warning.htm. While there are many examples related to backup and most do not mention databases specifically, the following examples all serve to illustrate agency concern with the concept of backup as a critical part of guaranteeing data integrity:
   Apheresis Technologies, November 1999 — In relation to a spreadsheet used as a database: "There is no documentation covering Excel application software, or any procedures instituted covering the protection of electronic records or *an established backup system*."
   Glennwood LLC, May 1999 — Regarding a chromatography data system (which includes a database of chromatography results): "The software allows for overwriting of original data. There are no written procedures for the use of passwords, levels of access, *or data backup*."
   Hydro-Med Sciences, February 1999 — "There are *no procedures for backing-up data* files and no levels of security access established."
2. Barman, Dilip, "Dilip's Brief Introduction to Relational Databases," http://www.cs.unc.edu/Courses/wwwp-s98/members/barman/databaseLesson/.
3. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (www.ispe.org).
4. FDA regulation 21 CFR Part 58.130(e) states: "… Any change in automated data entries shall be made so as not to obscure the original entry, shall indicate the reason for change, shall be dated, and the responsible individual shall be identified."

34 Case Study 16: Electronic Document Management Systems (EDMS)

Robert Stephenson, Pfizer
Roger Dean, Pfizer

CONTENTS

The Shorter Oxford English Dictionary[1] defines the word "document" as "that which serves to show or prove something." In the pharmaceutical industry, the unavoidable need to "show or prove something" ensures that documentation is an essential component of critical business processes and activities. Within pharmaceutical manufacturing, examples of documentation required by GxP regulations include master production records, batch records, standard operating procedures, validation documentation, analytical test procedures and results, cleaning logs, calibration records, etc.

Other documentation not directly applicable to GxP requirements is also of vital importance; for example, regulations governing safety at work and environmental control also require the provision of appropriate documentation in order to be able to demonstrate effective working practices, the results of monitoring and risk assessment activities, etc.

Thus, documentation is a valuable asset to the company as it can contain information that, if lost, may not be recoverable and, even if it is possible to be recovered, is likely to cost a considerable sum to restore to a useable condition.

Traditionally, documentation has meant paper. GMP has, when incorrectly applied, quite rightly become a euphemism for "Great Mounds of Paper." Even when correctly applied, the result is, inevitably, a large amount of documentation (therefore, paper) to serve as evidence of properly conducted work commensurate with the requirements of good practice regulations. For GxP documentation, systems have been developed to manage paper and ensure that the right pieces get to the right place at the right time. These have been generally based on the multicopy approach with controlled copies being distributed to known locations and being withdrawn as required. This is expensive, time consuming, and error-prone; paper copies can be easily lost or damaged and strict controls are required to ensure that controlled copies are kept consistent with each centrally held master copy.

The problems associated with the management of paper can be overcome by implementing an electronic document management system (EDMS).

WHAT IS EDMS?

Electronic document management systems control and retain documents from creation to archiving and all stages in between. Thus, a word processing package used to prepare a document for use in its paper form would not be part of an EDMS. However, if the same word-processing package is integral to a system in which a document is created, reviewed, approved, viewed, superseded, and archived then it is part of an EDMS. It would be wrong to restrict the term *document* to the output of a word processing package as documents can contain a variety of formats including diagrams, pictures, and spreadsheets.

A quick search of the Internet will identify a number of EDMS solutions. Some may require a lot of customization to produce a package that will suit the requirements of your business. Others are particularly geared toward the pharmaceutical industry and have built in much more of the functionality required to meet GxP regulations. It is still almost certain that some customizations will be required, but as EDMS providers continue to improve their understanding of regulatory expectations (in particular, 21 CFR Part 11[2]) and provide appropriate solutions, the fit with industry requirements will continue to improve.

The needs of the organization will determine the type of EDMS to be implemented. Systems can be local or, more commonly, distributed throughout the company to maximize the sharing of information and, hence, benefits. This distribution can be site-wide or even intersite using local area networks (LANs) and/or wide area networks (WANs). The validation of such networks is covered in Chapter 38: Case Study 20. Many systems are client-server based and may use a Web browser to provide read-only access into the system. Increasingly, Web-based interactive packages are becoming available. Figure 34.1 provides an example of EDMS system architecture for a multiuser distributed system.

EDMSs can be configured in many different ways to support the way documents are managed. Figure 34.2 shows an example life cycle for a document such as a standard operating procedure.

THE REGULATORY ENVIRONMENT

The regulations governing the manufacture of pharmaceuticals demand that documentation on the manufacturing and associated processes be in place. Regulatory inspections use this documentation as the primary source of evidence of compliance. A summary of the GxP regulations involving

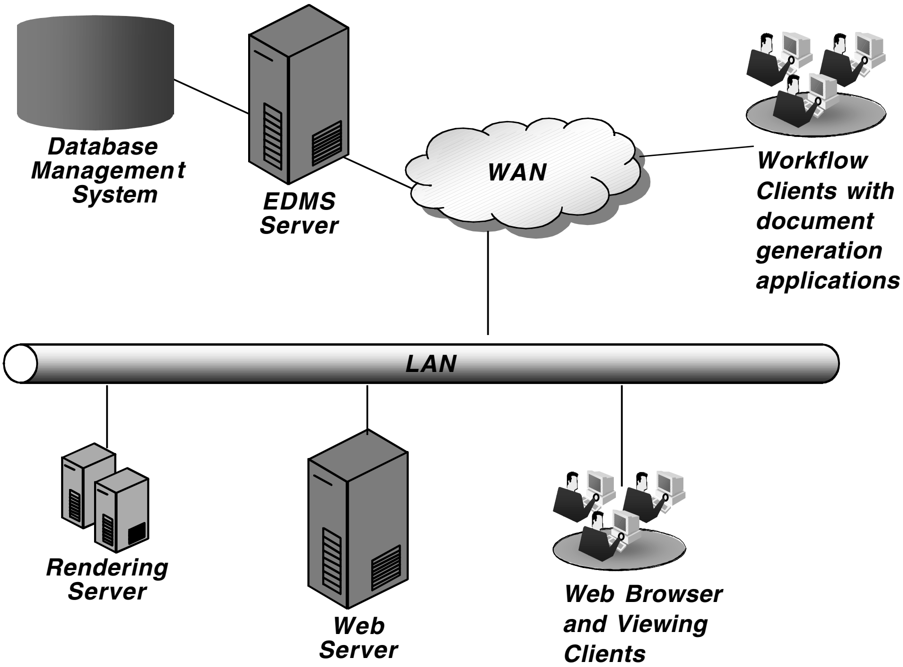

Database Management System
EDMS Server
WAN
Workflow Clients with document generation applications
LAN
Rendering Server
Web Server
Web Browser and Viewing Clients

FIGURE 34.1 Example of a System Architecture.

documentation and documentation management as well as some example citations in this area can be found in Reference 3. These clearly illustrate that the management of documentation is a key GxP function.

The regulatory environment applicable to electronic systems such as EDMS became clearer with the publication of the U.S. Food and Drug Administration (FDA) Electronic Records and Electronic Signatures Rule 21 CFR Part 11 in 1997.[2] Although only applicable to systems subject to inspection by the FDA, it gives guidance on what should be considered when developing and implementing any system containing electronic records and electronic signatures. The rule is certainly highly relevant to any existing or anticipated EDMS. Further information regarding the interpretation of 21 CFR Part 11 and its relevance to EDMS can be found in Reference 4.

Many of the requirements explicitly required by Part 11 are implicitly stated in European and other non-U.S. Regulations and Directives. Current thinking by international regulatory agencies is presented in the document "Good Practices for Computerised Systems in Regulated GxP Environments" (2003).[5]

IMPLEMENTATION AND VALIDATION OF AN EDMS

Implementing a site-wide or intersite EDMS is a major undertaking requiring a significant investment of time, resources, and money. The opportunity should be taken to review current document management practices to determine if they are still appropriate for the business and for use with an EDMS. An EDMS offers the capability to streamline document management processes and procedures, and some lateral thinking may help to drive a positive change into the organization. Consideration should be given to the part the EDMS will play in the overall computer integrated systems strategy of the organization. Failure to do so could require costly modifications as other elements of the strategy are implemented.

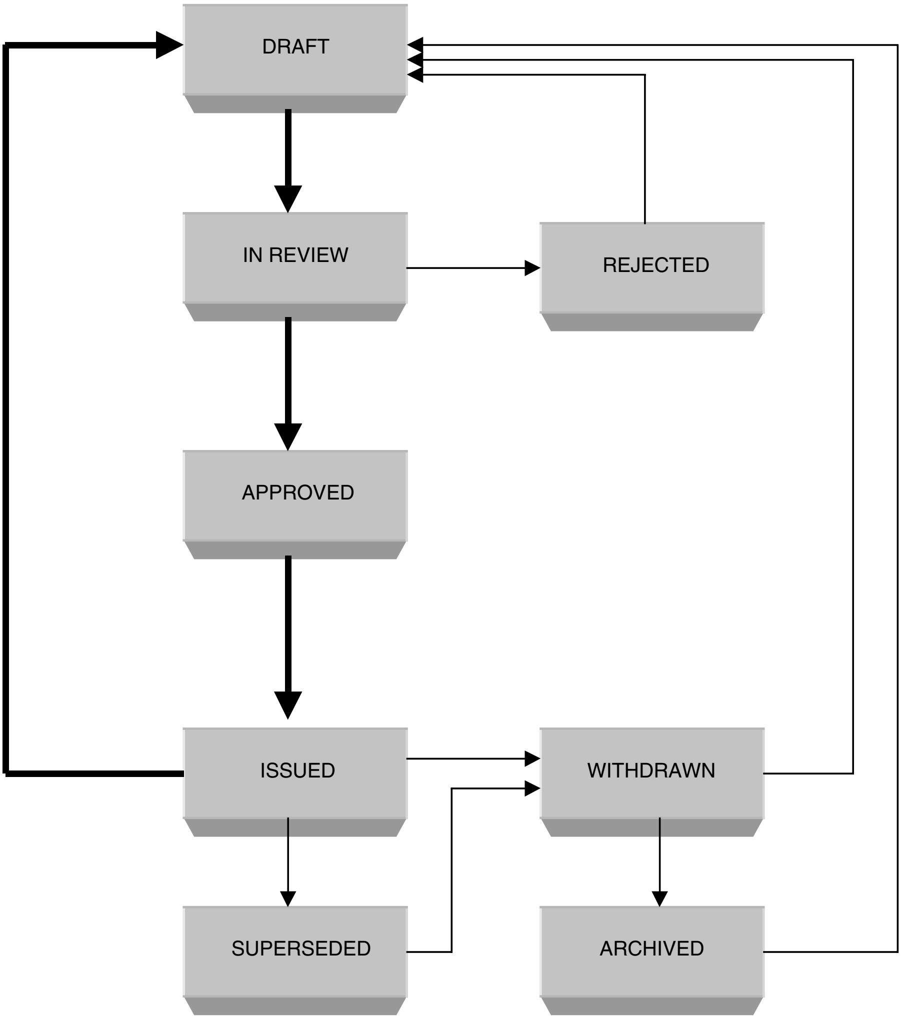


FIGURE 34.2 Example Life Cycle for a Controlled Document.

Life-Cycle Validation

One of the major factors to be considered when implementing a GxP-compliant EDMS is validation. The basic validation approach is no different from that applied to other information management systems such as MRP II or Laboratory Information Management Systems (LIMS). An approach based on the validation life cycle in GAMP[6] is appropriate. Figure 34.3 shows how validation documentation relates to typical project activities. Validation permeates all stages of the implementation process, as described below.

Project Team

The project team should consist of a core team of representatives from the key project areas. This team should include IT to provide expertise on the EDMS and IT infrastructure, user groups to ensure the system meets their needs, and QA and Validation to assure quality. The project team should meet regularly to review quality practices and set up regular communication sessions with the wider user base to keep them aware of all facets of the project so that critical decisions can be made in a timely manner.

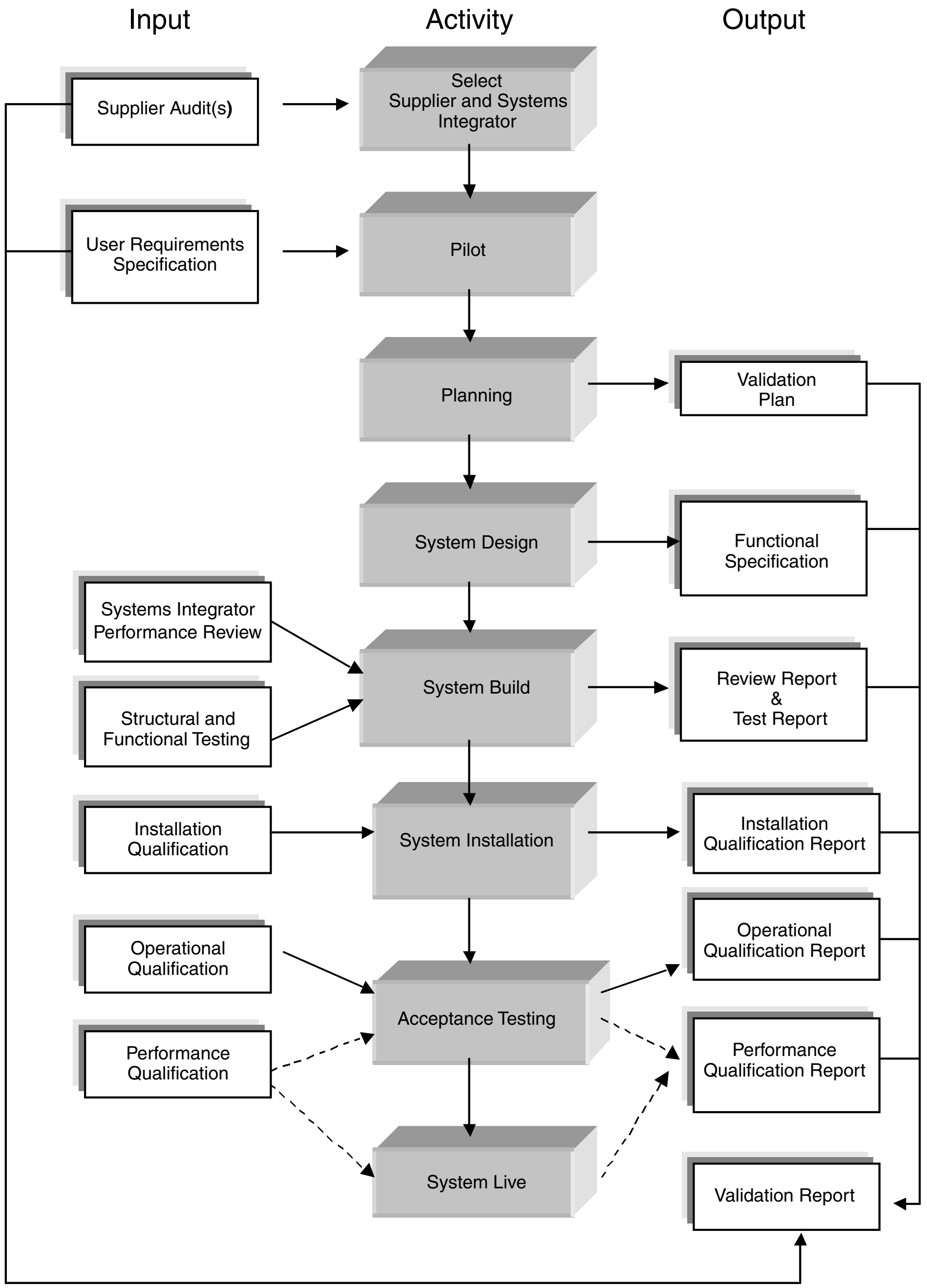

Input
Activity
Output
Supplier Audit(s)
Select Supplier and Systems Integrator
User Requirements Specification
Pilot
Planning
Validation Plan
System Design
Functional Specification
Systems Integrator Performance Review
Structural and Functional Testing
System Build
Review Report & Test Report
Installation Qualification
System Installation
Installation Qualification Report
Operational Qualification
Acceptance Testing
Operational Qualification Report
Performance Qualification
Performance Qualification Report
System Live
Validation Report

FIGURE 34.3 Project Phases with Validation Deliverables.

User Requirement Specification (URS)

The URS is a very important first step as it forms the basis for the definition of the system, supplier selection, and the approach to validation. It should be the mechanism by which the users have the opportunity to express their needs. Dividing the needs into musts and wants provides the project team with an indication of how to weight the requirements during supplier selection.

Examples of items covered in the URS are:

- Potential number of users
- Ease of use
- Ability to create documents in the company's standard packages, e.g., word processing, spreadsheet, and drawing applications
- Ability to update, withdraw, or archive documents
- Ability to print or prevent the printing of controlled documents
- Speed of access to the system and document retrieval times
- Good search facilities
- Presence or absence of hypertext linking
- Human Machine Interfaces (HMIs) required and their design
- Interfaces to other computer systems
- Audit trail requirements
- System availability requirements
- Impact of the existing IT infrastructure
- Access control and security requirements
- Use of electronic signatures
- Signature manifestation
- Signature/record linking
- Possible need to transfer legacy data
- Future use/growth of system

The URS should then form the basis of the evaluation criteria for suppliers of EDMS.

Selection of a Supplier and System Integrator

In large multisite organizations the system to be implemented may be governed by a corporate standard. This gives advantages both at the implementation stages and throughout the system's life in terms of knowledge of the product in the organization, availability of skilled implementation teams, and, of course, cost.

Where a new supplier is being selected a number of factors need to be considered and a detailed supplier selection process may be undertaken. Evaluation may consist of gathering information from various sources such as the suppliers themselves, companies who already have a system installed, and trial demonstrations. Where possible a fixed-duration trial (pilot) should be set up on site for the project team and user representatives to run through some scenarios of how the system may be used.

In order to validate the EDMS a supplier audit is essential in order to determine if the system has been developed and will continue to be managed within the framework of an adequate Quality Management System (QMS) and to good software engineering standards. Auditing the suppliers of electronic systems is becoming a specialized field in its own right; there are a number of references that give guidance in this area.[6,7] A positive Supplier Audit gives the user confidence that the system can be validated and that, once implemented, the supplier's activities will not adversely affect maintenance of the validated state.

The supplier selection process may result in the selection of a system that requires a lot of customization in order for it to meet stated requirements and GxP. If this is the case then a systems integration partner may be required for the implementation. The QMS of the systems integrator should also be audited to ensure that their methodology will result in a validated system.

Pilot Trial

The pilot is a short trial of the most probable supplier's system set-up, approximating requirements outlined in the URS and involving a cross representation of the user community. The pilot system is, by definition, temporary and, will not be built to the same standards as the actual EDMS. It is meant to convey a feeling for the system. The pilot is important because it will:

- Provide evidence that the system will meet the users' needs.
- Allow the team to form a better understanding of what the EDMS in question will provide.
- Assess possible configurations.
- Identify potential pitfalls of the technology.
- Expose the users to the screen interfaces, thus allowing any problems in this area to be highlighted at an early stage.

The concept of a pilot can also be used to compare possible suppliers. However, this decision should not be taken lightly as conducting a useful pilot requires considerable resources and time on the part of the team, plus the setting up of a pilot system by the supplier may involve a large cost.

The completion of the pilot will result in a decision to continue or abandon the EDMS project. If the decision is to continue, then it is essential that the URS is refined in light of the experience gained, both adding and deleting functionality and expanding detail as necessary. It must be remembered that these changes to the URS will affect the supplier's quotation, which must be resubmitted based on the revised URS.

Planning

Completion of the pilot is an appropriate time to prepare a validation plan for the project. There is a case for preparing a version of the validation plan at the start of the pilot but the effort may be abortive if the project is subsequently abandoned.

Completion of the validation plan will usually accompany the preparation of a detailed project plan. The quality representatives on the project team should ensure that key validation activities are included and that adequate resources are assigned. For systems that need to be customized and require systems integrators, planning is very important to get best value for money out of an expensive resource.

Any modern EDMS system will generate a comprehensive set of user requirements and equally detailed functional specifications. It is therefore crucial to build traceability controls into the project documentation from the very start. Tools such as documentation matrices and requirements traceability matrices should be used to keep track of the necessary interrelationships throughout the system life cycle.

Functional Specification (FS)

The initial part of the Functional Specification of the system should be to examine the document management processes and identify possible improvements rather than just mimic the current manual system. This can be achieved by the use of user discussion groups.

The next stage is to expand the basic requirements in the URS from what is required to how the users want that functionality to look and behave. The users must be involved in this stage, both to give them ownership of the product, and, more importantly, to obtain the benefit of their experience and knowledge of documentation management. This process, however, must also utilize experts on the EDMS system, such as the systems integrator, in order to facilitate user group discussions. The involvement of the systems integrator helps the team to achieve a realistic and practical set of requirements by identifying key deliverables from the system while advising against functionality that is unworkable or which will require significant amounts of customization. Discussion groups can also highlight inconsistencies between requirements of different user groups, especially when the system is designed to be used by all departments rather than by a small select group such as a dedicated Documentation Management function. [An example of this is print protection where one department may want free access to printing, whereas another group, possibly Quality Assurance, insists that printing of certain documents must be limited. Hence, a compromise must be reached which provides the required functionality in a manner that complies with GxP.] In some cases, however, GxP requirements will dictate the system design with no room to provide the requested functionality.

From the above discussions a detailed FS can be prepared which can be compared against GxP requirements and, if deemed to be acceptable, can be approved by members of the project team, including the quality representative. The FS must be cross-referenced with the URS to ensure that it encapsulates all of the users' requirements. The creation of the FS is critical in terms of validation as the Operational Qualification (OQ) test scripts will be written against this document, ensuring that all implemented functionality is tested.

Example elements of an FS include:

- Hardware configuration
- Software configuration
- Performance criteria and system availability
- Document database structure
- Document types supported
- Document workflows
- Viewing capability
- User Group configuration
- Access control, e.g., passwords
- Audit trail definition
- Search features
- On-line help
- Interfaces to other systems
- Training requirements
- Maintenance functionality

System Build

The EDMS should be configured from the FS. Wherever possible the project team should review the system as it is being configured to ensure that it meets the business requirements. Redesign and reconfiguration (performed under project change control) at this stage will significantly reduce user dissatisfaction, delays, and cost compared to similar activities after the system has been implemented.

To enable the system to be developed and user tested simultaneously it is useful to create separate instances of the EDMS, one for the systems integrators to configure (development system) and one for users to try out the functionality during its development (test system). Once the development of the system reaches an implementation stage, a validation EDMS is required. This may be the test EDMS put under strict control to prevent unauthorized changes, or more usefully, a separate validation EDMS to ensure that the validation is carried out under controlled conditions and is not affected by the users "testing" the system. Upon completion of the OQ the live EDMS can be created and implemented after conducting user acceptance "testing" to ensure that it behaves identically to the validation system. The provision of several EDMS systems requires a lot of space on the platform. As such, it may not be possible in all cases. However, it is essential to have an EDMS in addition to the live system in order to be able to correct and validate any faults found without endangering the integrity of the live database and to remove the need to take the live system out of use. The above process requires strict software/configuration version control to ensure that the various systems are using the appropriate version.

It is also important to audit the integrators during the configuration stage to ensure that they are complying with their own QMS. This gives the added assurance of good practices being adhered to and that the system is validatable.

Comprehensive unit and structural testing should be carried out by the systems integrator. If this is carried out properly, the number of faults found during OQ should be significantly reduced. This testing should be documented according to good practice guidelines such as GAMP.[4] The test scripts and evidence of testing must be handed over to the customer in a formal hand-over meeting on completion of the installation phase. QA must be present at this meeting and accept the documentation as satisfactory.

Installation Qualification (IQ)

Installation Qualification (IQ) is a phased process for these systems. Prior to OQ the hardware platform must be qualified against the specified design. This is no different from standard IQ and checks on the following may be included:

- The installed hardware
- The installed operating system software
- Environmental conditions
- Support procedures
- Maintenance agreements

The second phase is the installation of the configured software. This is also little different from a standard IQ for information management software and may include checks on the following:

- Installed core software
- Installed configuration
- Installed bespoke code (if appropriate)
- Software licenses
- Support procedures
- Maintenance agreements

This should put the system in a suitable state for OQ to commence.

At the completion of testing and when the system is ready to go live, a final IQ is required. This consists of promoting the tested EDMS from the validation to the production system (if differing EDMSs are used) and rolling it out to the users. Rolling the EDMS out usually requires some specific activities such as setting up desktop icons for connecting to the system, setting up user passwords, etc. A user IQ should be performed for each client by starting the system and accessing a known test document from each user station to ensure that the installation has been set up correctly. This should all be recorded.

If the development and validation have been carried out on a different server to that to be used for the live EDMS, then the installation of the software onto the live server must be qualified. If the servers are not identical then the full OQ should be repeated on the "live" server. However, usually the two servers are identical, and in this case it is only necessary to perform a subset of the OQ to ensure that the system works as intended. It is also very important if different servers are used to ensure that the configuration of the two servers is identical — the responsibility of the developers. They then need to maintain an accurate record via change control of any changes on the development server and ensure via the IQ that these are also migrated to the live server along with the application. This obviously also applies when applying future upgrades to the system once the system is live.

Operational Qualification (OQ)

All functionality as defined in the FS should be tested to demonstrate that the system is fit for purpose. It is important that the Operational Qualification (OQ) tests the functionality as a whole rather than just checking that the isolated modules behave correctly. This will involve taking a document through all workflows from start to finish and if the workflow is a continuous cycle then at least two cycles should be tested.

Example elements of OQ testing are:

- Administration
- Security of access to the system
- Security surrounding system functionality
- Creation of draft document
- Document review and commenting
- Document approval and release, including use of electronic signatures
- Document rejection
- Document made effective, i.e., in use
- Document superseded
- Document withdrawal
- Document revision — content
- Document revision — template or format
- Importation of legacy documents
- Viewing of documents
- Controlled printing of documents
- Production of "canned" reports
- System robustness
- Components of the system such as the system for creating a rendition, e.g., MS Word to PDF format, and the software for this process

It is equally important to identify and test for functionality what should not happen as well as checking that the system works as expected; e.g., in a workflow that involves parallel review followed by approval, the test should check that the document is not forwarded to the Approver until all the parallel reviews have been completed. Unless the users have been provided with a test EDMS, this will be the first time the full system is available to the project team and so there may be a tendency to try to make improvements to the system as it is tested. The project team must be clear whether such enhancements, or in the systems integrator's language "functionality creep," will be implemented with the associated cost and delays or whether only major concerns such as noncompliant GxP functionality will be corrected. If faults are found during OQ it is better to complete the full protocol, if possible, in order that all required changes are identified and resolved prior to running the OQ again.

When a document is prepared in an application such as MS Word and then rendered into another format, e.g., portable document format (PDF), there is a possibility that a particular character in a particular font or symbol set will not be able to be rendered by the rendition software. In this case the software will substitute its best guess. It is, therefore, necessary to validate each character of each font or symbol set used in order to ensure that it is accurately rendered by the system in use by creating a document in the font/symbol set in question, rendering it and comparing the characters on the two documents. This validation is required for all native application software fonts or symbol sets that will be rendered by the rendition server. It should be part of the OQ and made ongoing.

A summary report should be prepared following OQ testing, highlighting outstanding issues and their criticality to the project, and assigning responsibilities with a time scale for completion. The performance qualification phase cannot begin if there are unresolved critical issues from the OQ tests.

Performance Qualification (PQ)

There are differing schools of thought on whether Performance Qualification (PQ) is performed before the system goes live or afterward. In the former, PQ may consist of testing the system in a live environment with a restricted user base but using the system as envisaged when rolled out to all users. Alternatively, PQ may be used to assess the system after it goes live, checking system attributes that cannot easily be tested as part of the OQ. Testing here may include:

- System availability including ability to log on and access documents
- System access times and document retrieval times with the full user base, network traffic, and expected number of concurrent users
- Performance of the server
- Ability of the users to use the system
- Number of incidents and change requests
- Password management

A summary PQ report should be prepared, again identifying issues from the testing, and their criticality to the project, and assigning responsibilities with a time scale for completion. This type of qualification may be termed *ongoing assessment* or *performance monitoring*.

User Procedures

A validated system must have written procedures that have been formally reviewed, approved, and issued. These procedures should be reviewed (by someone from the intended user base who has an appropriate level of expertise in document management and who has been trained on the EDMS) prior to approval by the QA function. In addition, controlling procedures for the system administration function must also be established.

Database Population

Early in the project it must be decided whether existing documents will be imported onto the EDMS. If the decision is to bring documents into the system there are a number of ways of doing it. For example, either the electronic files can be imported into the EDMS or hard copies of the document can be scanned and the resulting file imported. Generally, it will be necessary to employ a mixture of methods particularly where old documents on obsolete word processing packages are involved or where not all of the electronic files are available. For GxP critical documents a validation program should be established to ensure that the version of the document in the EDMS is a true representation of the regulated document. For electronic files it is possible that the way they have been managed has not been to the same standard as that for the management of the paper system. Care must be taken to ensure that the correct document, i.e., the current approved and issued version, has been imported into the EDMS and that the file has not been corrupted or changed.

For scanned images the validation of the document in the EDMS should check that:

- It is the current approved and issued document
- All the pages of the document are present and in the right order and orientation
- There are no erroneous pages
- The image is legible

For imported electronic files the validation of the document in the EDMS should check that:

- It is the current approved and issued document
- That it has not been modified, e.g., a user has started to produce the next version using the file for the current approved version
- It is in a validated text (see section on "Operational Qualification")
- Any symbols or special characters have been correctly rendered (see section on Validation of Fonts and Symbol Sets)

Both of the above types should involve checking the document in the EDMS against the current approved document. This should incorporate a check on the accuracy of the attributes entered on importing the document.

The importance of this exercise cannot be overemphasized. If the system contains incorrect information, there could be GxP compliance issues and, from a practical perspective, users quickly become disillusioned with systems if they cannot rely on the information they contain. Hence, it is important that the information is also maintained during the implementation phase to ensure that any documentation updated in the hardcopy system is also updated in the EDMS.

In addition to validating each individual imported document a check should be made to ensure that all required documents have been imported. Failure to do this could result in critical documents being missing from the EDMS. As part of this final check the documents should also be checked to see that the EDMS contains the current version of all the documents in question in case documents have been updated since import. On completion of this validation step in a full life-cycle system the management of the documents in question should then be transferred to the EDMS.

Training

Acceptance and the continued use of a system are reliant on the perception of the user base about its usefulness. Training is key to helping users to have a positive impression and ensure that they know how to use the functionality that they require in their job functions.

The timing of user training will depend on whether all users will use the system immediately when it goes live or whether there will be a phased rollout of users. The former obviously demands that all user training is completed prior to implementation of the live system, whereas the latter means that each individual user must be trained before being allowed access to the live system. Training should use the procedures that will be available for the system. This not only checks the procedures to determine if they are correct and that they are easy to follow, it also familiarizes the users with them. If the user base is large, it may be useful to train a group of people who can provide on-the-job support to their colleagues.

Training is also required for the administrators of the system so that the EDMS can be maintained. The system vendors usually provide this.

Ongoing training is also needed to retain the validated state of the system, e.g., for new users and refresher training for current and lapsed users. All training must be documented.

Validation Report

At the completion of the implementation, a validation report is required to summarize all of the validation activities. It should summarize the outcome of each of the steps identified in the validation plan and review the progress of any outstanding actions from the IQ, OQ, and PQ reports. There may also be issues from the supplier audit or the review of compliance of the systems integrator against their own QMS that may need to be assessed. The report should then assign a validation status. It should also set a time for when a quality review of the system should be conducted. This is usually one year but if the project is being implemented in phases the report may defer assigning a review date until the completion of subsequent phases.

MAINTAINING THE VALIDATED STATE

Getting to a validated state requires significant expenditure of time and money. As well as being required for regulatory compliance, it makes good business sense to retain the system under control. A formal set of procedures and systems are required. These should include:

- **Change Control** — A system that manages change. Whether they be changes to hardware, version changes of the core software, or local configuration changes, it is critical to maintaining control of the system. The identification of categories where changes can impact the system can help to decide the degree of revalidation required.
- **Access Security** — Control of access to the hardware, software, and to the system via the user interface is very important. Access to system administration functionality should be controlled, particularly where a user performs significant events such as the creation or modification of user accounts.
- **Incident-Reporting Mechanism** — An easy-to-use system for users to report unexpected events with the system is an important monitoring tool.
- **Version Control** — In modern systems, the interface software, including customizations, is often on the client or the user's desk and, hence, is open to the possibility that the wrong version of the software is installed, e.g., not updated during an upgrade operation or interference by the user. Some automatic means (also subject to validation) should be found to check that the correct software is installed and preventing use if this is not the case.
- **Contingency Plans** in the event of system unavailability — This is particularly important if the EDMS manages the instructions on how to make product. Paper copies of the instructions may need to be held with some mechanism to prove that they are official copies and are true representations of those that are held on the system.
- **Disaster Recovery Plan** — Required in the event of a major failure to the server or other crucial elements of the system. A risk assessment should be performed to determine the criticality of the system to the business. The higher the degree of risk the more comprehensive the plans should be to quickly restore the system. This adds to the cost. Disk mirroring, platform mirroring, backup strategy, and identifying a business partner who will provide a similar platform in an agreed time frame, should all be considered, together with what to do in the event of the unavailability of the main platform and recovery actions due to failure of other crucial elements of the computer infrastructure, such as networks.
- **Backup Strategy and Media Storage** — How it is done, records to demonstrate that the procedure is being followed, how the backup can be restored and shelf life of the storage media all need to be considered. It is also essential to prove that the restore procedure works before it is required.
- **Maintenance Agreements** with the hardware supplier and the systems integrator should be considered.
- **Periodic Reviews** of the system, required at the frequency assigned in the validation report. They will include reviews of the change control and incident reporting

methodology, and an assessment of the cumulative effect of any changes. Training, procedures and records, and any outstanding actions from the validation report or previous reviews will also be reviewed.

SUMMARY

EDMSs are integral to the drive toward a paperless manufacturing environment. EDMSs also provide a useful tool to share information in a way that minimizes duplication and ensures that it is easily accessible when required.

In the pharmaceutical industry, validation is a prerequisite to use of EDMSs for GxP purposes. The validation methodology used is similar to that used for other information management systems. As with all systems, the more attention that is devoted to the design and validation, and ensuring that the users will be happy to use the system, the greater will be the benefit to the business.

ACKNOWLEDGMENTS

The authors would like to acknowledge the first edition of this case study by Richard Mitchell and Roger Dean of Pfizer Ltd., Sandwich, Kent.

REFERENCES

1. *The Shorter Oxford English Dictionary*, Volume 1, Third Edition, Oxford University Press, Oxford, U.K., p. 589.
2. FDA (1997), Electronic Records and Electronic Signatures, U.S. Code of Federal Regulations, Title 21, Part 11, Rockville, MD.
3. GAMP Forum (2001): *Good Practice and Compliance for Electronic Records and Signatures: Part 2 — Complying with 21 CFR Part 11*, *Electronic Records and Electronic Signatures*, published by ISPE and PDA (www.ispe.org).
4. Vesper, J.L. (1998), *Documentation Systems Clear and Simple*, Interpharm Press, Buffalo Grove, IL, pp. 10–23.
5. PIC/S (2003), Good Practices for Computerised Systems in Regulated "GxP" Environments (PI 011-01), Pharmaceutical Inspection Convention, Geneva, September.
6. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP4), published by International Society for Pharmaceutical Engineering (www.ispe.org).
7. Garston-Smith, H.T. (1997), *Software Quality Assurance, A Guide for Developers and Auditors*, Interpharm Press, Buffalo Grove, IL.

35 Case Study 17: MRP II Systems

Guy Wingate, GlaxoSmithKline

CONTENTS

MISSION IMPOSSIBLE?

"I want a system to replace all the stand-alone pockets of automation handling the warehouse, purchasing, and materials management systems. The system is to interface to my laboratory management systems and provide Internet links with my customers and suppliers … I expect to be able to reduce my inventory, reduce lead times, reduce IT costs, improve regulatory compliance, and install a system which is flexible to change."

The answer — besides "Don't we all want this?!" — is the implementation of an MRP II system. Well, MRP II does offer a solution, but it is no panacea. Indeed, there are examples where the cost of a poor implementation of an MRP II system has led to a pharmaceutical company's demise.

An additional requirement in the pharmaceutical industry is the need to fulfill the requirements of Good Practice (GxP) regulations that impact the use of computer systems. This requirement is often expressed as an afterthought to the quote at the beginning of this study — "Oh, and by the way, I want the project to fully comply with GxPs but at a minimum cost."

The GxPs (covering Good Clinical Practice, Good Distribution Practice, Good Laboratory Practice, and Good Manufacturing Practice) necessitate the ability to demonstrate that a drug product can be consistently made to its specified quality criteria. Failure to satisfy these regulations can result in a regulatory authority refusing to accept pharmaceutical products made using the computer system concerned. Lost sales revenue for a single top-selling drug could exceed 2 million Euros per day. An MRP II system usually coordinates operations across an entire site or sites.

Deficient application, operational error, or system malfunction could potentially affect the manufacture of all the products using the MRP II system. A single medium-sized secondary manufacturing site may have an associated annual drug sales revenue in excess of 1 billion Euros. Factoring up this rule of thumb over a number of drug manufacturing sites quickly demonstrates how "super critical" MRP II systems are.

It is vital that when a company commits to the implementation of an MRP II system, it does so knowing how critical project management with compliance is.

Project Approach

The implementation of GxP is often referred as validation and the well-known regulatory authorities such as the U.S. Food and Drug Administration (FDA) and the U.K. Medicines and Healthcare products Regulatory Agency (MHRA) have given guidance on what they expect to see during an inspection. A life-cycle approach should be adopted in the implementation project and care taken to ensure that after cut-over, the system is maintained for ongoing compliance. The GAMP Guide[1] provides general industry guidance but this must be adapted to the needs of an MRP II project whose activities will typically include

- Project Initiation
- Supplier Selection
- Install Development System
- Define Business Processes
- GxP Assessment
- Conduct Conference Room Pilot
- Review Legacy Data and Data Upload
- Readiness Review and Go Live
- Performance Improvement

Project initiation will scope the business processes and systems integration needs for the system within what is usually referred to as a User Requirements Specification (URS) or "To Be" document. The typical operability of an MRP II system is shown in Figure 35.1. Project initiation will also include project planning, budgeting, and project risk analysis. For the pharmaceutical industry it is also the point at which quality assurance practices begin. A Validation Master Plan (VMP) will need to be prepared to identify the project process, procedures to be adopted, personnel requirements, roles and responsibilities, documentation to be delivered, and milestones showing the rollout of the project to completion: in essence the principles of ISO 9000, but recognizing that validation for GxP goes beyond such quality management systems. The VMP may itself reference a number of Validation Plans covering specific aspects of the MRP II system. A project quality plan may also be produced.

It is likely that selection of the supplier has occurred by default rather than choice. For instance, SAP R/3 is the clear market leader for MRP II systems. Nevertheless, a supplier audit should take place and include the original vendor of the software product suite being used and any system-integrating companies taking responsibility for delivery of whole or part of the system. This audit needs to assess the confidence that a pharmaceutical manufacturer can place in the quality of the software and hardware products used in the MRP II implementation. The GxP regulatory authorities hold the pharmaceutical manufacturers directly accountable for such quality and where there are deficiencies or insufficient evidence of quality they expect the pharmaceutical manufacturers to remedy the situation. This may involve working directly with suppliers to improve their quality management systems, or working through a third-party consultancy.

Defining the business processes to be implemented by the MRP II system, either within the system or in conjunction with other interfaced systems, is a key task. SAP refers to this activity

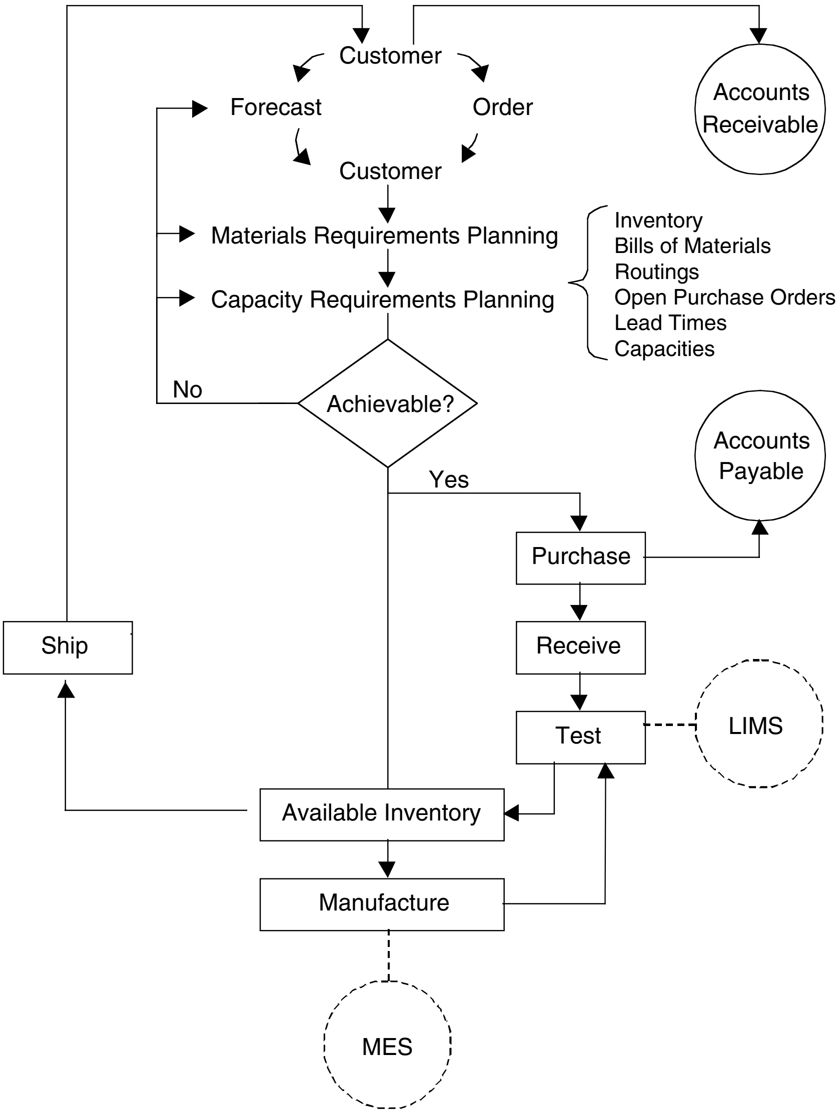


FIGURE 35.1 Overview of MRP II Functionality.[1]

as "blueprinting." This will involve reviewing the current ways of working and perhaps embarking on a program of change in these working practices — Business Process Reengineering (BPR). An overview diagram showing how top-level business processes fit together should be prepared along with diagrams illustrating the operability of the main functional elements (business processes) making up the MRP II system.

Once the business processes have been agreed, a GxP assessment can be conducted. This should address those operational aspects of the system that impact the quality of finished pharmaceutical products and will include supplier details, batch records, laboratory quality control records, batch release, and recall. An example of a GxP impacting functionality in an MRP II system is given in Appendix 35F. Experience suggests that perhaps between 25 to 50% of MRP II functionality* is GxP critical.[2,3] The GxP operational aspects will form a focal point during any GxP regulatory

* SAP R/3 Modules: CO (Costing), FI (Finance), MM (Materials Management), PP-PI (Production Planning — Process Industries), QM (Quality Management), and SD (Sales and Distribution).

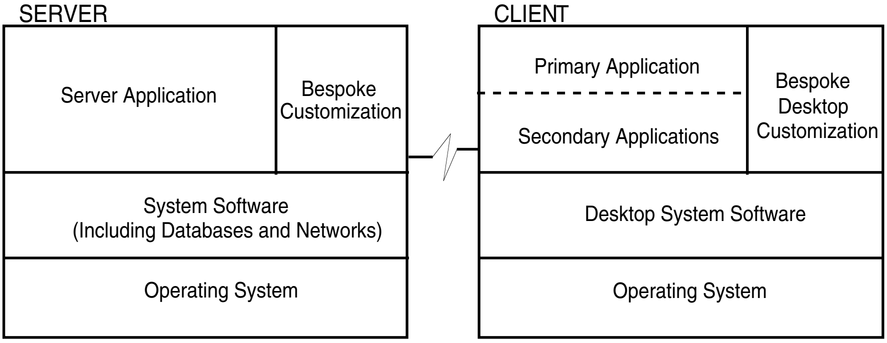


FIGURE 35.2 Client–Server Software Schematic.

inspection. It is very important to document why these aspects and not others are deemed GxP impacting, and where these operational aspects are defined and tested. Remember that inspectors will keenly challenge the distinction on non-GxP impacting. They form a robust assessment and err on the side of caution as to whether something is, rather than is not, GxP impacting. This determination will bring focus to the validation exercise.

Almost immediately a system will need to be installed to provide development and testing environment. A separate system is usually installed later to provide a go-live production environment for cut-over.

Set-up of the various system environments must be managed. Documentation must be developed to describe the hardware platform and installed software, including any network infrastructure. Hardware architecture design documentation should be prepared. A diagram should be included to illustrate the geographic distribution of any client–server hardware. Client–server software also needs to be defined. Clients are often referred to as either "thick" or "thin," depending on whether they require substantial or minimal application-related software. Client–server software can be considered to consist of:

Operating System: Operating system independent of the client or server application. GAMP level 1 software requiring version to be recorded (e.g., UNIX OS).

System Software: Standard software specific to intended use of client (e.g., desktop utilities) and server (e.g., network and database utilities). GAMP level 3 software requiring the version to be recorded and operability confirmed (e.g., Oracle Database and Microsoft SMS) unless software is held on firmware in which case it is GAMP level 2, requiring the configuration and version to be recorded.

Server Application: Application software products such as the MRP II software product. GAMP level 4 software requiring a supplier audit, validation of the configuration, and confirming the operability of the standard element of the software (e.g., SAP R/3). There may also be some standard software such as GAMP level 3 requiring the version to be recorded and operability confirmed (e.g., third-party utilities provided with the server application).

Client Applications: A client may be used for more than one application (e.g., MRP II, LIMS, and EMS). Each application will have an associated file set providing what is often referred to as its Graphical User Interface (GUI). File sets are usually built into standard client set-ups. Individual files may include some element of configuration. GAMP level 3 (e.g., Windows NT) and GAMP level 4 software require the version to be recorded, operability confirmed, and any configuration validated. Supplier Audit requirements are usually satisfied as part of the server application validation.

Bespoke Customization: Bespoke programming (e.g., macros defining reports and forms, and interfaces especially written for the client or server). GAMP level 5 software requiring a supplier audit and validation of the bespoke code (e.g., SAP R/3 ABAP form and report programs, and Microsoft SMS client scripts).

Robust server architectures are required to provide a dependable service to what may number hundreds or even several thousand clients. Basic configuration management of the server and network are expected. It is also important to define client builds (sometimes referred to as the desktop) and maintain them under configuration management. Client builds should have their applications integration tested to check there are no conflicts. It is quite common for clients to run multiple applications, and it cannot be assumed that conflicts will not occur, even between standard application products. Automatic desktop configuration tools should be validated in their own right.

An Installation Qualification (IQ) is needed to define and execute tests to verify successful installation of the hardware platform and resident software. As the project ramps up, the system is likely to require expansion to cope with a larger user base in which case the IQ must be revised. The IQ usually includes:

- Inventory and configuration checks for the hardware platform (clients, server, and network)
- Inventory check of software used
- A check of all vendor-supplied manuals to be sure they are present and correct
- A check to make sure necessary SOPs are available
- Environmental checks made in computer room housing hardware platform on power supplies, backup power supplies, temperature, and humidity
- A check of physical security mechanisms
- A check of system boot-up diagnostics

Following on from the URS, a system definition consisting of Functional Design Specification (FDS) needs to be collated. The URS does not necessarily specify the chosen MRP II system, and if this is the case, the FDS will need to introduce and overview the selected MRP II system. The FDS will define the URS business processes at a transaction level. Referenced documentation published by the supplier defining the standard MRP II software product and its functionality should be retained and maintained with the current version of the MRP II software used. It is important to identify those functions of the standard MRP II system that are used and specifically document which functions are not being used.

Process flow diagrams should be considered as the basis for SOPs developed for the transactions implementing the business processes as they are generally easy to understand and can be designed to highlight user interaction and interfaces to other systems linked to the functionality provided by the MRP II system. SOPs, forms, and reports must be drafted and under version control ready for piloting in what is sometimes referred to as a Conference Room Pilot. Appendices 35A through 35E present some typical business processes for procurement, production planning, production, sales and distribution, and finance with associated example SOPs.

Once defined, the business process transactions can be configured within the development environment of the MRP II system. There are normally instances when it is easier to amend the business process to fit the standard functionality of the MRP II product software than to make a customized bespoke modification. Any bespoke modifications, like the interfaces, must be fully documented in design specifications, test specifications, and test records. One important aspect to avoid during configuration is to set up the system to accept default user entries. There have been several recalls within the pharmaceutical industry because users failed to recognize that a default entry on their MRP II systems was incorrect. It is always a good idea to have positive user confirmation of key data entry or decision points. If defaults are still required then make them

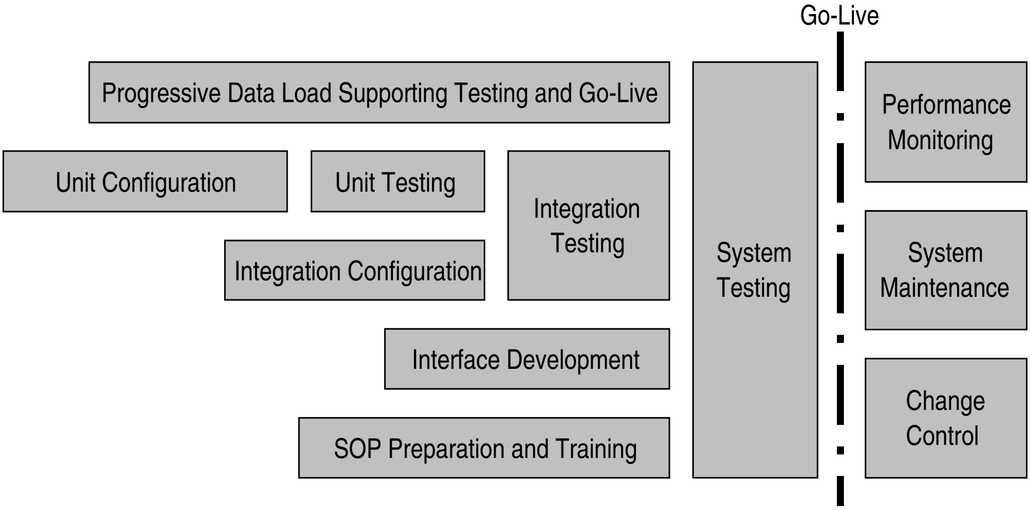


FIGURE 35.3 Schematic MRP II Testing Plan.

TABLE 35.1
GxP Data Elements in MRP II Systems (Based on Reference 4)

Batch Information	Assets	Bill of Materials
Batch Number	Purchase Order Number	Items
Batch Status	Contract of Supply	Quantity Per
Dates of Manufacture		Units of Measure
Expiry Dates	**User**	Conversion Factors
Quantity/Potency	Name (and Password)	Work Centers Conversion
Approval Restrictions	Security Access	Yield Factors
		Approval
Item	**Customer Orders**	
Item Number	Shop Order Number	**Shop Order**
Item Classification	Customer Order Number	Quantities
Location	Customer Addresses	Receipt Date
Type		Transactions
Quality/Potency	**Supplier**	
Shelf Life and Retest Interval	Quality Approval	

fail-safe, i.e., default entry on product sample status should be "reject" and require positive selection of alternatives such as "retest," "rework," "pass." Figure 35.3 indicates how configuration can be split into unit and integration activities. The pace of MRP II projects usually brings pressure to begin testing as soon as possible. This can be facilitated by testing unit configurations and then, as a follow-on activity, their integration. Perhaps as much as 80% of the configuration activity can be attributed to unit configuration.

The completed FDS should be verified that it is consistent within itself and with SOPs implementing the business process transactions, and that it fulfills the requirements of the URS. The activity is often referred to as a Design Review (DR) or Design Qualification (DQ). The use of a requirements traceability matrix (RTM) should be considered to demonstrate how URS elements are addressed in the functional specification and design documentation. This RTM can later be extended to trace test specifications and results.

When the business processes have been implemented, they are transferred to the testing environment. A key prerequisite to testing is data load. Ensuring the integrity of data is a must — garbage in, garbage out! It is important to review legacy data and new data entry requirements in readiness for testing and cut-over to the production environment of the MRP II system. Not all data from the replaced legacy systems needs to be transferred to a new MRP II system. Decommissioning and archiving of legacy data must be carefully considered. Some pharmaceutical companies have tried to distinguish between critical and noncritical data and set different data accuracy requirements for each. In reality, there is little difference when it comes to user and customer satisfaction. All data should be checked for accuracy and if its integrity does not pass, cut-over must not occur. Transport mechanisms for data load must be validated to provide assurance of data integrity. The distribution of data (once loaded into the system) and its control must also be defined.

Testing in a Conference Room Pilot, referred to at this stage as Operational Qualification (OQ), can largely be limited to "black box" functional testing where a standard system is used without modification as long as the supplier audit determines a high confidence in the embedded quality of the system. If a system is customized or a supplier audit notes significant issues with the supplier's quality development of the system, then "white box" structural testing should also be conducted. Either testing should include challenge tests to verify the system can detect within reason operator error and bad data. As with the IQ there should be a preapproved protocol before testing begins, and test records must be collated. At this stage in the OQ, the system is still under refinement and any changes must be logged and necessary retesting carried out. The size of MRP II systems means that Conference Room Pilots are often organized to exercise certain areas of functionality, based around the pharmaceutical company's organization or the MRP II systems standard functionality. It is important not to forget to rigorously test the integration and interfaces interconnecting these areas. OQ tests will include, but are not limited to:

- SOPs implements transactions (see Appendix 35A through 35E in this chapter)
- Verifying the processing of batch and laboratory records
- Challenging user interactions
- Testing accurate data manipulation and presentation (e.g., rounding errors and number of display digits)
- Verifying backup and recovery procedures
- Checking product recall processes

The OQ should also include system performance tests to confirm the system can cope with high numbers of active users and large volumes of data.

Education and training programs should be established for the project team and the end users. Education is based on presenting principles while training is based on practical hands-on tutorials. Course modules must be documented and their content approved and delivered by authorized trainers. Staff training records, including those for contractors, should be maintained to track attendance on courses. The use of competency questionnaires to verify learning should be considered. Mere attendance does not necessarily imply that an individual has understood and taken on board course material.

There may be a significant training requirement associated with the new MRP II system. The Conference Room Pilots provide an opportunity to train users in new SOPs and hence reduce the need for separate training events. Organizational structures and ways of working often alter with the implementation of an MRP II system, and with large scale training requirements many pharmaceutical companies employ change management consultants as well as MRP II specialists. Pharmaceutical manufacturers should anticipate training requirements as demonstrating the competency of staff — a key aspect of the GxP regulations.

Successful Operational Qualification means that cut-over of the system into live operation can be considered. Other issues affecting cut-over are whether all procedures and software have been

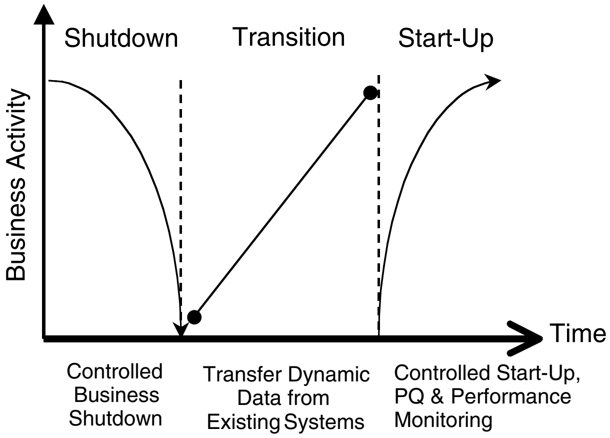

Shutdown
Transition
Start-Up
Business Activity
Time
Controlled Business Shutdown
Transfer Dynamic Data from Existing Systems
Controlled Start-Up, PQ & Performance Monitoring

FIGURE 35.4 Business Cut-Over Period.

frozen and issued, whether all tests have been completed, whether all tests have passed, completion of project documentation, and relevant business managers are in themselves confident of a successful cut-over. It must be stressed that in the pharmaceutical industry cut-over must not be allowed if the project's GxP-related validation documentation is not complete. A formal "Go/No-Go" decision should be taken to document the cut-over decision with signed approvals. It is advisable to incorporate this within an Interim Validation Report authorizing cut-over of the system from a regulatory compliance standpoint.

The cut-over process can be considered as comprising three main stages (see Figure 35.4). First, the business operations must usually be shut down in readiness for the decommissioning of existing systems and the switch to active use of their replacement system(s). This can be a complex management exercise if many existing systems are being decommissioned. The next cut-over phase involves dynamic data upload which cannot, by the nature of the data, occur earlier. Static data will have been loaded earlier, usually in the OQ phase of the project. Finally, a controlled start-up of operations can begin. Back-out plans and procedures should be put in place in case a major problem occurs during the cut-over period.

For many implementations cut-over is the point of no return! It is vital that the MRP II system is ready for cut-over before the cut-over is authorized. Validation case studies on ERP, MRP II, EDMS, LIMS, and Warehouse Business Systems have stressed the importance of cut-over management.[4–7] There is often considerable pressure to cut-over on time and a reluctance by individuals to be the first to say that their aspect of the system implementation is not complete and ready for cut-over. As far as possible an open and honest culture should be established. It is better to delay a cut-over and take corrective actions than cut-over on time and live through operational difficulties directly attributable to not being ready for cut-over. Hindsight is a wonderful thing, but not when you are unable to release drug products to customers. Try to use terms such as *breakpoint* to describe the time at which a decision to go forward or not is taken. Breakpoint implies work will stop if criteria are not met. Referring to milestones does not have the same impact, and avoid using very emotive terms such as "drop-dead" date. Who drops dead? The organization for going live when the system was not ready or the messenger who brought this to the attention of senior management? In any event, it is wise to develop contingency plans (sometimes also called *business continuity plans*) and challenge their feasibility before cut-over just in case.

After cut-over, the performance of the MRP II system should be monitored and evaluated. This is sometimes referred to as Performance Qualification (PQ). The PQ protocol should be prepared identifying key performance metrics such as

- Successful batch release in live environment
- Number of new change requests
- Number of outstanding change requests
- Number of help desk calls
- Changes to business processes
- Data accuracy
- User enquiries and retraining requirements
- System outage (partial or total)
- Security profile changes

Following a period of, say, 3 months from cut-over, it should be possible to demonstrate that the MRP II system is enjoying a period of stable operation. A PQ report should collate data, possibly graphical, that can demonstrate these trends (see Figure 35.5).

To conclude, the implementation project Validation Report is prepared in response to the Validation Plan issued at the beginning of the project. It summarizes what went according to plan, and explains what did not go to plan. Amendments to the plan must be justified. Some issues may still be outstanding, in which case forward audit trails to corrective actions must be made. The Validation Report must demonstrate that the MRP II implementation is fit for purpose and can be used to support drug manufacturing. Due to the large number of documents often associated with these projects a library index or route map may also be useful to include either in the Validation Report or, more commonly, to reference in the report as a separate document.

ROLL-OUT STRATEGIES

Within large corporate roll-outs of MRP II systems there is likely to be a core system configuration providing a company a standard way of working. Individual sites will implement the preconfigured system with minimum variations to the standard core system. In this way the site implementations can share the standard core system documentation. Each site will normally have its own Validation Plan directing (and hence Validation Report responding to) site-specific validation activities and placing these activities in the context of the standard core system documentation so that when regulators come to inspect the system, they will understand it from a site perspective. To date, the vast majority of regulatory inspections of MRP II systems come in from a site inspection.

Corporate roll-outs are often phased: typically finance, inventory, and warehouse in the first cut-over, then customer services (sales and operations planning in the second cut-over, and distribution), followed by production as a third cut-over. However, some roll-outs consist of a single cut-over and as such are sometime referred to as “big-bang” events. There is a high risk associated with big bang cut-overs because of their complexity and the total dependence on the new system. Instituting interim procedures to bridge phased roll-outs, however, also has risks, and a balance must be struck to manage the issues posed between big-bang and phased roll-out approaches. It is worth noting that it is often not practical to continue running original systems being replaced by the new MRP II system for any length of time after a big-bang cut-over. This position, however, as the GxP regulations stand, might be considered noncompliant.

MAINTAINING OPERATIONAL COMPLIANCE

Controls must be put in place to ensure the MRP II system is maintained in a validated state. These controls will include change control (hardware, software, data, and documentation), configuration management, desktop management, maintained list of users, security and access management, service level agreements for maintenance and repair, ongoing education and training, and periodic reviews. Business processes and SOPs must also be maintained.

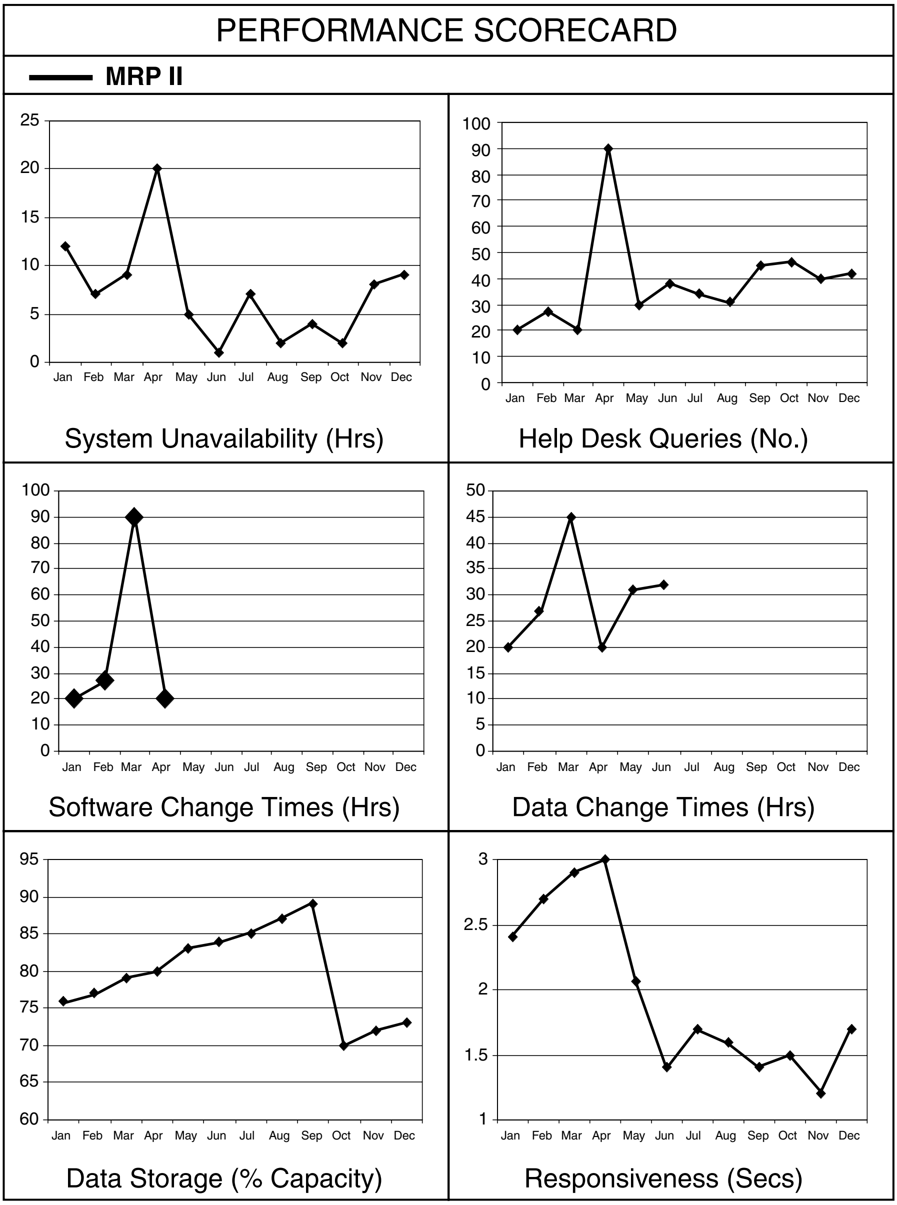


FIGURE 35.5 Performance Scorecard.

The strategy toward version upgrades and bug fixes of the MRP II product software should have been specified in the Validation Plan. Each software release should have been successfully regression-tested and be market-tested before it is used. The term market-tested has not been defined but it has been suggested that the version of a software product being used, or bug fix, has been released into the market for at least 6 months and that there are a large number of users of that particular software. The aim is to reduce the risk associated with installing new software that does not have a track record of successful operation. It has been well documented elsewhere how complex software such as that making up MRP II systems can degrade as it ages because additional functionality and bug fixes can actually introduce more problems than they solve — the “software death cycle.” Even so, it could be argued that not implementing bug fixes is negligent. Any upgrade should be conducted under change control. Release notes from the supplier will require review in conjunction with how the system is used to determine whether there is any impact. It must be recognized that revalidation of the MRP II application may be needed.

Inevitably the MRP II system will become unavailable to users from time to time. Planned maintenance activities and development activities can be managed to control their GxP impact. If the system crashes for any reason (e.g., server goes down), however, then the system will need to be recovered to a known controlled state. This usually involves rolling back the system to its last archived state. There is then a need to catch up data entry and processing to reflect what happened while the MRP II system was unavailable. Data centers providing these services are subject to the same GxP requirements in this respect as site-manufacturing the actual drug products. Backup and restoration procedures must be defined, tested, and approved. The size of the catch-up task will depend on how long ago the last archived backup was taken, the duration of the outage, and what data processing was achieved prior to the outage and, subsequently, during this period. Regular backups will reduce the necessary catch-up effort but will require more standby hardware and storage media.

Business Benefits

The business benefits of MRP II systems are well documented. Better warehouse and inventory control saved enough money during the first year of operation to pay for the MRP II implementation in one Irish pharmaceutical company. Savings in this regard by other companies are not always so dramatic but generally they are significant. Other pharmaceutical and healthcare companies may also see the benefit of automation in reducing user error and speeding up data processing. This coupled with a need to validate their existing unvalidated MRP II systems has led many companies to replace the old with new systems. Whether it is possible to successful retrospective validation of MRP II systems has been questioned during FDA inspections. The cost of retrospective validation (perhaps 20 times cost of original unvalidated MRP II implementation) and the uncertainty of satisfying regulatory inspection has led to a general replacement rather than a fix solution being adopted in industry. Not that validation is all cost. As a consequence of validation, better maintenance documentation should make modifications easier, faster, and hence cheaper. Indeed, one U.K. healthcare company managed to reduce its operation and maintenance costs for its MRP II system by about 80%.

Electronic Records and Electronic Signatures

There has been much debate in the pharmaceutical industry over a U.S. regulation known as 21 CFR Part 11 affecting electronic records and electronic signatures that became effective in 1997. Paper records have been the traditional medium for regulatory records demonstrating validation compliance. Electronic records are allowed but must be reliable and secure and facilitate bound signatures. Controls for electronic signatures are required such that they are legally equivalent to handwritten signatures. Do not assume that all electronic records and electronic signatures are compliant by default. The complexity of this issue has led many MRP II systems to fall back on paper records as masters, using electronic versions as working copies. The concept of working electronic records being incidental compared to paper copies needs careful thought as, more often than not, the electronic version is effectively the master.

Some MRP II vendors offer a special edition of their product to address pharmaceutical industry needs (e.g., SAP offer "PharmaPack"). A pharmaceutical manufacturer should work with the vendor of its MRP II system to ensure such special editions do indeed meet its needs. Cilag Pharmaceuticals recently shared its experiences with SAP in this regard.[8] Some of the issues raised were:

- Multiple entries were being posted in an audit trail for a single change to an electronic record.

- Some non-GxP records created entries in GxP electronic record audit trails.
- The large amount of information being processed and stored for audit trails was impacting overall system performance.

Any pharmaceutical manufacturer implementing an MRP II system must pay careful attention to electronic records and electronic signatures, and justify its position in a discussion document so that it can respond if challenged by a GxP regulatory authority. Hybrid solutions based on adding procedural controls and possibly supplementary software may be necessary to establish compliance with the regulation.[9]

Regulatory Inspection

Regulatory inspection of the operational MRP II system may occur several years after implementation and can be just as critical if the system is found to be noncompliant with the regulator's expectations for validation. Pharmaceutical manufacturers who have been subject to detailed scrutiny of the computer validation for their MRP II systems understand the need to validate properly. The financial consequences of noncompliance to a multinational pharmaceutical manufacturer can be immense.

Pharmaceutical manufacturers implementing MRP II systems should carefully consider briefing their appropriate regulatory authorities in advance of any potential inspection. Few regulatory inspectors would claim to be MRP II experts, and while they will understand the principles of computer validation, they may not be familiar with an individual pharmaceutical manufacturer's validation philosophy for enterprise applications. Each party will benefit from understanding the other's perspective. Any concerns or misunderstandings can then be proactively managed. Figure 35.6 presents a variant of the well-known validation V-Model for enterprise applications used in this case study — but how does this fit individual implementations?

A regulatory inspection is likely to look at the MRP II system as it is used from a site perspective rather than from a corporate perspective. The availability of site-specific document sets including Site Validation Plans and Site Validation Reports will be key. Documentation for the core system should be readily accessible. Managers must consider who owns the MRP II for the purposes of fronting an inspection and access staff who are knowledgeable on the core system and site application. Some staff will naturally move on to new jobs within and outside the company. It is very important to ensure that a critical mass of knowledge about the MRP II project is maintained within the company.

The RTM linking the specification, implementation, and testing of functional aspects of the MRP II system, together with the GxP assessments, provide a very useful tool to assure that all aspects of the system, especially the GxP impacting elements, have been successfully validated. It also provides a route map through the documentation set for the MRP II system. There can be many thousands of documents and the ability to quickly retrieve appropriate documentation during an inspection is very important. It is no good having done the validation if you cannot retrieve it for an inspector! The route map will also provide those preparing to receive an inspection with a means of reminding themselves of project and document organization so that a knowledgeable and professional front can be presented during the inspection. A high-level overview of a validation document set is presented in Appendix 35G. As an aside, the benefit of having documentation presented in a common format and in neat labeled binders should not be underestimated. Impressions count for a lot during an inspection; remember the pharmaceutical manufacturer is basically trying to demonstrate that the organization is in control.

Inspection Case History

Here are some actual observations made by the FDA during an inspection of a SAP R/3 application at Solvay Pharmaceuticals.[10] The observations were made by Thomas Arista and Robert Tollefson

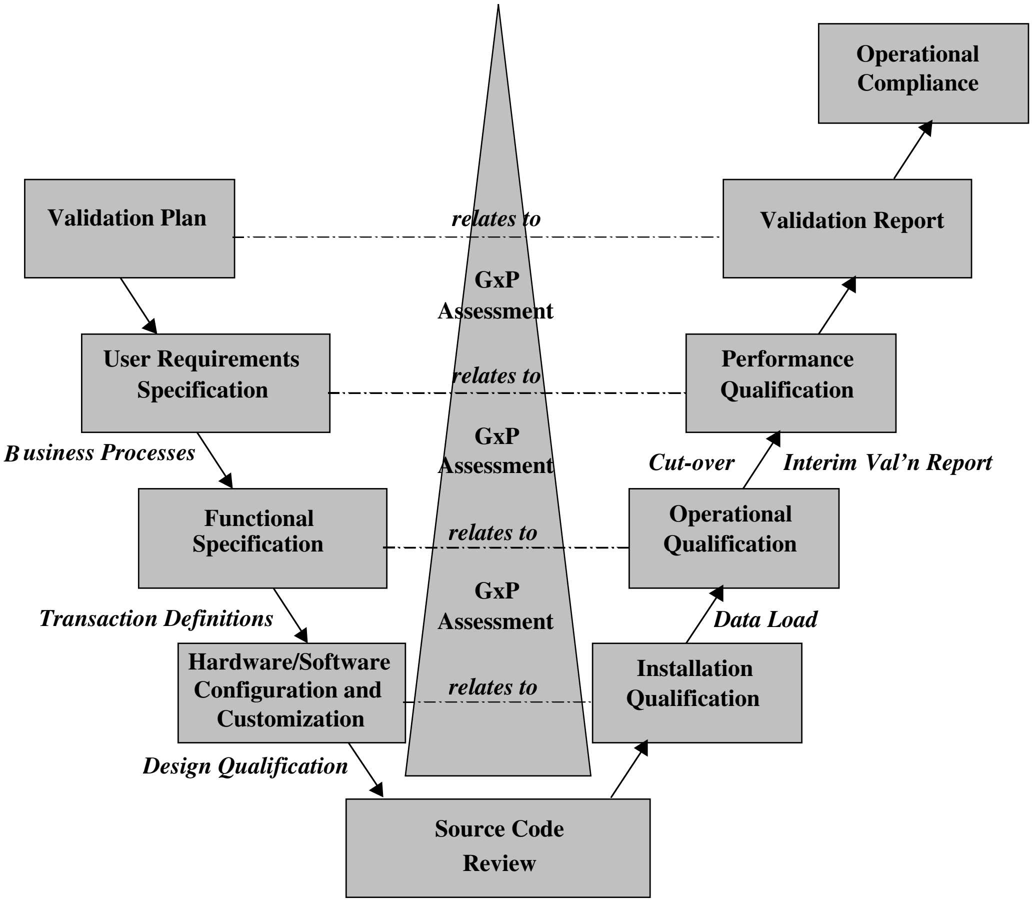


FIGURE 35.6 Validation Life Cycle for Enterprise Applications.

and recorded as an FDA 483 observation item. The SAP R/3 application was inspected over a 2-day period shortly after go-live.

- No final validation report (PQ still in process after cut-over)
- No application version number mentioned in validation documentation
- No formal approval of vendor-supplied documentation
- Not all required SOPs in place with users as formally issued copies
- No list of current approved users with levels of access

Other comments raised during the inspection by the FDA were:

- Distinguish between "validated state" and "authorized for use."
  The validation methodology did not have a rationale for justifying why the MRP II system was acceptable to support production and release after cut-over but before the PQ was complete.
- Do not refer to internal audits in validation documents.
  The FDA was possibly concerned here that because the internal audits were presented as open to inspection. This could lead to any direct comment being made to the pharmaceutical manufacturer's senior management not being as explicit as they otherwise might be.

- IT Departments will be part of future inspections.
  To date, many IT functions with pharmaceutical manufacturing companies have not been subject to inspection.
- Project documentation should adhere to good documentation practice.
  There are often many thousands of project documents for MRP II systems. They should all be subject to document (life-cycle) management including following approval processes, indexing, and archiving. Document management must cover the implementation project, operation, and maintenance of the system.
- Key project and system documents should be available at sites.
  It is generally not practical to maintain a complete set of system documents at each site that is supported by an MRP II system (remember, there may be many thousands of documents). A complete set of documents should be managed at a central location, with key stage and site-specific validation documents being formally copied to sites.
- Major milestones should be formally authorized by senior management.
  The successful implementation of a MRP II system would normally be considered business critical. As such, senior management would be expected to meet to agree on progression between major work stages, possibly connected to the release of project budget. These authorizations to proceed should be formally recorded and retained.
- Focus effort on GxP-relevant processes within system validation.
  The size and complexity of MRP II systems mean that full validation of everything is not practical. The adoption of good practice should therefore be considered for the whole system implementation with full validation of GxP directly impacting processes and functionality.

It is important to understand that these comments were passed on a limited audit of the application. The observations cannot therefore be considered comprehensive.

Concluding Remarks

This chapter has outlined the basic approach to implementing and validating an MRP II system. Managers must tailor their project approaches to match their particular business organizations, availability of in-house and external resources, scope, and size of implementation. Successful implementation, validation, and operation of an MRP II system will also depend to a great extent on ensuring the project does not:[11]

- Unwittingly compromise the standard nature of a configurable standard software product by too much customization.
- Lose control of quality during what are often fast track projects. There are often large numbers of project staff from a variety of backgrounds, not all of which are necessarily conversant with either IT systems or validation.
- Unacceptably increase project risk by business-process reengineering instead of limiting the implementation to current established ways of working.
- Ignore known shortcomings with the core supplier product (how they are tackled should be documented).
- Disregard the potential financial and validation impact of upgrading/integrating their current IT infrastructure (clients, servers, networks) to support the business system implementation.

In addition, managers should closely monitor stress levels and morale in the project team. Illness among key team members can sorely hit a project's progress, an issue that becomes ever more critical toward cut-over. The retention of staff has already been discussed in relation to fronting inspections, but key staff are also necessary to maintain and further enhance the configuration and use of the implemented MRP II system after cut-over. It is quite common for a company implementing an MRP II system to lose one third to one half of its original project team within a year of cut-over. Reasons for the departure of staff are many: permanent staff taking highly paid contractor MRP II positions elsewhere, individuals suffering from stress or uncertainty on how they will fit back into their own organizations, and individuals being poached by other companies embarking on MRP II implementation. It is very difficult to retrieve the situation when an individual has come to the point of leaving a company. It is better to actively manage to minimize the potential problem from the outset.

REFERENCES

1. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (www.ispe.org), The Hague, the Netherlands.
2. Gottschalk, F. (2000), Validation of SAP R/3 and Other ERP Systems: Methodology and Tools, *Pharmaceutical Technology Europe,* December, pp. 26–30.
3. Hambloch, H. (2000), *An Approach to Risk Assessment for IT Systems*, ISPE Conference on GAMP Concepts and Case Studies, Zurich, September 18–21.
4. Wingate, G.A.S. (1997), *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice*, Interpharm Press, Buffalo Grove, IL.
5. Wingate, G.A.S. (2000), *Validating Corporate Computer Systems: Good IT Practices for the Pharmaceutical Industry*, Interpharm Press, 2000, Buffalo Grove, IL.
6. Snelham, M. and Wingate, G.A.S. (2000), Validating Laboratory Information Management System, *Journal of Validation Technology,* 6(4), August.
7. Thompson, D. (2001), Wiring Up the Warehouse, *Pharmaceutical Visions,* February, pp. 62–66.
8. Smith, C. (2002), Validation of SAP Release, ISPE Conference on Principles and Applications of GAMP 4, Zurich, September 25 and 26.
9. FDA (2003), Part 11, *Electronic Records; Electronic Signatures – Scope and Application*, Guidance for Industry (www.fda.gov).
10. Rakhorst, W. (2000), *Validation of SAP R/3: Experiences with a Big Bang in an International Environment*, Institute of Validation Technology Conference on Computer and Software Validation, London, February 21 and 22.
11. Wingate, G.A.S. (1998), Finding a Way, *Performance Chemicals International,* 13, 10, pp. 28–30.

APPENDIX 35A
EXAMPLE PROCUREMENT BUSINESS PROCESSES WITH SOPs

Business Process	Standard Operating Procedure
Purchasing	Request Quotation
	Create/Change/Approve Purchase Order
	Purchase Item Receipt
	Purchase Order Archiving
	Set-up/Change Material/Item Details
	Set-up/Change Supplier Details
	Set-up/Change Buyer Details
Warehousing	Goods Receipts from Suppliers
	Goods Returns to Suppliers
	Customer Returns
	Warehouse Palletization
	Stock Placement & Removal
	Hazardous Material Handling
	Material Requests/Reservation/Staging for Production
	Goods Received from Production
	Return of Unused or Partially Used Materials
	Relocation Movements within Warehouse
	Stock Accuracy Checks (e.g., Perpetual Inventory)
Invoicing	Purchase Item Invoicing
	Invoice Matching, Approval, & Payment
	Damaged Goods Processing
	Spend Approval
	Quota Arrangements
Quality Management	Assign/Revoke Supplier Approval
	Managing Preferred Supplier Lists
	Quarantine Materials/Goods

APPENDIX 35B
EXAMPLE PRODUCTION PLANNING BUSINESS PROCESSES WITH SOPs

Business Process	Standard Operating Procedure
Sales & Operations Planning	Planned Order Conversion
	Schedule Creation
Demand Management	Planning
	Validate Customer Demand Data
	Planning Hierarchy Maintenance
	Review/Maintain Flexible Planning Data
	Identify Demand/Forecast Changes
	Historical Forecast Creation/Monthly Table Review
	S&OP Demand Review
Master Production Schedule (MPS)	Maintain Batch Run Parameters
	MPS & Exception Message Operation
	MPS Review
	Run MPS Manually
	Manage Write-Offs
	Customer & Dependent Demand Review
Material Requirements Plan (MRP)	Maintain MRP Batch Run Parameters
	MRP & Exception Message Generation
	MRP Review
	Run MRP Manually
	Manage Write-Offs
	Customer & Dependent Demand Review
Capacity Planning	RCCP Set-up
	RCCP Execution
	Capacity Evaluation Review
	Capacity Evaluation Issue Resolution
	Demand Version Creation
	Scenario Creation and Initiation
	PIR Management
	Plan & Capacity
	Purchasing & Financial Simulation
	Feedback to Operative System
Forecasting	Forecast Demand Control
	Replenishment
	Purchase Order & Receipt Control
	Sales Order & Dispatch Control
	Inventory/Material Requirements Control
	Forecast Demand Control
Maintain Master Data	Manage Forecast Master Data
	Manage MRP Master Data
	Manage MPS Master Data

APPENDIX 35C
EXAMPLE MANUFACTURING BUSINESS PROCESSES WITH SOPs

Business Process	Standard Operating Procedure
Process Order Management	Process Order Creation
	Process Order Preliminary Costing
	Process Order Approval
	Process Order Archiving
Manufacturing	Release Individual/Collective Process Orders
	Material Ordering & Staging
	Print Shop Floor Documentation
	Task-List & Work Center Processing
	Issue Material to Process Order
	Missing Parts Processing
	Product Labeling
	Batch Record Processing
Packaging	Release Packaging Orders
	Material Ordering & Staging
	Print Shop Floor Documentation
	Task List & Work Center Processing
	Issue Product and Packaging Materials to Process Order
	Missing Parts Processing
	Packaging Labels and Documentation
	Batch Record Processing
Labeling	Create/Change/Approve Labels
	Print/Reprint Labels
	Reconcile Labels
Waste Management	Back-Flushing
	Rework
	Disposal
	Stock Reconciliation
Quality Assurance	Inspection Checks
	QC Sampling and Sample Labels
	Certificates of Analysis
	Variance/Defects Reporting
	Change Management
	Retest Materials
	Changing Shelf Life of Products
	Reassigning Products
New Product Introduction	Annual Product Review
Maintain Master Data	Maintain BOM Master Data
	Maintain Master Recipes

APPENDIX 35D EXAMPLE SALES AND DISTRIBUTION BUSINESS PROCESSES WITH SOPs

Business Process	Standard Operating Procedure
Presales Handling	Customer Records
	Pricing Data
	Customer Purchase Restrictions
	Product Equivalence
	Batch Allocation
	Foreign Trade Processing
Direct Sale to Consumer	Creating an Inquiry
	Providing a Quotation
	Order Placement
	Shipping Processing & Tracking
	Transportation Processing (including Freight Forwarding)
	Billing
	Inter-Site/Company Transfers
	Export Sales & Special Documentation
	Contract/Tender Management
	Toll Sales
	Free of Charge Sales
	Rush Orders
	Bonus Goods Allocation
Third-Party Order Processing	Creating an Inquiry
	Providing a Quotation
	Order Placement
	Transportation Processing
	Billing
Returns Processing	Batch Tracing
	Canceled Customer Orders
	Recall Processing
	Returns Order Processing
	Bill Corrections (Credit & Debit Notes)

APPENDIX 35E
EXAMPLE FINANCE BUSINESS PROCESSES WITH SOPs

Business Process	Standard Operating Procedure
Asset Management	Asset Master Record Maintenance
	Asset Acquisitions & Capital Expenditure
	Asset Decommissioning
	Depreciation Simulation (including Tax)
	Asset Calculation
	Asset Revaluation & Write-Ups
	Insurance Revaluation Calculation
	Investment Support
	Period End Closing & Reporting
Revenue and Cost Controlling	Actual vs. Planned Operating Costs
	Cost Center Allocation
	Profit Center Balance Sheets
	Creation & Maintenance of Internal Orders
	Budget Values & Availability Controls
	Commitment Accounting
Product Costing	Stock Valuation
	Planning and Comparison
	WIP Valuation
	Variance Calculation
General Ledger	Account Maintenance
	Posting of General Ledger Journals
	Period End Processing (Day, Month, Year)
	Financial Reporting
Accounts Payable	Supplier/Vendor Data Maintenance
	Invoice Verification
	Matching Purchase Order to Invoice
	Down Payments on Purchase Orders
	Settlement (Payment) of Supplier/Vendor Account
	Invoice and Credit Note Processing
Accounts Receivable	Customer Data Maintenance
	Invoice and Credit Note Processing
	Customer Down Payments
	Bills of Exchange
	Letters of Credit
	Debt Collection
	Customer Settlement (Payment) of Account

APPENDIX 35F
EXAMPLE MRP II GXP IMPACTING FUNCTIONALITY[4]

GxP Impacting Functionality

Manufacturing and Packaging	Master Production Schedule
	Routings
	Shop Orders
	Issue Materials against BOM
	Batch Release
	Labeling
	Package Product Documentation
Supplier Management	Purchase of Materials
	Order Amendments
	Repetitive Supplier Scheduling (Approved Supplier Status)
Warehouse/Inventory Management	Goods Receipt
	Quality Control Inspection
	Movement of Materials/Products
	Location Creation
	Movement of Work in Progress
	Movement of Finished Goods
	Returns to Supplier
Quality Control/Quality Assurance	Release of Materials/Products to Production
	Quarantine of Materials/Products
	Scrapping of Materials/Products
	Testing of Materials/Products
	Retesting of Materials
	Release of Materials (QP Release)
Distribution	Customer Batch Allocation
	Product Returns
	Customer Complaints
	Adverse Event Reporting
	Batch Investigation
	Recall

APPENDIX 35G
EXAMPLE MRP II VALIDATION DOCUMENT SET

Validation Master Plan

- Project Procedures (GAMP)
- System Overview
- Supplier Audit (Application Vendor, Service Providers)

URS

- URS with Business Process Models
- GMP Assessment (Risk Assessment)
  - Part 1: Business Process Evaluation

Functional Specification

Design

Implementation

- Functional Specification
  - Procurement
  - Production Planning
  - Manufacturing
  - Sales & Distribution
  - Finance
- User Procedures
- GMP Assessment (Risk Analysis)
  - Part 2: GMP Tests for SAP R/3
  - Part 3: GMP Tests for Associated Interfaces
- Configuration Definition & Management
  - Software Components
  - Hadware Components
- Bespoke Reports and Forms & Other Custom Code
  - Software Design
  - Programming Standards
  - Source Code Review
- Vendor Manuals
  - Standard Software
  - Standard Computer Room Hardware
  - Standard IT Infrastructure
  - Standard Database
- Training and Education Materials
- Requirements Traceability Matrix

Qualification

- Design Review (Phased?)
- IQ (Initial & Upgrades)
- Data Load Reports
- Conference Room Pilot
- OQ (Phased?)
- PQ (Phased?)

Validation Report

- Validation Report
- Glossary of Terms
- Document Index & Navigation Aid
- Updated Training Records
- Support Procedures (System Management, Change Control, Disaster Recovery, Security Practice, Performance Monitoring, Periodic Review)

36 Case Study 18: Marketing and Supply Applications

Louise Killa, LogicaCMG

CONTENTS

The data held within marketing and supply applications are key inputs into the efficiency of the supply chain and provide the vital link that connects Information Technology to the physical world of raw materials, intermediate stock, finished inventory, business processes, and people. What is done with that data — how data are collected, processed, communicated, stored, or otherwise manipulated, determines their true value to any organization as Information Technology takes its place as an enabler for efficient enterprise coordination.

Contemporaneous control of these activities should also enable business benefits to be achieved such as increased productivity arising from more efficient use of key equipment and personnel, greater accuracy and elimination of common errors, and the possibility of lowering stockholding levels without risk to customer service. However, organizations must ensure that such benefits are not achieved at the expense of regulatory expectations.

The computerized systems used by pharmaceutical organizations are expected to operate in accordance with their intended design to reliably and consistently support the regulated business processes. Whether it is agreeing on artwork with the marketing organization, communicating medical information on licensed products, placing a purchase order for the supply of products, reading the barcode on the item, tracking its movement through the storage and distribution network, capturing relevant data at various transit points throughout the supply chain, or recording customer complaints, in order to retain the various licenses that are required to operate in the pharmaceutical markets around the world, the various regulatory authorities require to see formal evidence that these systems have been validated to confirm their suitability for use.

In recent years, there has been an increasing regulatory focus on the marketing and supply aspects of the pharmaceutical supply chain. While these activities can be carried out manually, it is more commonplace to find that a computer system is used to support them either wholly or partially (a hybrid system). The reliance on these systems as the sole mechanism of recording information means that they should be developed to an appropriate level of compliance and validated for their intended use to ensure that they are able to provide a consistent output to support the regulated process.

This study outlines the considerations that need to be made when determining the validation activities that need to be undertaken on these types of systems which wholly or partially provide functionality to support the regulated processes outlined in Figure 36.1.

MARKETING APPLICATIONS

Marketing applications used within the pharmaceutical and healthcare industry include those computerized systems used to support international artwork and the provision of medical information supporting pharmaceutical and healthcare products released to market.

Artwork needs to be agreed upon between the pharmaceutical, medical device, or healthcare company and the marketing organization. Traditionally this has been achieved using fax and/or e-mail attachments. More recently the Intranet and Internet has been used to facilitate such transactions in conjunction with electronic document management systems (EDMS).

A variety of applications may be used to support the artwork process within an organization in order to ensure that it is able to rapidly respond to changes that are required to meet appropriate market regulations. These range from different standard artwork generation packages, supporting templates, portable document format (PDF) technology, and corresponding applications capable of reading and editing these files, and the various e-mail and infrastructure applications used to facilitate the secure transmission of the artwork to the approved external studios.

Additionally, where an organization may find itself managing a significant amount of artwork changes and approvals, a workflow tracking system may be used to determine the status of a particular piece of artwork at any point in time.

When using workflow media, care needs to be taken to allow for the fact that colors that are displayed on a computer screen may actually look different if printed out on different printers; this can at times give a false confidence regarding the acceptability of the material, which might differ again in the final printed output.

All artwork produced should take into account the technical requirements of the relevant printing process and the materials used. Failure to do so could result in a process that could present problems if the validation of activities on supporting systems is required.

Because of its potential to have a considerable impact on patient safety, artwork processes generally involve several stages of critical checks within the pharmaceutical organization to ensure that the final copy is as error free as possible. The completion of these checks is recorded in supporting workflow applications at key checkpoint stages.

Depending on the level of technology involved in the supporting system, approval may use a hybrid form involving the appendage of a physical signature on printed output to authorize its progression to its next stage, or may use encryption technologies such as electronic signatures.

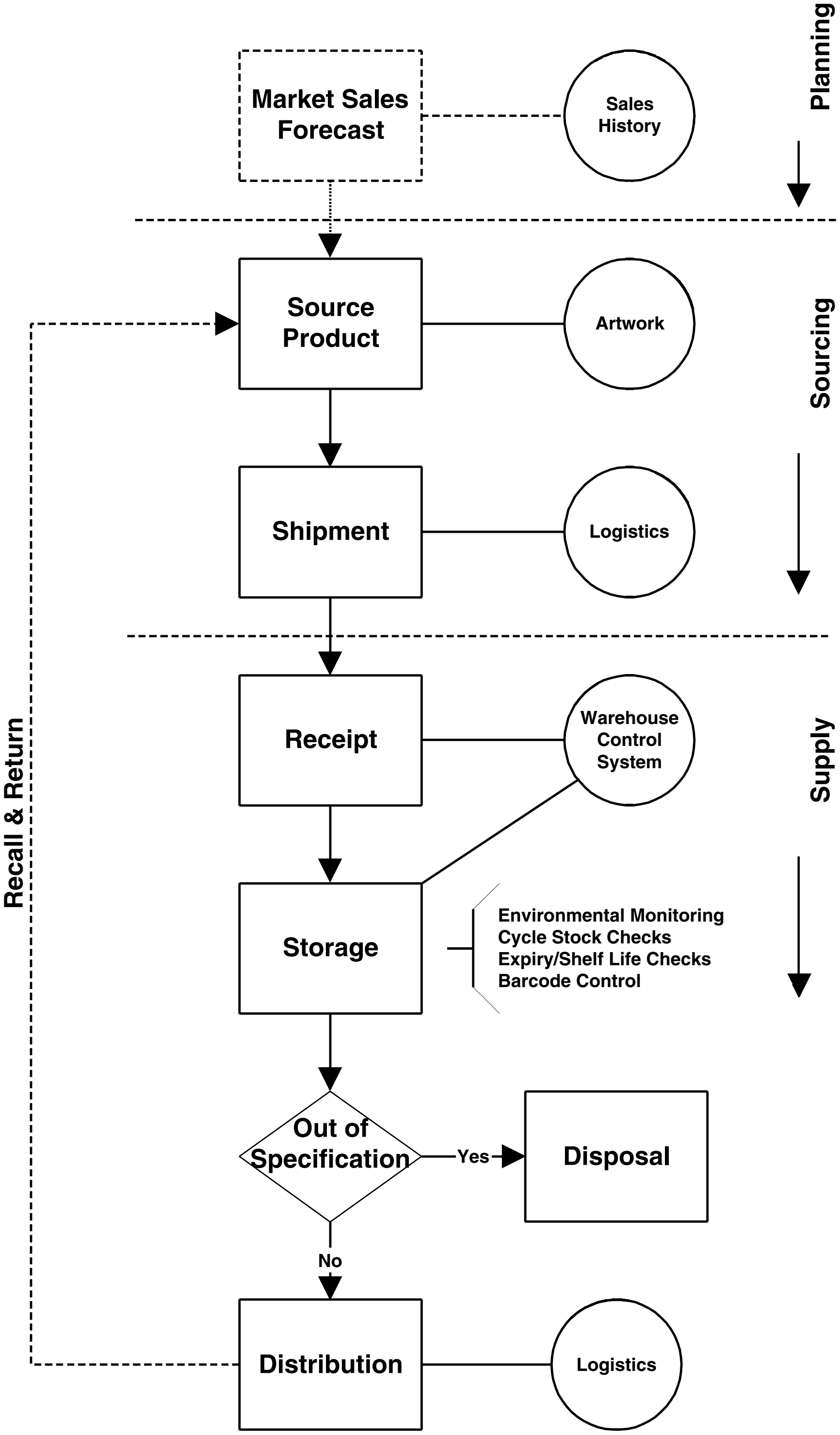


FIGURE 36.1 Context of Supply Application Functionality.

Medical Information supporting products released to market can exist in many different formats. The medical information produced for the finished product, whether in label format or as a package insert, should correspond with the actual ingredients that the product is formulated to contain[1] and the method and dosage that corresponds with its license.

Regulatory authorities have also stated[2] that labeling is not limited to the immediate product container or package insert, but also includes all promotional material that is distributed and/or published in connection with a particular product or medical device.

The Internet has become increasingly popular as a means of publishing medical information about pharmaceutical and healthcare products, including press releases, sales sheets, brochures, and advertisements. Such statements should be backed up with process, controls, and checks to ensure the information displayed is accurate and relevant to the product.

Statements made by, or on behalf of, marketing organizations during promotional audio conferences will also attract regulatory attention[3] if they minimize crucial risk information and promote a drug or device for an unapproved new use. Incorrect, misleading, or incomplete artwork and medical information could lead to the inappropriate use of a drug or medical, device, in which case a recall of that product is usually required. These systems should therefore be validated as they can impact public health.

Other marketing applications that may fall within the scope of regulatory scrutiny are those used to record complaints received regarding drugs or medical devices, and any other EDMS applications and systems that retain and develop documentation used in dossier submissions.

Not every marketing process supported by a computer system would fall within the scope of regulatory scrutiny. Computer systems such as forecasting applications, cost control systems, and systems that are used to purely support only financial or accounting activities may not need to be validated at all. However, each of these systems is still expected by industry regulators to undergo some form of formal documented regulatory assessment to determine whether or not this is the case.

SUPPLY APPLICATIONS

In addition to corresponding changes within manufacturing operations, in recent years the warehousing and distribution aspects of the industry have tended to become more automated. This has minimized the need for human intervention, thereby eliminating many user errors, and has removed the reliance on paperwork for the operation of the plant and the management of the many specialist products and suppliers, leading to improvements in productivity.

The Distribution Requirement Planning (DRP) process that is supported by supply applications focuses on the flow of goods through the downstream channels of the supply chain. Planning information regarding the demand for goods cascades from customers to licensed local distribution depots, and from these to a main warehouse and finally back to the supplier's factory (or external source of supply), taking into account ordering and delivery between the levels. These systems can be seen as a front-end extension to a Manufacturing Resource Planning (MRP II) system.

The supply applications discussed in this study are those applications within the overall supply chain that are used to support the sourcing, receipt, storage, disposal, and distribution of licensed products and may also be used to support product recall activities, should this activity become necessary. The principles discussed in this chapter would apply equally to applications that are used to support the receipt, storage, disposal, and distribution of raw materials, intermediate products, and material for clinical trials.

The application solution selected by an organization to support its supply process may consist of one integrated application that has different modules capable of handling all elements of the supply process. Alternatively, it may comprise a series of separate complementary applications that have been linked together by a series of system interfaces and business processes to provide an overall solution to the business.

The connection of these systems within and between organizations is key to the successful operation of the supply chain because it allows important information to be shared by suppliers, partners, distributors, and even customers, who may view data to see the progress of their orders.

Typical systems that support the business in supplying its inventory to licensed wholesalers or distributors may include the following.

A **Procurement/Purchasing System** that manages the sourcing of items from approved suppliers. This is a function that has traditionally generated considerable quantities of paperwork in order to communicate information from one function to another in order so as to facilitate action, to indicate requirements to suppliers, and to obtain the necessary goods required by the supply chain on time and to specification.

In recent years purchasing has been recognized as playing a more strategic, rather than a purely transactional, role within an organization. The advent of more integrated purchasing software applications has facilitated integration with other applications to improve the performance of the overall supply chain.

Goods and services should only be sourced from suppliers assessed as being capable of providing materials products and services that meet the standards of the organization. Regulatory authorities expect the organization to have a documented system for the assessment and approval of the suppliers appropriate to the type of item or service being sourced. If the procurement process followed is supported by a computer system that records the approval of the potential supplier, then the usage of this computer system is expected to be validated.

Applications used to support the procurement process for items having a regulatory impact should clearly indicate the approval of a supplier and should prevent the selection of unapproved suppliers for products and services that have been deemed to have a regulatory impact. The validation of the system will be expected to demonstrate that this relationship is correctly established within the system.

Any computer system responsible for supporting the shipment of the items into the supplying warehouse, or for the return of goods (for whatever reason), is generally under the direct responsibility of the supplier of the goods and is likely to exhibit the characteristics of the distribution system outlined later on in this section. However, where the goods being shipped fall within the scope of regulatory scrutiny, the data provided by any shipment applications is expected to be created and maintained in line with regulatory requirements in order to meet these requirements.

An **Inventory Management System** that contains all relevant supplier details to enable the receipt of goods, checking against purchase orders placed, allocating suitable storage locations, any requirement to repackage or relabel in order to meet local market requirements, inventory status management, segregation of products following recall or return activities, stock rotation and associated checks, disposal and stock level monitoring, and replenishment from the source of supply. This system would also record any status changes necessary to reflect the activities that are carried out at this stage of the supply chain, including approved release by the Qualified Person (QP) or Quality Control (QC) function.

The complexity of such systems can vary greatly from large-scale turnkey solutions to simpler PC-based packages. Systems available also include front-end applications capable of adding real-time communications, bar code scanning, and advanced productivity management to existing legacy systems. Advanced systems often incorporate an alerting capability that is able to monitor and detect when specific programmed events need to happen (e.g., replenishments needing to be completed 2 h before a picking cycle commences), or when expected events fail to materialize (for example, the late arrival of an expected urgent delivery from a particular supplier).

Depending on the requirements of the operation, this system may also include breakdown or aggregation of inventory received into smaller or larger stock keeping units (SKUs) for onward distribution. It may also include interfaces to purchase order systems, labeling applications, automated materials handling systems such as sortation systems, stock location systems, automated storage and retrieval systems, Radio Data Terminals (RDTs), Radio Frequency Identification units

(RFIDs), Automated Guided Vehicles (AGVs) or conveyor systems, and could involve integration with devices such as barcode scanners, balances, asset tracking solutions, and microchip identification systems.

A **Warehouse System** encompassing all aspects of the management and maintenance of the storage facility in line with the relevant regulatory and local health and safety expectations. This will encompass maintenance of standards of cleanliness, sterile areas, temperature, humidity, pest control systems, and physical security (including the restriction of access to controlled drugs). This system may interface with physical alarm systems for the building and other automated access control systems (e.g., swipe-card systems).

A **Sales Order Processing System** that contains all relevant customer and inventory details to enable a sales order of the correct characteristics to be placed against the actual stock available to fully or partially satisfy the order. The characteristics of the contract between the company and each customer may differ greatly; for example, some customers may only be licensed to receive a very limited subset of the total inventory held within the system, while others may be licensed to receive everything but a few specified inventory items available from the supply chain. In both cases, the system would be expected to contain functionality that allows each type of circumstance to be correctly set up to ensure that sales orders for onward wholesale suppliers are only raised for inventory that the supplier is licensed to hold.

Additionally, the contract to supply different licensed customers may specify different supply terms for the same inventory item. For example, a pharmacy wholesaler may only accept deliveries of orders containing inventory with at least a 12-month expiry date, while a grocery wholesaler may be willing to accept inventory with a minimum 6-month expiry date. Therefore the Sales Order Processing system would be expected to differentiate between the stock rotation dates of the inventory made available to satisfy the orders placed by each type of customer.

The Sales Order Processing system is likely to be one of the prime sources of information used should any product recall activity become necessary. Depending on the type of Sales Order Processing System, it may have interfaces to other systems that electronically feed in sales orders to the organization and to automated or manual order picking systems.

A **Distribution System** to ensure that the items specifically picked to satisfy a sales order are successfully and safely delivered to the required destination with no deterioration in product quality or risk to public safety. The system should ensure that orders can be tracked throughout their distribution cycle to enable clear control and traceability to be demonstrated when required. This system may be required to interface with logistics management systems run within the organization or by third-party logistics suppliers. This system may include functionality that interfaces with a separate Proof of Delivery (POD) System and financial systems containing general ledgers managing the payments to distributors for services received and invoicing to licensed wholesalers for the goods they have ordered.

The distribution system may also be the initial point at which returns to the warehouse reenter the supply chain system, for example, if goods are supplied to wholesalers on a sale or return basis. Depending on the functionality and interfaces of the system, upon completion of successful deliveries, it may provide confirmation of this activity back to the Sales Order Processing System.

Any system used by representatives from manufacturing or distributing organizations to record the distribution of drug samples to licensed healthcare professionals upon request would also fall within this category.

REGULATORY REQUIREMENTS

The validation of marketing and supply applications encompasses exactly the same fundamental activities as any other validation exercise carried out on a system supporting activities within the supply chain that have a regulatory impact. Addressing these activities should ensure that the software is developed, and adequately tested and maintained to remove the likelihood of system failure, and that the system possesses the necessary resilience to ensure a rapid and complete return to operation in the unlikely event that a failure does occur.

Requirement to Validate Marketing and Supply Applications

Depending on its use, supply chain applications supporting the business processes of marketing, product sourcing, shipment, receipt, storage, disposal, and distribution could fall under the scrutiny of various regulatory authorities covering the countries in which the organization operates and those to which it supplies products.

Within the European Union (EU) there are several regulations[4–10] that govern marketing and supply operations. Within the U.S. there are different regulations[11–18] that govern marketing and supply operations, although the principles are broadly the same in covering the manufacturing, holding, and distribution of products and medical devices and their associated records. Regulatory citations such as the ones below are not uncommon:

> *You have failed to validate XXXX computerized systems used for drug product distribution information.*[19]
>
> *Failure to validate the computerized system used by your firm to track drug products from receipt through distribution, in accordance with 21 CFR 211.100(a). This computerized system is also used by your firm to generate the "Unique Barcode Labels" that are applied to cases containing drug products, and/or to the individual drug product containers, as part of your relabeling operations.*[20]
>
> *Failure to exercise appropriate controls over and to routinely calibrate, impact, or check automatic, mechanical, or electronic equipment used in the manufacturing, processing, and packaging of a drug product according to the written program designed to ensure proper performance (21 CFR Part 211.68) in that the Installation Qualification (IQ), Operational Qualification (OQ), or Performance Qualification (PQ) for the XXXXX was not performed.*[21]
>
> *Prescription drug products stored at your firm, with temperature range controls from 68°F to 77°F, were not held in accordance with the label requirements to ensure the identity, strength, quality, and purity of the drug products as set forth in 21 CFR Part 211.142(b). There were no reading logs or data of temperatures or relative humidity conditions of the warehouse since March 13, 2001 as required. The temperature indicator at your firm during the inspection indicated a storage temperature of 81°F. The air conditioning system is not run continuously and is turned off overnight, weekends, and holidays.*[22]
>
> *Failure to have a written individual record of major equipment cleaning, maintenance, and usage [21 CFR 211.182]. There were no equipment logs for the bar code scanner, programmed label reviewer, and the roll splicing equipment.*[23]
>
> *Your firm failed to conduct quality audits at the intervals listed in your Quality Audit Procedure xxxxxxxx, to verify that the quality system is effective.*[24]

Developers of applications that are intended to be used by organizations operating under the scrutiny of more than one regulatory body should ensure that the validation activities within their project life cycle encompasses the requirements of all these organizations. This will ensure that each of the intended end-user sites can successfully complete its on-site validation activity. Appendix 36A identifies some of the possible GxP business processes supported by marketing and supply applications.

Requirement to Follow Standard Operating Procedures and Quality Standards

Failure to adequately document standard working practices is a deficiency commonly cited by regulatory authorities.[20,25–27] Recent FDA inspections have indicated that in some circumstances no attempt has been made to cover this activity:

> *Failure to establish procedures for the warehousing and distribution of stock.*[28]
>
> *Failure to establish written operating procedures for drug production and process control steps. For example, the receipt and handling of drug components and containers ....*[19]
>
> *Failure to establish written procedures for the receipt, identification, storage, handling, sampling, examination, and/or testing of labeling and packaging materials; preparation and printing of labels, examination and review of labels; disposition of rejected labeling; issuance of labeling, reconciliation of quantities of labeling issued, used and returned; destruction of unused labels bearing lot numbers; and the 100% visual inspection of labels hand applied to drug products.*[21]
>
> *Failure to establish written procedures for the monitoring of temperatures, humidity, and the air handling system in the production area.*[21]

The absence of written operating procedures is particularly relevant to the validation and operation of bespoke computer systems used to support warehousing and distribution processes. Regardless of how PC literate an individual may be, it is unlikely that any staff members would be immediately able to use a bespoke warehousing and distribution application without some form of training. SOPs form an important part of this training. Failure to establish SOPs can also result in Warning Letter citations.[19,25,27]

Electronic Record and Electronic Signature Requirements

Regulatory authorities only require pharmaceutical and healthcare organizations to satisfy any relevant Electronic Record and Electronic Signature (ERES) considerations such as those in the respective EU[7] and U.S.[11] regulations if the usage of the application has been deemed to support a GxP critical process.

Where this is the case, the organization is expected to be able to provide evidence that it has assessed these applications against all relevant ERES requirements and initiated satisfactory remediation plans to address any issues identified by this assessment. It is essential that controls are established to ensure the authenticity and integrity of the electronic records and, where appropriate, any associated electronic signatures used as the legally binding equivalent of a handwritten one.

Organizations should ensure that they are aware of exactly which activities they capture electronically within their marketing and supply applications would fall within the scope of any relevant ERES regulations. Records transmitted by electronic means such as fax, or word-processed documents that are subsequently printed, authorized, and maintained as paper records, may not always fall within this category. Mechanisms should be put in place for prospectively assessing the ERES capability of any new applications that they are intending to commission for their use.

Even if an application does not have a GxP regulatory impact, it may still need to meet other local ERES requirements if this is a mandatory criterion for the organization for any other reason, for example to meet the expectations of local financial, legal, or health and safety regulatory authorities. Proof of Delivery (POD) functionality using a component capable of capturing a recipient's signature with a stylus such as a handheld Personal Digital Assistant (PDA) might be one such example of this.

VALIDATION LIFE CYCLE

Validation Determination

Authorities regulating healthcare companies expect regulated companies to have adequate documentation to support the GxP/non-GxP determination processes recorded for each of their systems. Also, for every system used to support a regulated process, a corresponding documented ERES assessment is expected to be present.

The business process that the system supports should provide the primary determination regarding whether or not any validation activity is required to be undertaken on the system. Each system that has been identified must be assessed to see whether it performs quality or business critical functions, whether there are sufficient controls in place to ensure its performance, and whether it is required to be validated or not.

This assessment can only realistically be performed by representatives from the business community within the organization who have an understanding of the business processes undertaken. Other personnel who have specific expertise in the regulatory expectations surrounding that particular activity may assist staff. The assessment should consider the effect of the use of the system with respect to

- Product purity
- Product identity
- Product efficacy
- Patient safety
- Regulatory submission process

In addition, organizations need to consider if the business process could be used as the basis for any regulatory discussion even if it is not the primary purpose of the process.

Only when it is confirmed that the business process requires validation do the regulatory requirements for any system (or manual process) that supports it need to be considered in any further detail.

VALIDATION PLAN

Successful validation of computer systems cannot be built in as an afterthought; validation planning should commence as soon as possible after the requirement to validate the application has been determined. Retrospective validation is far more expensive and resource intensive than prospective validation, with no guarantee that regulatory expectations can be satisfied at the end of it.

An overall Validation Master Plan (VMP) for a marketing and supply application should be established. It is essential that the plan extends to encompass all of the components of application including hardware network, infrastructure, interfaces, and other systems that are necessary for its successful and continued operation.

The user site should ensure that the scope of validation reflects the actual use of the system by the end users, and not just the intended use documented. For example, an off-the-shelf warehouse application may contain functionality developed to meet the requirements of many different companies and so could actually contain far more functionality than is actually required by the end users from one particular company. Users may select this functionality if they believe that it provides them with additional features that would enhance their business processes in preference to the intended documented usage that the organization intends to validate. Such functionality should be disabled, or users should be expressly prohibited from using this alternative functionality through SOPs and training.

Care should also be taken to ensure that GxP data are not being extracted from the application manipulated in an uncontrolled manner by another application, and then input back into the original application. Situations such as this destroy the data traceability necessary to meet regulatory requirements. They often arise because users have developed a local workaround to a system problem rather than reporting the issue and getting it resolved and fixed properly by the nominated support team.

Subordinate Validation Plans may be developed for individual user sites or for some other logical business grouping, e.g., a particular system type within the same site, such as all DRP systems, or the computer systems operated by a particular business unit or department within the supply chain covering multiple sites. Where services have been outsourced to external third parties, the relevant aspects of the work of the third parties are required to be verified by a Supplier Audit of each party that forms part of the overall Validation Plan.

Validation Plans should reference any governing Validation Master Plan (or Quality Plan as appropriate) in addition to defining site-specific validation activities. Both Site Validation Plans and central Validation Master Plans should clearly differentiate central team accountabilities and deliverables from site validation accountabilities and deliverables.

Site Validation Plans should define where external supporting documents will be used in support of site validation activities. Central and site teams should agree upon relationships between their activities and documents and should work together in order to ensure that all required activities are covered. If the Validation Plan covers an identical usage of a system on more that one site, consideration should be given to developing templates to ensure that activities are performed in a consistent manner across the organization and to maximize use of central documents without duplicated site effort. Where responsibility for validation activities is shared in this way, each site should maintain an entry in their system register for those applications/products they use that are centrally developed/supported, with the central support groups maintaining a corresponding system register of sites using the GxP systems they support.

Configuration Management

A configuration management plan should be established to outline the process to be followed to ensure that configuration and version control is established for the software, development tools, and supporting documentation that are used. This will enable accurate configuration baselines to be taken at specified points in the software development life cycle.

It is important that electronic master copies of documentation are placed under configuration control to prevent accidental changes being made to current or future revisions of a document.

All staff should undertake GxP training so that they are aware of the significance of the expectations placed by regulators on the accuracy and fitness for purpose of the software they are developing and the reasons behind why the development life cycle needs to be undertaken without any unapproved or undocumented deviations.

User Requirements Specification

The User Requirements Specification (URS) for any system is typically written by the business community (users) and describes what the system is intended to do. This is a key factor for consideration in any system validation determination as it should indicate whether the business process the system is intended to support has a regulatory impact. The requirements outlined in the final URS document will subsequently be tested by user qualification.

Site needs should be established and documented in central URS and site-specific URS as appropriate. Wherever possible a common set of user requirements should be developed between sites using the same system. Specific local regulatory requirements should be clearly stated in addition to any other statutory requirements necessary to satisfy other bodies such as the financial reporting requirements of the Inland Revenue within the U.K.

Outsourcing systems development is becoming increasingly common, and where this is the favored option, a version of the URS should be included in the Invitation To Tender (ITT) sent to potential Software Suppliers, and must clearly distinguish between mandatory requirements and requirements that may only be preferable if the design could accommodate them. The outsourcing organization must ensure that the companies that have been invited to tender clearly understand the implications of developing the software that is required to meet regulatory expectations and that the company has the necessary processes in place to support this.

Supplier Audits

Organizations that have decided to outsource key elements of their regulated computer systems should ensure that only approved suppliers capable of consistently supplying software products and services that meet regulatory expectations are used. Contracts placed with the supplier should include provision for ensuring immediate and ongoing regulatory compliance. This is no different from the expectations for suppliers satisfying other types of supply criteria within the pharmaceutical and healthcare supply chain,[29,30] and regulatory authorities may request documented evidence to support the supplier selection process.

Regulatory authorities will also expect to see evidence that the software supplier has been audited against appropriate documented assessment criteria by suitably qualified individuals before the products are used in the production environment. This assessment is expected to be performed again at regular intervals after the initial assessment to determine the supplier's continued suitability to supply and support the software.

Central development and support teams should ensure that the quality of suppliers supporting central activities have been assessed. Centrally organized supplier audits should be conducted in conjunction with regulatory groups that are familiar with the organization's operating model. Suppliers supporting local modifications should also be subject to supplier assessment.

Functional Specification

The Functional Specification (FS) for any system is typically written at a detailed level by the supplier (developer) and describes what the system is intended to do. The requirements outlined in the final FS document will be subsequently tested during user qualification.

The signed FS forms the agreement between the technical staff and users that the requirements stated in the URS have been correctly understood, and provides a level of confidence that the intended system will meet user requirements. If the development of the application is being outsourced, a preliminary version of the FS may be included in the supplier's response to the ITT, although the final version is expected to be prepared by the successful software supplier in conjunction with the user.

When considering the requirements specified for the functionality of a system, the planned local usage of any report writing and user configurable utilities present in the software should be assessed in order to determine whether or not this usage is required to be subject to local validation activities. The basic functionality of utilities of this type does not generally contain any GxP data types, although they should have some form of user controls placed on their use. However, the business processes that determine the way in which these utilities are used on site could make them subject to regulatory requirements, and if so, they would need to be validated.

Identification and Segregation of GxP and Non-GxP Data

Where the use of an application has been assessed as having a GxP impact, it does not necessarily mean that every module or area of functionality within the software has a regulatory impact.

It is acceptable for an organization to determine that it only needs to validate a subset of the overall software used (that part with functionality that has GxP impact) and not the whole application if that software is distinct. Appendix 36B indicates the data within marketing and supply applications that typically have a regulatory impact. However, regulatory authorities will expect to see evidence of some form of impact or risk assessment that has determined that this segregation will not present a risk to the successful operation of the system in a regulatory environment.

System functionality that is common to both regulated and unregulated application modules, such as system security and menu access, should always be considered as having a GxP impact.

Not all of the data identified as having a GxP impact may be considered to be GxP-critical by an organization, and the organization may undertake a risk-based approach and subsequently decide to validate only the data considered to be GxP-critical in line with its own usage. If preliminary assessment identifies a significant amount of data as having a GxP impact, it may be of little benefit to proceed with a full GxP data segregation exercise, as it may involve fewer resources and overall effort to consider the whole application as having a GxP impact.

Regardless of whether the specific module has been determined as having a regulatory impact or not, all modules within an integrated application will be expected to follow consistent change control and configuration management processes.

Design Specification

The Design Specification should be a complete definition of the equipment or system in sufficient detail to enable it to be built and is written by the developer (or supplier). Unless the system developed is very basic, it is usual for the design to be broken down into several different documents. Where this is the case, regulatory authorities expect an overall summary of the system design to be generated showing traceability between these documents. A documented Design Review should also take place to confirm that the design meets the stated requirements.

Throughout the life cycle of a project, it is common to find that this documentation evolves as the project progresses to incorporate subsequent approved changes or to correct design defects that are found during any phase of the testing process. One of the commonest regulatory failures cited regarding system design is that the software development documentation set is not kept up to date, or where it has been updated, does not bear any approval signatures.

The requirements outlined in the final Design Specification document will subsequently be tested during user qualification to verify that the correct equipment or system is supplied to the required standards and that it has been installed correctly.

Software Coding

The development of validatable applications requires evidence that the code is an accurate translation of the intended design of the system. Inspections involving computer systems frequently involve an analysis of the coding work undertaken.

Every project should indicate the mandatory documented program coding standards, directory structure standards, file naming conventions, and configuration management processes that are expected to be undertaken for all coding output. All of these documents should be readily available, either in electronic or hard copy, for all staff to consult as necessary.

All software development tools and the configuration management tool used should be assessed for suitability and fitness for purpose and this assessment should be documented.

All code should be subject to configuration and version control, and where command files or compilers are necessary to facilitate the software build process, these should also be controlled in the same way. The lack of any obvious or consistent version control is often cited as a regulatory deficiency.

Once the program code has been compiled, it is expected to undergo an independent documented coding review to confirm that it meets its design requirements and has been developed in accordance with the relevant coding standards. If these conditions are not met, the coding review should document the deficiencies and the code should then be reworked until it has been verified that these conditions have been met.

Development Testing

The testing of an automated system is required to be performed at several levels that should demonstrate that the controlling specification has been implemented correctly. Successful completion of a particular testing phase will therefore allow the project to progress to the next phase of the project life cycle. Testing should also encompass the testing of the hardware and infrastructure as well as the software itself.

Testing should be fully documented, including approved, detailed test specifications, test cases, and the results of testing in the form of signed test sheets and raw data to provide a complete record of the testing undertaken.

As part of the validation activity, testing should be independently checked to confirm that it demonstrates traceability back to the corresponding requirement and that it verifies that the requirement has been met. For example, unit testing should verify that the design specification has been satisfied, system testing should verify that the FS has been satisfied, and user acceptance testing should verify that the URS has been satisfied.

The Validation Plan should also clearly indicate the split of responsibilities between user and supplier. It should be noted that the supplier is normally involved in all levels of testing. The Validation Plan should state that an independent expert (often QA), who will confirm that the testing is being carried out in accordance with the documented test strategy, may witness testing. This is important where the software is developed externally, or where the development is taking place centrally on behalf of several sites.

User Qualification

User qualification activities involve the completion of the Installation Qualification (IQ), Operational Qualification (OQ), and Performance Qualification (PQ) activities.

The hardware and software IQ processes may be reasonably straightforward if the physical system is intended to be located on the same site and only used by that site. However, the introduction of centrally managed systems, shared service operations, and data warehouses as a means of lowering costs and providing a more streamlined infrastructure within an organization may make the overall validation activity for the site more complex.

It is also common to find several different supply chain applications residing on the same host machine. This may impose additional complexity for the overall site validation activity should some of these have a regulatory impact and others not. Where an operating system is shared, this could result in non-GxP-critical applications operating with additional controls in order not to compromise the activities of the GxP-critical application.

Site validation activities should include configuration management, data load, and specification/design/testing of any locally developed specific site modifications. Validation of site-supported infrastructure should be incorporated within site activities.

During the OQ phase, tests should be conducted to verify that any wireless data solution employed in addition to hardwired networks meets the roaming and resilience criteria documented in its corresponding requirements specification. Central testing should be used in support of qualification wherever possible. However, should the OQ have been run centrally, it may be possible for the site OQ activity to be waived, provided that the software has not been locally modified and the central test environment used has the same characteristics as the local site.

PQ is a site-specific activity but may be coordinated across multiple sites if appropriate. Issues raised during PQ should be reviewed by both the central support teams and the local site. In some instances, it may be beneficial for templates for validation documents to be developed by central teams in order to provide a consistent approach across sites.

PQ is heavily dependent on the business producing test cases based on its current or intended SOPs that correspond to the URS and providing the required operational and maintenance controls. It is important that these test cases are up to date and reflect all of the relevant agreed changes that have been implemented since the start of the project. The use of computer systems should be consistent with relevant company policies and relevant regulatory expectations for the business processes that they are intended to support.

Validation Report

To complete the project life cycle, the Project Team should produce a Validation Report that aligns with the site Validation Plan. In addition to confirming site validation activities, Validation Reports should confirm the adequacy of all relevant central activities. A central Validation Summary Report (Quality Report) should be developed reviewing the adequacy and release of each application/product version.

Independent experts may review the results of specific validation activities and the summary report may incorporate their findings and any corrective actions necessary following their review of the original activities.

OPERATION AND MAINTENANCE

User Documentation

SOPs for the use of computer systems should be established by the Business Process owners to provide the correct operational and maintenance controls for the validated system. Following SOPs, plus any supporting detailed localized Work Instructions, should ensure that a state of control is established and maintained at all times while the system is being used to support a regulated process.

These SOPs should clearly define the inputs and expected outputs of the process covered in order to provide objective measurement criteria to determine whether the process is being successfully operated or not. This state of control should then allow the required electronic records to be created and maintained by the system, in addition to the creation of any documentation specified as part of its requirements, as outlined in the traceability requirements diagram shown in Figure 36.2. Any relevant supporting process flows, system overview, or network diagram documentation should also be reviewed and revised as necessary to reflect the new business processes if the requirements and SOPs for the system are updated in the future.

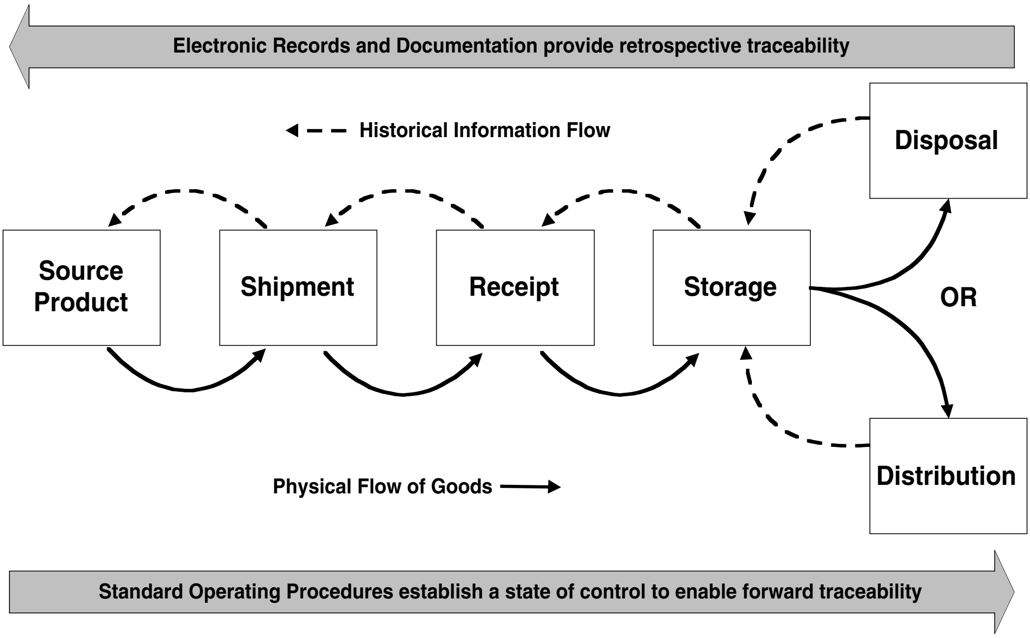

Electronic Records and Documentation provide retrospective traceability
Historical Information Flow
Disposal
Source Product
Shipment
Receipt
Storage
OR
Distribution
Physical Flow of Goods
Standard Operating Procedures establish a state of control to enable forward traceability

FIGURE 36.2 Traceability Requirements for Supply Applications.

Particular care should be taken to ensure that suitable SOPs are in place to monitor finished inventory in line with regulatory expectations so that it retains the purity, identity, efficacy, and safety characteristics required by its licensed specification. If it does not remain within acceptable boundaries of this specification, then it will be considered to be "Out of Specification" and must be immediately allocated a restricted status such as "quarantined," followed by disposal in a safe manner in line with local health and safety and regulatory expectations.

The business should ensure that adequate SOPs are available and routinely tested to ensure that business continuity can be maintained in the event that the system becomes unexpectedly unavailable. These SOPs should also outline alternative procedures that should be followed if the process is expected to continue in the event of failure of key system components such as barcode scanners that may not necessarily render the system inoperable.

In addition to system and technical configuration documentation created as part of the development of changes, it is essential that any user manuals issued to support the use of computer systems in conjunction with the local SOPs are updated whenever required. These manuals should display indication of review and subsequent approval, and should also clearly indicate which revision of the software that they apply to. This is especially important if these user manuals are developed by external software houses.

Training

The training of staff who operate systems is of key importance to the delivery of its expected benefits. Regulatory authorities expect companies to be able to demonstrate that they have evaluated the training needs for their staff.[30] They also expect that all staff have received an appropriate level of training in the specific processes and systems they are expected to follow and operate, respectively, before they commence their work. Staff will also be expected to have the language capabilities to be able to fulfill their roles.[31] In addition to any specialized training necessary to fulfill their roles, general GxP training is required.

Staff training should be recorded in the training records kept for the individual. Training should be delivered by suitably qualified individuals who can demonstrate their current regulatory awareness.[32] If external consultancies are used, evidence should be obtained of their qualifications to deliver this training.[5,6,17] Training is also required to be repeated at regular intervals so that staff knowledge remains up to date.

Warehouses are one area within the supply chain where contractor staff are sometimes used on a regular basis. Contractor staff are often used at short notice to cover for sickness or other unplanned absences, or to provide additional resources at peak busy periods to supplement the existing operators. Another area that may involve the use of contractor staff is the logistics and transportation process that delivers the goods to the licensed wholesaler. In this instance, often the entire operation is subcontracted to a third party.

Both areas involve the handling and processing of finished inventory, with the possibility of contractor warehouse operatives also being involved in the packaging of the goods or assembly of packs. Therefore organizations must ensure that all contractor staff have received satisfactory process and GxP training appropriate for the tasks they are to undertake before they start work.

Organizations are also responsible for ensuring that any contract manufacturers, suppliers, or third-party distributors used are made aware of the requirement to ensure that their staffs receive adequate GxP training. This requirement extends to IT staff who develop the systems used to support the organization's operation, even though they may not physically handle the goods themselves. Regulatory authorities make no distinction between the training requirements for permanent, temporary, part-time, or third-party staff when it comes to training.[32]

Companies should seek to attract and retain knowledgeable staff who will be able to contribute to the further development of supporting systems at appropriate points and also should the system's usage change at some point in the future. Where staff leave the organization, or move on to different roles, it is essential that knowledge transfer to replacement staff is properly effected and recorded in order to maintain a stable operation.

However, managers should be aware that if problems are subsequently encountered regarding the use of the system, and training has been eliminated as the root cause of the problem, then further training in itself is unlikely to solve the problem. Managers should remember that staff may need encouragement to apply the knowledge gained during their training properly.

Change Control

As in any validated application, subsequent changes to marketing and supply applications within a production environment should be controlled and documented to demonstrate that the validated status of the application has not been affected by the change being made. Change control and configuration management processes need to be documented,[33] and should be considered as parallel activities, particularly during the assessment of the impact of the change.

Care should especially be taken regarding changes made to intranet sites where situations could easily arise such as a change in the content of the site without a full impact assessment being carried out on the effect of the change. The control maintained over the intranet site itself as well as its contents are crucial. The data displayed must be up to date, correct, accurate, and maintained in accordance with documented procedures.

There have been numerous instances in recent years of pharmaceutical and healthcare companies operating via the Internet to publish medical information about drug products or devices without realizing the GxP relationship between the internet information and the actual labeling of those products. Drug press releases have been defined as labels by regulatory authorities,[13] and making false and misleading representations can have serious consequences for any pharmaceutical organization.[34] Depending on the extent of the misbranding, this could cause a product to be considered as a drug and legally prevent its marketing without a corresponding approved New Drug Application (NDA).[2]

Regulatory authorities such as the FDA have established routine monitoring and surveillance programs, operated by its Division of Drug Marketing, Advertising, and Communications (DDMAC), and any violations detected will result in Warning Letters being issued without regulatory authorities actually visiting any of the organization's operations.[3]

Care also needs to be taken regarding the consistency between the labeling of the product itself and any promotional statements being made by third parties such as licensed distributors on behalf of the manufacturer. Should these claims cause the product to be misbranded, it is the manufacturer who may receive the actual citation from the inspector, in addition to the distributor.[35]

Where changes are being made by a central development and support group, this group should ensure that changes to the system are communicated to all affected local sites and receive appropriate agreement prior to being acted upon. This ensures that any local implications of the change can be considered prior to its development. The central group is responsible for ensuring that a change is completed in such a way that the validation status of the core application is not affected during the change control process.

Emergency Changes

Unless an emergency situation has developed that would result in a safety hazard or loss of product quality arising due to a system shutdown, sites should not install software in a production environment until all of the relevant validation activities have been completed. If such an emergency situation develops, a documented justification is required to indicate why they considered it necessary to install the software without completing the necessary validation activities. Regulatory authorities expect completion of any outstanding activities as soon as possible after the installation has taken place. A scheduled system change where the implementation plan does not allow enough time for it to be properly executed is generally not considered to be a sound justification for an emergency change. Indeed regulatory authorities will expect to see corrective action being put in place, wherever reasonably practicable, to minimize the likelihood of the risk that the emergency situation will recur in the future.

Security

The integrity of the data held within any system is essential to the successful operation of a regulated business process. This can only be guaranteed if effective security measures have been established to prevent any unwarranted access to the data, whether intentional or not. The generation of electronic copies of master production records without any apparent controls to ensure their authenticity or data integrity will not be overlooked by regulatory authorities.[25]

The security within an application may be a combination of application system security and the standard security features provided by a proprietary operating system. Care should be taken to ensure that the ERES audit trail required by an application is created to track any work undertaken in this area, especially the activities that may be performed by a system or database administrator without using the "front end" Graphical User Interface (GUI) of the system.

A list should be readily available of system users and their access levels to various functions within the application. This list does not have to be available to all users via a formal menu option within the system, but should be easy for a competent System Administrator to generate.

Care should be taken where a system logon is given to a temporary member of staff to use for the duration of employment (e.g., a contractor forklift driver engaged at short notice to cover for unexpected sickness in the warehouse). Regulatory authorities expect to be able to clearly trace who actioned a particular transaction, and the practice of using a generic logon such as "Contractor Temp" used by different individuals on different days should not be encouraged.

Routine IT activities such as the resetting of user passwords should contain steps that require some form of authentication check on the person requesting the password reset.

A key security area that often gets overlooked is the removal of system access or specific privileges from a user's profile for an application when the privileges are no longer necessary. This is often done if an employee leaves an organization, but is often forgotten if someone moves to another job within an organization.

Regular reviews of system privileges should be encouraged to ensure staff members are not granted more privileges than their current roles may actually require. The withdrawal of privileges should also include the removal of the ability to apply electronic signatures where a change in role no longer requires this capability.

However, managers should ensure that any deletion or alteration of user profiles does not affect the transactions held within the system. For example, while the system ID is active, a master file for a particular piece of artwork may show the name and system ID of the person who approved it and thus meet regulatory expectations. However, if this information is sourced from a master file of current systems users, depending on how the design of this link has been established, deleting the user may end up removing the user's detail from the artwork record, thus taking a previously compliant record out of regulatory compliance.

Backups

To prevent the loss of critical GxP data, backups of systems should be taken at regular intervals, as documented in local SOPs. This prevents the physical loss of important system data or its accidental deletion by users. A backup also provides a basis from which the application can be restored in the event of an unforeseen event arising that results in the need to invoke the documented Disaster Recovery Plan. Data that should be subject to backup activities includes system software, such as the operating system and any preconfigured software modules, application software, and configuration parameters. Within a regulatory context, examples of the type of operational data that would be subject to a backup process would be the critical data items identified from the list of data types shown in Appendix 36B of this chapter.

The frequency at which the backup is taken is largely dependent on how frequently the data changes. For example, backups of sales order processing systems that may process thousands of new orders per day are likely to be taken more frequently than backups of system configuration parameters, which are less likely to change on a daily basis. During the process, data is copied onto media external to the system and then stored in a secure off-site location in line with documented SOPs. The backup process should be tested at regular intervals and this testing should be documented.

Disaster Recovery

The support group for the system must ensure that adequate procedural documentation is in place to ensure that system continuity can be restored in the event of a system failure. Business Continuity Plans should be developed in line with relevant Service Level Agreements (SLAs) contracted between the business and the supporting IT organization.

Regulatory authorities expect that the process that will be followed to recover a regulated application will be documented in SOPs and that training will have taken place at regular intervals to ensure that all applicable staff are aware of their responsibilities in relation to the Business Continuity Plan. Evidence should be available to confirm that these processes are regularly tested to ensure that supporting SOPs are adequate to complete the activity, or modified if these tests indicate that events during the test did not quite turn out as planned.

This expectation is exactly the same for any third-party systems used, for example, logistics systems. Organizations can expect regulatory authorities to inquire about whether this topic was covered during the Supplier Audit phase.

Upgrades

All upgrades, regardless of category, must be carried out in accordance with documented change control and configuration management plans. The upgrade of a validated system should be considered as a site- rather than a system-specific activity, as it is likely to involve several different aspects of the IT operation on site. System upgrades can be split into three broad areas:

- Upgrades of the application itself
- Upgrades of the supporting operating system
- Upgrades of the supporting infrastructure or hardware

An upgrade of any part of a previously validated system does not necessarily mean that full revalidation is required. The Validation Plan that addresses the upgrade should incorporate an impact assessment to determine the exact nature of the change, how much of the validated system will be affected by it, and whether it would be within regulatory expectations to undertake a partial validation only. It is recommended that this assessment should take a documented risk-based approach.

If the configuration item being upgraded is an established commercially available operating system (Category 1), then less validation activity is normally necessary. However, upgrades to configurable software (Category 4) or bespoke packages (Category 5) are likely to involve more validation work.[36]

Decommissioning

The decommissioning of any supply chain application is an important regulatory process that often receives less attention than the commissioning of new applications. Decommissioning should be planned well in advance of the deployment of the new solution, and the plan should include a full assessment of the impact of the withdrawal of the system, including the effect of this on any external third-party applications that interface with the application to be retired.

Early planning will allow for full consideration to be given to the best method of preserving any GxP data to allow it to be retrieved at any time in the future within its specified regulatory retention period. This may involve migrating legacy data into any application replacing the retiring application.

Data Archiving and Record Retention

From time to time it may be necessary to remove data or other electronic records from an on-line computer system to another durable secure location for long term off-line storage. This removal also includes the removal of the metadata associated with the record, for example, the properties of a Microsoft Word document or a series of data definition and table relationships.

Such data removal may happen as part of a planned migration process when active records are moved from a legacy system to a new system. It may also occur in systems that have been operational for some time when inactive records are removed from a database in order to improve the performance of the existing system. A record transitions from an active to an inactive phase when its documented aging or status requirements are satisfied and the record is no longer subject to change.

How legacy data should be retained is often a complex issue. The approach taken depends on many different factors, but the most important of these is usually whether or not the method of retention can be maintained throughout the required retention period. It is possible that the technology chosen may become unsupportable during this period, and if this is the case, then an alternative method may have to be chosen either before the original archiving takes place or while the data is in the archive itself. Depending on the circumstance, this may also include archiving to nonelectronic media.

The periodic review of the data contained within the electronic archive often tends to be overlooked. Organizations must ensure that they monitor the records for signs of deterioration over the record retention period, including the media used for storage, if this is being done electronically.

The business owner of a particular system is ultimately accountable for the electronic retention of business data. Support groups are often delegated responsibility for the electronic record retention of technical data such as operating systems, source code, configuration records, and any other software necessary to operate the application itself, including supporting documentation for these components.

Organizations are also responsible for ensuring that any contract manufacturers, suppliers, or third-party distributors used are made aware of the length of time they will be required to retain records of their activities either electronically or on paper, in case these may be required in the future.

There are many other reasons why pharmaceutical and healthcare organizations are required to hold records, not just for regulatory purposes. The record-keeping requirements of local environmental, safety, or financial regulations should also be considered when determining the length of time a record is required to be held. Whatever requirement stipulates the longest retention time is the one that should be adhered to and recorded to determine the disposal criteria.

Where records are retained centrally for use by more than one site, then all of the applicable regulatory, environmental, safety, or financial retention periods for each site need to be considered when determining the record retention period required.

Records are not required to be held indefinitely. When the chosen record retention period has been met, they may be destroyed. The approach to record destruction will vary according to the design of their archive location, but one should consider the maintenance of the data integrity within the archive system. The frequency of record destruction also needs to take into account factors such as operational cost, performance, and ease of disposal.

FUNCTIONALITY ISSUES

PACKAGING AND LABELING

All labels generated by supporting application should be printed and applied so as to remain legible and affixed during the customary conditions of processing, storage, handling, distribution, and, where appropriate, use.[10,12] Labels should not be released for storage or use until they have been inspected for completeness and accuracy by the designated individual.[31]

Any labels generated by a supporting application for finished pharmaceuticals must ensure that the way in which the information is displayed on the label gives prominence to the active ingredients.[37] Where a label for a drug or medical device has been generated by the supporting application, it is still expected to be accounted for in the same way as labels produced by other means.[26]

Computer applications used to support repacking operations, and subsequent relabeling should contain functionality that operates checks to verify that the integrity of the data relating to the batch is maintained. For example, repacked products from the same parent lot still retain the original lot number and individual items do not display multiple expiration dates.[38]

Where the final item being shipped consists of a collection of units, the final pack may often comprise different items that each have different expiry dates and batch numbers. Records that are held within supporting computer applications should accurately reflect the lot number and expiry dates for all of the elements contained within each pack to facilitate traceability. Once the pack has been produced and entered into the system as an item in its own right, the expiry date shown in the system for the pack and on its external packaging should be the earliest component expiry date. This should be verified as a specific test case during OQ and PQ testing.

INVENTORY CYCLE CHECKS

Regulatory authorities may request that a pharmaceutical organization can account for all inventory of a particular item even if a product recall has not been necessary.[37]

Inventory management applications should have functionality and supporting SOPs to enable both planned and unplanned cycle stock checks and other stocktaking activities to be easily undertaken — the latter is especially important as it may form part of any product recall activities. This checking may be required to cover stock at several locations, not all of which may be under the direct control of the organization responsible for managing the recall. Coordination is very important. The system should be configured to rotate stock either a First In–First Out (FIFO) or First Expired–First Out (FEFO) basis.

STORAGE CONDITIONS

Pharmaceutical distributors should ensure that the storage of finished goods meets accepted temperature and humidity conditions for both product quality and regulatory compliance. This requirement applies to all areas within a warehouse where temperature, humidity, and airflow are required to be monitored for specific hot or cold spots within the racking, loading bays, or repackaging areas, areas of entry and exit, and other restricted access areas (e.g., sterile or hazardous areas). Goods in transit should also conform with storage conditions.

Where a computerized system is used to support this activity, the validation of the computer system should include ensuring that any automated monitoring systems are functioning within the specified range.[21] System alarms and other alerts should be adequately tested prior to use to verify that they would be triggered correctly. This testing should be repeated periodically to ensure the validated status of the system is maintained. It may be appropriate to undertake a warehouse mapping exercise to identify climatic variations. Validation activities should also verify that perimeter access systems are adequately tested to ensure that staff who should not be permitted access to restricted areas are actually prevented from gaining this access.

Stock Selection

Developers of regulated systems must ensure that the system incorporates functionality that prohibits the selection of stock with a restricted status to satisfy a sales order. An example of this would be a situation where the testing and release procedures for a particular stock item require appropriate laboratory determination of satisfactory conformance to its final specifications prior to release.[39] This requirement should be stored within the computer system as an attribute of the particular item. The system should be configured in such a way that the stock cannot be released into the sales order processing system until the conformance of the product to this specification has been recorded in the system by a suitably authorized individual. Another example would be where the business operates a process within its supply chain that allows stock with an expired lot date to be received back into a warehouse via a credit note for final disposal. The supporting system must ensure that this stock is allocated a system status such as quarantine that will prevent it being reselected to satisfy another sales order. The system should also ensure that this stock can only be released for disposal under the authority of a Qualified Person. In both circumstances, regulatory authorities would expect to see evidence that this functionality has been adequately tested during validation.

Segregation of Stock

Organizations need to ensure that the functionality within supporting computer applications can be configured to align with the physical characteristics of any warehouse used. The application needs to be able to make clear distinctions between the available storage locations that can be assigned to items with a particular allocated status or storage characteristic. For example, the put-away algorithm logic needs to ensure that only appropriately restricted locations would be displayed as being available for the storage of any item with a "quarantined" status. Any automated selection of storage locations needs to take into consideration the type of items stored in any adjoining locations, following relevant approved industry guidance. For example, it may not be acceptable to store antibiotics next to other items because of risk of contamination. OQ and PQ activities within the validation life cycle would be expected to cover scenarios such as this in detail, in addition to ensuring that only authorized system users were permitted to authorize the removal of items from certain locations using an automated process.

Sample Management

Inventory records need to account for samples withdrawn for stability, expiry, or shelf-life checks. Depending on the allowable tolerances for controlled drugs, such sampling, if not recorded properly, can lead to unacceptable levels of unaccounted stock in the event of a product recall being initiated. Sample request or receipt forms may be transmitted photographically or electronically, for example by facsimile transmission (FAX) or electronic data transfer, provided that the method of transfer meets the security requirements outlined in 21 CFR Part 203.[14] Computer systems used to manage or record the distribution details for samples held by manufacturers' or distributors' representatives also require validation.

Out of Specification (OOS)

There are many aspects of the inventory storage process that can have a direct bearing on the characteristics of a product; for example, the characteristics of a product may change if it is stored at an inappropriate temperature or passes its shelf life/expiry date. Therefore, steps must be put in place to ensure that the application used to control this storage highlights areas of concern before they happen and does not directly or indirectly contribute to the generation of an OOS result.[37] Such steps may include ensuring that the warehouse control system only selects locations with appropriate storage conditions, or ensuring that stock is rotated following an agreed strategy. Functionality should exist within the application to provide information well in advance on those products whose shelf life is approaching the next check cycle or whose expiry dates is approaching. This should be established in accordance with the requirements specified by the business for the particular item and tested during validation. OOS records generated by the computer application must be retained as part of the batch production, packing, or control records[36] and are expected to be investigated.[33]

Inventory Disposal

Where it is necessary for inventory to be disposed of, for whatever reason, this should be done in accordance with any relevant regulatory and local health and safety legislation. This will require certain information pertaining to the disposal of the lot/batches in question to be recorded in the appropriate system in accordance with all legal requirements relating to the type of drug being destroyed. The type of data that is required to be recorded could vary greatly in accordance with the legislation that governs its management. For example, the rules applying to the disposal of controlled drugs are generally far more stringent than those applying to the disposal of OTC drugs regardless of the regulatory body involved. The activity required to validate this functionality should include the record-keeping requirements of all types of drug disposal that it is anticipated the application will be required to support.

Artwork Transfer

Artwork transfer processes should be validated including storage of electronic files on servers. External artwork studios and printers are expected to be subject to a Supplier Audit and periodic review in the same way as any other supplier to the pharmaceutical and healthcare industry. All software, including fonts, used for artwork preparation should be licensed to the organization. An experienced market representative should proofread the hardcopy printout of files (such as PDF) to ensure correct content. Once this has been confirmed, the artwork administrator should check that the printed copy, the electronic version, and the original artwork file are controlled and in alignment before engaging the print supplier. A final check is required[17] by QC to confirm that the packaging components delivered to site correspond with their intended specifications before they can be released for use. Validation needs to consider the regulatory requirements relating to any e-mail or Internet applications used in the artwork transfer process. If the use of these technologies is taking place within a closed environment, the amount of validation activity required could be considerably less than if these technologies were employed within an open environment.[7,11] If an open environment is created by using e-mail to transfer artwork files to design studios, regulatory authorities will expect to see all aspects of this activity covered in the Validation Plan. Even if the final agreed hardcopy is hand signed to confirm approval of the artwork, the system infrastructure used in the artwork process still needs to be validated because it controls the integrity of the artwork that has been approved and demonstrates the validity of the approvals process itself. The final approver is only one member of a group of experts that may have viewed this artwork electronically during its generation and approval process.

Product Recall

Validation needs to be completed for those systems that would support any product recall activities. Regulatory authorities expect that traceability is established throughout the supply chain to facilitate the recall of an entire lot/batch of a product if this becomes necessary to minimize, among other things, any adverse impact on patient safety. Sites may have a locally developed IT solution, a centrally supported IT solution, or may have even subcontracted this activity to a third party as part of an outsourced Sales Order Processing activity.

Outsourcing of Sales Order Processing operations may add additional complexity to the product recall process. Depending on the overall systems solution being employed, the third party may not have direct access to the manufacturing systems, meaning that in some cases a hybrid system may be involved. For example, a signed physical shipment note containing GxP data might travel from the factory with the goods, with its details being input into a third-party system on arrival in the distribution warehouse. In this instance, it is likely that the third-party system would be used to effect any recall and not the internal one.

If a particular lot is the subject of a recall, and the information relating to the lot is not easy to retrieve, then it is possible that a regulatory issue could develop. An example of a more complex situation might be where the vendor lot number has been manipulated in some way and there are no easy means of tracing back to the original vendor lot number. If a formal report has been set up within the application and validated to ensure it that it gives the ability to generate a listing of sales orders by the original vendor lot number, as opposed to any different system allocated lot number, the information required for any recall activity could be easily obtained.

Regulatory authorities do not mandate any one specific process to be followed for product recall, but they expect the process selected by an organization to be adequate for the task[38] and for the recall to be carried out within a reasonable time frame. Accuracy is also of key importance — organizations need to be able to account for all of the items that are subject to a recall, not just those that are nearest. Both the relevant U.S.[17] and EU regulations[4,5] clearly state they expect the chosen process to be fully documented[27] and evidence to be available to indicate that the relevant staff has received training in it prior to it becoming effective.

Contemporaneous Data

Less automated systems that need manual input to confirm that an activity has been completed require data entry to be contemporaneous. If this were not the case, an inspector might walk, for instance, round an available stock area of the warehouse, notice an unattended pallet on a floor in the picking area which is available for picking, and ask someone to show him where this pallet should be. If an enquiry is made into the system that shows that the pallet should be in a separate quarantined area, as outlined in a current SOP, then staff should expect to discuss this further with the inspector.[25,40]

Applications Storing GxP Data in Multiple Languages

Over recent years there has been a distinct trend toward the introduction of global solutions that support the supply chain processes for different countries using the same production instance. This sometimes requires the same information to be stored in more than one language and the operator would be expected to select the language required from within the range of options specified in the application. It is a stated requirement of some regulatory authorities[12] that the translation must convey the meaning properly in order to avoid confusion and dilution. This is particularly important in the case of warning statements. Unless the marketing company has made a prior arrangement with the supplying site for translations to be obtained locally, any foreign language text that is required to be maintained within the supporting application should be supplied by the marketing company.

Database fields that are used to record label information are often free text fields. Care should be taken when validating fields that contain entries in different languages because mistakes may not be apparent to systems testers or operators if they are not familiar with the language concerned. It is often the case that IT development and support staff may only be given the file containing the approved translation from a third party. Care should be taken to ensure that any specific character or accent characteristics required by the new language will be recognized within the new application and will therefore appear exactly as they are intended to on the screen or printed output. Responsibility for the final checking of the text remains with the market, and this requirement should be incorporated into the Validation Plans for the pharmaceutical and healthcare manufacturer's commercial organization.

Presence of Additional Functionality in Externally Developed Software

Where an industry standard modular application is selected as a software solution, it may include functionality within the application that supports different business processes operated by other companies who use the same application. Unless this additional functionality can be disabled within the application, which is not always technically possible, there is a risk that this functionality may be accessed by end users even though the site has not validated this usage. This situation should be controlled by ensuring that all end users are trained in the specific functionality within the application that they are expected to use to support their business processes, and that this training should be supported by SOPs that clearly state that no alternative processes are to be undertaken.

ORGANIZATIONAL ISSUES

Relationships between Local Sites and Central IT Development and Support Groups

Regulatory inspections are not necessarily limited to sites where the application is operated. Some production instances may be centrally managed and supported by a particular IT group. Other applications may only be centrally supported by a particular IT group, while the control of the production instance may be managed locally by the site using the application, or physically managed on behalf of a particular region or user group by another site using the production instance. The scope of an inspection can be extended to cover central IT development and support groups. Central IT groups must therefore ensure that they are ready at all times to face any inspection.

Service Level Agreements for Inspection Support

The local site is responsible for ensuring that the dependencies for inspection support are understood and accurately documented in all appropriate supply agreements or Service Level Agreements (SLAs). Depending on the complexity of the documents, consideration should be given as to whether it might be more appropriate to develop a dedicated SLA for inspection support activities. Where organizational changes are taking place within a company, care should be taken to ensure that continuity of support between the central IT support team and its dependent local sites is planned from the outset and is monitored to ensure that any unacceptable risks are mitigated.

Outsourcing Activities to Third Parties

Outsourcing activities may bring about cost efficiencies and allow pharmaceutical and healthcare organizations to specialize in the core activity in which they believe they have a competitive advantage. However, where outsourcing activities fall under the scope of regulatory scrutiny, this does raise the issue of the computer systems no longer being under the direct control of the pharmaceutical and healthcare organization.

It is essential in any outsourcing operation that there is a written contract covering exactly what is to be supplied and the standards expected for the supplied item(s). These may be complemented by SLAs for hardware, software, and network support. The contract should also state that an adequate Quality Management System (QMS) is expected to be in place to establish the necessary controls to ensure that the software systems meet all of the stated requirements given by the outsourcing organization. Third parties should ensure that they clearly understand exactly what is permitted within the relevant regulations. A satisfactory Supplier Audit is not a substitute for the relevant validation deliverables such as design specifications.

For software development, documents such as URS and FS are the controlling specifications for the items to be delivered, and as such form part of the contractual document set for the software. These should be signed by all parties before any subsequent development work is undertaken, and changes in requirements should follow a standard contract variation process and should not be agreed informally between staff from both parties. Organizations should also consider whether it would be beneficial to include the type of documentation that would be required in an inspection situation as part of any escrow agreement that is established. In the event that the third party unexpectedly ceases to trade, the pharmaceutical or healthcare organization would then have access to the relevant documentation to back up its systems development.

TECHNOLOGY ISSUES

USE OF BARCODES

Applications interfacing with electronic reading components such as barcode or microchip readers must be validated fully before they are used in the same way as any other application. The barcode for an item may be incorporated into the artwork of the packaging for the item in a standardized format (e.g., ISO/EAN). Validation should include the use of barcode reading components.

In most instances it is not practical for the actual lot number or lot expiry dates to be captured in the artwork itself. If the warehouse management system chosen is barcode driven, consideration may be given to producing a further barcode to record this information and affixing this to the items from the relevant batch. However, if this is to take place, care should be taken to select a location for the new label that does not obstruct or deface key information on the packaging, and to ensure that the risk of the new barcode becoming detached from the packaging is eliminated. The font used to print out the bar code should be carefully selected. Bar code fonts have been known to create EAN/UPC symbols with serious design defects. The design of the font, an operator input, or a combination of both may sometimes cause problems. In addition, most fonts do not automatically calculate and add the check digits and other security features to bar codes expected by regulators. A separate application is usually needed to calculate the check digits first so that they are available to be added to the bar code created. Even if a barcode system is used within the supply chain to generate the lot number and lot expiry for each SKU, and thereby increase efficiency, both types of information still need to be present in a format visible to the human eye for the end user. This is usually achieved by punching or embossing this information on one or more of the primary, secondary, or tertiary packaging solutions used.

In addition to the placement of bar codes on product packaging and labels, some pharmaceutical manufacturers are now considering printing barcodes on individual drugs. The intention is to ensure that medicines used in hospitals are compatible with computerized systems used to support day-to-day operations to ensure that a patient gets the right medicine in the right strength at the right time. Data types that could be encoded onto the product to support this are the medicine type, its dosage, lot number, and expiration date. In addition to reducing human errors, the codes would simplify recalls, investigations of adverse events, and the purging of expired medicines from inventory.

INTERNET APPLICATIONS

The increased use of the Internet in recent years as a means of communicating information on drugs and medical devices has provided considerable benefits to the healthcare industry. The Internet offers many possibilities in terms of graphic representation of data and the ability to publish information to a wider audience at a faster speed than by the use of traditional marketing channels. The overriding compliance consideration, however, is the accurate transmission of information to the reader, whether the reader is a member of the public or a pharmaceutical or healthcare professional.

Web site developers should consider the possibility that features visible with some Internet browsers may not always be visible if another browser is used. This can be difficult to fully scope during testing because it is unrealistic to expect testing to encompass all of the Web browsers that are known to be available within the marketplace. It is suggested that the most practicable approach that could be taken would be to:

- Ensure that the Web site and its development environment remain under configuration control at all times and changes to its content should be made in accordance with a documented change control plan appropriate to the technology in use.
- Establish and document the standards to be used for all Web site development following the same approach used for other coding standards within the software development life cycle.
- Ensure that these standards outline the use of standard HTML/XML with no use of browser-specific extensions.
- Ensure that standards specify minimal use of browser scripts (if at all) and plugins and should prohibit the use of any platform specific plugins.
- Incorporate any relevant considerations such as requirement for accessibility for users with visual handicaps and text-only browsers into the layout and design of the Web page.
- Consideration should be given to clearly marking the information with the date/time on which it is valid, to distinguish it from the date/time it could be printed out by readers.
- Verify during the Design Review phase that the code produced contains valid logic. Appropriate tools such as HTML checkers, etc. may be used to assist in this process.
- Ensure that the test cases used for OQ and PQ should test a reasonable range of browsers currently available to determine that the information is displayed in a consistent format across this range.

This approach should ensure that the design of the Web site avoids creating warnings, hazards, or other pertinent facts within medical information statements. If a different browser is being used, that does not support the original technology used, then important medical information may not be brought to the reader's attention.

As Internet information may be held in the "cached" memory on servers other than those of the originating organization (e.g., some public search engines), consideration should be given to clearly marking the information with the date/time on which it is was published. This is needed to distinguish it from the date/time it could be printed out by readers, and should be incorporated into the OQ and PQ testing undertaken before the Web site is released or updated.

Although not a regulatory issue, organizations intending to use the Internet as a method of communication should also consider taking control of the domain names they use and publicizing them as the official communication channels for the organization. Internet users may find it difficult to distinguish between information posted by an organization on its Web site(s) and other opinions stated by other organizations on different Web sites, especially if the Web site address on which the information has been placed is similar to the official Web site address. Available domain names can be legitimately registered that are extremely close to an official Web site address for the manufacturing or marketing organization. For example, two completely different organizations could register the same Web site, one using the ".co.uk" suffix and the other using the ".com" suffix.

USE OF SPREADSHEETS

The use of spreadsheets within the pharmaceutical and healthcare industries is one area that is coming under increasingly greater regulatory scrutiny. Spreadsheets need to undergo an assessment exercise similar to other computer systems during their requirements phase to determine if their usage needs to be validated before they can be used. Authors and users should have different access profiles to prevent accidental overtyping of data. Spreadsheets that have a regulatory impact should be stored on servers with managed access control. This may be achieved using the "password to modify" option or by establishing NT access control functionality. Consideration should also be given to the effect of upgrading the spreadsheet package on the validation of the spreadsheet. An upgrade of functionality could affect any existing prerecorded macros present within the spreadsheet file; for example, any calculations may not give exactly the same result that they did previously. Validation should take place to confirm that the upgrade has not introduced any undesirable features.

Radio Frequency Identification (RFID)

RFID is a technology that looks to have the potential to make a major impact on the Pharmaceutical Supply Chain over the coming years in the fields of product and resource identification and tracking. RFID is creating innovative new business opportunities by making everyday objects and products intelligent and interactive. RFID is a unique technology that enables data to be transmitted from a micro silicon chip at very fast speeds and without the need for line of sight as is currently required by barcodes. RFID has been at work for several years in systems where fresh food products are tracked through the supply chain. RFID is robust and will survive in harsh environments where a barcode would normally be destroyed. As the technology has significantly decreased in size and cost since its introduction, it is now becoming cost-effective to place an RFID tag, consisting of a chip and antenna, on virtually any object and allow that object to be identified uniquely and tracked accurately.

Awareness of Technology Limitations

When selecting any application that is intended to be used to support a regulated process, organizations should ensure that they are fully aware of any limitations within the technology methods they are considering and whether these limitations may make it difficult to successfully validate the use of the chosen solution. Limitations may include but are not limited to:

- Spreadsheet packages that have a limit on the number of characters that can be contained within one cell.
- Databases that have limits on the total number of records the database can contain, or on the maximum size of particular field types.
- Graphics or drawings packages that may have a limit to the number of pages that can be created in a particular file.
- E-mail applications that may truncate any attachments.

In some instances it may not be apparent to end users that the particular limitation is reached; for example, the sender or recipient may have no indication that the attachment to the e-mail sent has been truncated. In other instances, existing information at the beginning of a file may be overtyped by information entered once the data limit for the file has been reached.

If there are any known limitations for any chosen application and the decision to proceed with the use of the application to support a validated process is made by the organization, then documented warnings should be given to staff so that they are aware of the situations they should avoid in the course of their work.

VALIDATION ISSUES

Awareness of Relevant Regulatory Expectations

In addition to ensuring that staff receives adequate and frequent training that meets the relevant regulatory expectations in full, it is important that they also receive training in the inspection process itself so they are aware of how this process is likely to be carried out.

This should ensure that potentially awkward situations do not arise where they are reluctant to provide a regulatory authority with information that might reasonably expect to be discussed during an inspection.[33] A refusal to provide relevant information to the FDA for instance may contribute toward a Warning Letter being issued to an organization that might otherwise have been avoided.[41] Such refusals are regarded as serious violations because they are deemed by regulatory authorities to hinder an inspector's ability to thoroughly and completely evaluate an organization's ability to make safe and effective drug products[23] and medical devices.

Change in Regulatory Status

Even if a system has previously been assessed as having no regulatory impact, it does not follow that this assessment will remain correct for the lifetime of the system. A purchasing system, for example, that has previously only been used to purchase non-GxP items but subsequently used to purchase GxP items will now require validation even though the functionality remains exactly the same as documented in the previous assessment. All central groups developing and supporting a system are responsible for notifying their local user site if there are any planned changes in the use of the computer systems. The local site will then need to ensure that an impact assessment is undertaken to determine the effect of the change and whether its implementation means that a full or partial revalidation exercise is necessary.

Human Error

Where a system is not fully automated and requires transactions to be undertaken and subsequently confirmed by users, opportunities for human error arise. Human error in artwork is typically a major problem. A seemingly small error, often with decimal points, similar product names, or minor inconsistencies in dosage information can have far-reaching consequences for the organization concerned because it could jeopardize patient safety.[37] Another accidental error could result in inventory being put in the wrong physical warehouse location. While it could be easy to locate the misplaced items if the space is adjacent to the one that should have been used, the incorporation of functionality such as the requirement to input check digits or scan a unique barcode for each location is encouraged. This can then be used to automatically update the system to confirm that the put-away activity has been successful. A further common source of errors occurs when high bay racking locations are out of reach for the operator and as a result they are required to input the check digits or scan a barcode for the location that has been placed in a more accessible location as a substitute. Fully automated systems rather than hybrid ones are therefore recommended.

Reliance on Suppliers

Contracts placed with the supplier should include provision for ensuring regulatory compliance. This is no different from the expectations for suppliers satisfying other types of supply criteria within the pharmaceutical and healthcare supply chain,[30,32] and regulatory authorities may request documented evidence to support the supplier selection process.

Regulatory authorities will expect to see evidence that the software supplier has been audited against appropriate documented assessment criteria by suitably qualified individuals before the products are used in the production environment. This assessment should be repeated at regular intervals after the initial assessment to determine their continued suitability to supply and support the software. Regulatory authorities will not have been party to the confidentiality agreements between the pharmaceutical or healthcare company and the supplier being audited. Consequently, without due cause they will not expect to see the detailed findings of the audit itself. Instead, regulatory authorities expect to see evidence that audits have taken place, including evidence of the management of any follow-up activities.

Contracts to supply software are expected to list the documentation to be provided as part of the contracted deliverables. With an externally sourced software product, these generally include documentation such as installation instructions, user manuals, system administrator manuals, configuration manuals, data definition guides, and training materials, all of which are key documents for the validation process. Other documentation is likely to be proprietary and confidential to the software supplier for intellectual property reasons. Where this is the case, it is a regulatory expectation that the organization has undertaken a Supplier Audit to assure that the software product complies with regulatory expectations and is fit for its intended purpose.

Central development and support teams should ensure that the quality of suppliers supporting central activities have been assessed. Centrally organized supplier audits should be conducted in conjunction with regulatory groups that are familiar with the organization's operating model. Suppliers supporting local modifications should also be subject to supplier assessment.

Effect of Subsequent Local Modifications to Globally Released Applications

Applications that have been globally released to different sites and then legitimately modified locally to meet specific reporting requirements for the local site only may require more validation than just the change itself. The receiving site may no longer be able to wholly rely on any development documentation provided by the central support group. This is because it may no longer correspond to its actual business use where the local modification has amended the functionality present in the software. Local sites should ensure that any documentation generated for the new modification adequately addresses this gap.

Process Improvement

For medical devices, inspectors expect to see a process improvement program in place encompassing all activities within the supply chain, including the systems that support it. Activities that would provide evidence that such a program is in place include the implementation of documented processes such as Corrective and Preventative Action (CAPA),[41] self-inspection, and internal audits.[42] Organizations should ensure that where an SOP has been put in place to initiate a process improvement activity, the SOP is adhered to. For example, if a quality audit SOP states that all areas shall be subject to audit at least once every 2 years, then inspectors will expect to see evidence that this has been the case. Failure to do so is likely to result in a citation similar to that issued to Krieger Medical, Inc.[24]

Within any software development life cycle, the effectiveness of the change management process and the corresponding configuration management process are often indicators of whether or not any improvements need to be made. One opportunity for continuous improvement that is often overlooked in the software development life cycle is the post-project review process, which concentrates on any negative business and technical issues that have arisen as a result of the deployment of the new software system. Project teams should also be encouraged to invite members from other teams to take part in certain review processes to encourage two-way knowledge sharing on what they have both learned from the execution of their respective projects. This could shorten delivery times for later projects and help teams avoid earlier pitfalls by learning about some of the activities and methods applied by other projects.The effectiveness of the process improvement program depends to a certain extent on the culture in which the information is being exchanged. However, if this process is not managed constructively, it could be easy for this to develop from an "improvement culture" into a "blaming culture."

Use of Electronic Records and Electronic Signatures

The introduction of paperless systems has meant that specific consideration has to be given to the way that ERES are created and subsequently managed. The method and justification for preserving data on electronic media should be documented in sufficient detail and communicated to all relevant staff. This is important because there is generally no paper documentation to back up these transactions should insufficient electronic data be recorded or satisfactory data subsequently become corrupted. The electronic records created to support all transactions that fall within the scope of the various pharmaceutical regulatory authorities should generate satisfactory audit trails within the application that record the creation, amendment, and deletion of these transactions. This is particularly important in the case of record deletions because the electronic record itself will no longer be present to support the transaction.

The approval process for artwork is one area where the growth in electronic communications technology could deliver significant reductions in lead time to market for key information. Electronic transfers between the organization and external parties such as design consultancies and graphic studios means that the design of these processes needs special care to ensure that data cannot be amended in any way while it is being transferred between the parties involved.

IT staff should ensure that the boundaries between open and closed network components are clearly defined and there are methods specified for protecting data, such as access controls, firewalls, and cryptographic techniques. The validation of open systems especially needs to be carefully planned to ensure that the interests of all parties are protected.

Prospective vs. Retrospective Validation

Wherever possible, validation should always be carried out prospectively before the application is used for the first time in a GxP context. It is acknowledged, however, that this is easier to achieve with a new business process/facility. If the warehouse facility or system, for instance, has originally been used for non-GxP purposes and is then used for GxP purposes, retrospective validation may be unavoidable and will need careful planning in order to be successful. Regulatory authorities often raise concerns regarding retrospective validation decisions, unless a sound justification for the decision can be demonstrated. Even if computer system validation has been attempted retrospectively, there is no guarantee that any retrospective validation is going to be meaningful and therefore deliver a satisfactory outcome. There must be sufficient system documentation in place to demonstrate that the system has been developed against a formal Quality Management System (QMS) that takes into account regulatory expectations and good software development practices.

Global Impact of Regulatory Deficiencies (Global Commitments)

The ability to correct reported defects across an organization is an area that is coming under increasing regulatory focus with organizations being expected to provide a written commitment to regulatory authorities that they will prospectively correct similar potential or actual defects across other sites in their network. If the same known defect is found to be present in the same computer system on different operating sites, or on different computer systems supporting other operations on the same site, regulatory authorities could legitimately interpret this as a corporate pattern of bad practice.[43] If the situation only applied to one circumstance on one site, a less severe censure such as an observation might be raised against the organization. Should the situation be found to apply to multiple circumstances, the organization might find itself in a situation where the inspector considers that there is no other appropriate option but to escalate the issue (e.g., FDA Warning Letter).

CONCLUSION

The validation activities required for any marketing or supply application, whether central or local, must have active and visible support from senior management to succeed. This sends a visible signal to staff that satisfactory validation of the software is a management concern and priority and that it should bring tangible benefits to the organization in the long term.

Organizations should not wait for an inspection to detect or correct any issues arising regarding the validation of their marketing and supply applications. Regular monitoring of validation activities as they are taking place should be used as a basis to determine the effectiveness of the Validation Master Plan. Any system-related deviations or incidents should be investigated and used as part of the continuous improvement process that is a customary part of the standard operating procedures that should be in place to support the software development life cycle within all pharmaceutical and healthcare organizations.

ACKNOWLEDGMENTS

The author would like to thank Guy Wingate (Director, Global Computer Validation, GlaxoSmithKline), Don Valentine (Principal Consultant — Pharmaceutical Practice, Industry, Distribution and Transport Business Unit, LogicaCMG), Mark Lawson (Quality Support Scientist, Boots Healthcare International), and Paul Hetherington (Director, Emcee Squared Limited) for their contributory discussions and review comments on this work.

REFERENCES

1. FDA (2002), Warning Letter issued to Vita-Erb Ltd, June 7, by the Kansas City District Office of the FDA.
2. FDA (2002), Warning Letter issued to Earth & Plant, Inc., August 13, by the Seattle District Office of the FDA.
3. FDA (2001), Warning Letter issued to Pharmacia Corporation, February 1, by the Division of Drug Marketing, Advertising and Communication, Rockville, MD.
4. EU Directive 2001/83/EC on the Community code relating to medicinal products for human use, Official Journal of the European Communities, L311/67.
5. PIC/S (2000), Guide to Good Manufacturing Practice for Medicinal Products, Pharmaceutical Inspection Co-operation Scheme (PIC/S), Document PH 1/97 (Rev.), December.
6. Good Distribution Practice for Medicinal Products for Human Use 94/C 63/03, Official Journal of the European Communities L113, April 30, 1992.
7. EU Directive 1999/93/EC on the community framework for electronic signatures, Official Journal of the European Communities, L13/12, December 13, 1999.
8. United Kingdom Medical Devices Regulations (1994), Regulation 2(1): S1 1994 No. 3017 (this U.K. regulation implements EU Directive 93/42/EEC).
9. European Directive 93/42/EEC concerning medical devices — class 1lb Parts III and V.
10. MCA (2002), Best Practice Guidance on the Labeling and Packaging of Medicines, United Kingdom Medicines Control Agency, December 12, Her Majesty's Stationery Office, London.
11. 21 CFR Part 11 — *Electronic Records: Electronic Signatures*, U.S. Food and Drug Administration, Department of Health and Human Services, Rockville, MD.
12. 21 CFR Part 201 — Labeling, U.S. Food and Drug Administration, Department of Health and Human Services, Rockville, MD.
13. 21 CFR Part 202 — Prescription Drug Labeling, U.S. Food and Drug Administration, Department of Health and Human Services, Rockville, MD.
14. 21 CFR Part 203 — Prescription Drug Marketing, U.S. Food and Drug Administration, Department of Health and Human Services, Rockville, MD.
15. 21 CFR Part 205 — Guidelines for State Licensing of Wholesale Prescription Drug Distributors, U.S. Food and Drug Administration, Department of Health and Human Services, Rockville, MD.
16. 21 CFR Part 210 — Current Good Manufacturing Practice in Manufacturing, Processing, Packing for Holding of Drugs; General, U.S. Food and Drug Administration, Department of Health and Human Services, Rockville, MD.
17. 21 CFR Part 211 — Current Good Manufacturing Practice for Finished Pharmaceuticals, U.S. Food and Drug Administration, Department of Health and Human Services, Rockville, MD.

18. 21 CFR Part 820 — Quality System Regulation, Medical Devices, U.S. Food and Drug Administration, Department of Health and Human Services, Rockville, MD.
19. FDA (2001), Warning Letter issued to Farouk Systems Inc., August 1, by the Dallas District Office of the FDA.
20. FDA (2002), Warning Letter issued to Borschow Hospital & Medical Supplies, Inc., February 21, by the San Juan District Office of the FDA.
21. FDA (2003), Warning Letter issued to Pharmaceutical Distribution Systems, January 3, by the Baltimore District Office of the FDA.
22. FDA (2002), Warning Letter issued to the Alero Corporation, January 16, by the San Juan District Office of the FDA.

22. FDA (2002), Warning Letter issued to the Alero Corporation, January 16, by the San Juan District Office of the FDA.

23. FDA (2003), Warning Letter issued to Eon Labs Inc., February 6, by the New York District Office of the FDA.
24. FDA (2003), Warning Letter issued to Krieger Medical, Inc., January 10, by the New England District Office of the FDA.
25. FDA (2001), Warning Letter issued to Cardinal Enterprises, Inc., December 7, by the New England District Office of the FDA.
26. FDA (2002), Warning Letter issued to the Opti-Med Controlled Release Labs, Inc., January 9, by the Detroit District Office of the FDA.
27. FDA (2002), Warning Letter issued to Imperial Drug & Spice Corp., January 16, by the New Jersey District Office of the FDA.
28. FDA (2003), Warning Letter issued to Icon Laboratories, 4 February 2003, by the Florida District Office of the FDA.
29. FDA (2001), Warning Letter issued to Purdue Pharma Inc., November 9, by the New Jersey District Office of the FDA.
30. FDA (2002), Warning Letter issued to Hearing Aid Express, February 22, by the Dallas District Office of the FDA.
31. FDA (2001), Warning Letter issued to Trusted Care, December 14, by the New England District Office of the FDA.
32. FDA (1999), Warning Letter issued to PharmaScience Laboratories, September 21, by the New Orleans District Office of the FDA.
33. FDA (2002), Warning Letter issued to West Agro Inc., March 28, by the Kansas City District Office of the FDA.
34. FDA (2001), Warning Letter issued to Biogen Inc., March 29, by the FDA Center for Biologics Evaluation and Research, Rockville, MD.
35. FDA (2002), Warning Letter issued to CASA Lab, Inc. (d.b.a. Walking Bird International, Inc.) January 2, by the New Orleans District Office of the FDA.
36. GAMP — Supplier Guide for Validation of Automated Systems in Pharmaceutical Manufacture, Version 4, December 2001 (International Society of Pharmaceutical Engineers — ISPE).
37. FDA (2002), Warning Letter issued to Tom's of Maine, Inc., November 14, by the New England District Office of the FDA.
38. FDA (2002), Warning Letter issued to TYA Pharmaceuticals, August 6, by the Florida District Office of the FDA.
39. FDA (2003), Warning Letter issued to the PureTek Corporation, February 10, by the Los Angeles District Office of the FDA.
40. FDA (2002), Warning Letter issued to Hobart Laboratories, December 6, by the Chicago District Office of the FDA.
41. FDA (2002), Warning Letter issued to SOUNTEC Inc., November 14, by the Dallas District Office of the FDA.
42. FDA (2002), Warning Letter issued to Minnesota Extrusion Inc., March 29, by the Minneapolis District Office of the FDA.
43. The United States of America v. Schering-Plough Corporation and Schering-Plough Products, LLC (corporations) and Richard J. Kogan and Steven C. Chellevold (individuals) filed in United States District Court, District of New Jersey on May 20, 2002.

APPENDIX 36A
POSSIBLE GXP BUSINESS PROCESSES SUPPORTED BY MARKETING AND SUPPLY APPLICATIONS

This appendix lists possible marketing and supply business processes that may be supported by an application falling under relevant U.S. or EU regulatory scrutiny. This list is indicative only, and depending on the functionality offered by the chosen application, further regulatory conditions could also apply.

Business Process	GxP Relevance
Artwork	Organizations are responsible for ensuring that the text and graphic details of all printed packaging component artwork for all products and devices are accurate, complete, and compliant with all known local requirements. Critical items of information should be located together on the artwork and appear in the same field of view where practicable. Where practicable, artwork for packs should include space for the placement of a dispensing label. Checks should be carried out to ensure that hardcopy, electronic copy, and original artwork file are all in alignment before engaging the print supplier.
Packaging	Packaging materials should be representatively sampled upon receipt and again before use. Records should be maintained for each shipment received of each different packaging material indicating receipt, examination, or testing, and whether the packaging has been accepted or rejected.
Labeling	All relevant information must be presented in a legible manner that is easily understood by all those involved in the supply and use of the drug or device. Labels should be printed and applied so as to remain legible and affixed during the customary conditions of processing, storage, handling, distribution, and, where appropriate, use. No person other than the manufacturer, packaging organization, or distributor should be identified on the label of the drug, drug product, or medical device.
Medical Information	Marketing information should correspond with the terms of the license for the product. The information should not be false, lacking in fair balance, or otherwise misleading. Only positive statements should appear on labeling to avoid ambiguity of the message, for example, "For intravenous use only." Negative statements such as "Not for intravenous use" should not be used. Information on a particular product or device should be consistent across all marketing channels. Marketing information for prescription drugs should detail all facts pertinent to the use of the drug, including a true statement of information in brief summary relating to the side effects, contraindications, and effectiveness of the drug.
Installation Instructions (Medical Devices Only)	Each manufacturer of a device requiring installation should distribute the instructions and the procedures for this activity with the device or otherwise make them available to the person(s) installing the device.
Supplier Order Placement	Orders should only be placed with suppliers who have been assessed and subsequently authorized to supply a product meeting documented acceptance criteria.
Received Goods	Goods should be checked upon arrival to ensure the consignment corresponds to the order and that the goods have been checked for damage.
Materials Handling	Prevent contamination or mix-ups during the course of receipt, identification, storage, packaging, labeling, and quarantine operations.
Storage	Monitor and record environmental conditions such as temperature, humidity, and air quality in accordance with predefined standards and procedures.
Stability Testing (Out of Specification)	Assess the stability characteristics of drug products that should be established and followed to determine appropriate storage conditions and expiration dates. This should include the periodic retesting of finished products during their storage. Similar written procedures should also exist to outline how the expiry conditions and dates for medical devices should be monitored.

Business Process	GxP Relevance
Release of Product	Where the release of batches for sale or supply is carried out using a computer system, the system should only allow a Qualified Person to release the batches, and it should clearly identify and record the details of the person who released particular batches.
Status of Inventory	Goods should be assigned an accurate status that indicates their position within the supply chain at any particular point in time. Examples could include "Received," "On Hold," "Quarantined," "Available," "Planned," "Picked," "Dispatched," "In Transit," "Delivered," "Rejected," "Referred," and "Awaiting Life Extension." Quarantined Status may apply to goods shipped without Certificate of Analysis, returned products, damaged products, incomplete products, counterfeit products, expired products, misbranded and adulterated products, or goods from an unauthorized Supplier.
Inventory Reconciliation	Losses, errors, and inventory reduction following destructive testing should be reported and recorded. Stockholding inventories should be adjusted to reflect these activities.
Sales Order Processing	A Sales Order should only be raised in favour of persons who are authorized to hold and distribute the finished inventory or medical device.
Stock Rotation	A process should be in place to ensure that the oldest stock should be distributed first — First In–First Out (FIFO) or First Expired–First Out (FEFO).
Dispatch	Finished inventory should be checked by a trained individual for identity, damage, and to ensure they have been held under the correct storage conditions prior to distribution.
Distribution	The market supply planning undertaken should ensure continuity of supply in the event of an unexpected emergency situation occurring. For finished pharmaceuticals, distribution records should contain the name, strength of the product, description of the dosage form, name and address of consignee, date and quantity shipped, and lot or control number of the drug product. For medical devices, distribution records should contain the name and address of the initial consignee, the identification and quantity shipped the date of shipment, and any control numbers used. Where a device's fitness for use or quality deteriorates over time, procedures should exist that ensure that expired devices or devices that have deteriorated beyond acceptable fitness for use are not distributed.
Record Retrieval	Records that can be immediately retrieved by computer or other electronic means should be readily available for inspection during their retention period. Records kept at a central location apart from the inspection site and not electronically retrievable shall be made available for inspection within 2 working days of a request by an authorized official.
Complaints	Procedures describing the handling of all written and oral complaints regarding a drug product should be established and followed. A written record of each complaint should be maintained in a file designated for drug product or medical device complaints.
Product Recalls	A documented process should be established to specify how product recall activity can be readily undertaken should this become necessary.
Product Returns	Returned products are expected to be labeled as such and segregated from other stock to prevent reuse. Returned products should be destroyed unless examination, testing, or other investigations prove that the returned drug product still meets appropriate standards of safety, identity, strength, quality, or purity.
Product Salvaging	Drug products and medical devices that have been subjected to improper storage conditions, or where the history of their storage cannot be verified, should not be salvaged and returned to the marketplace.
Product Disposal	The destruction of defective or date-expired products should be carried out in accordance with written procedures.

APPENDIX 36B
EXAMPLES OF REGULATORY DATA TYPES WITHIN MARKETING AND SUPPLY BUSINESS PROCESSES

The table below provides examples of typical data types that may have a regulatory impact that can be found within various regulated marketing and supply applications. It should be noted that many of these GxP data types are common to more than one area of the marketing and supply process.

	Functional Area					
Process Area/Data Type	**Source Product**	**Receipt**	**Storage**	**Disposal**	**Distribution**	**Marketing Information**
User Control						
Security Access	X	X	X	X	X	X
Name & Password	X	X	X	X	X	X
Medical Information						
Product/Established Name						X
Generic/Proprietary Name						X
Dosage Form						X
Storage Conditions						X
Sterility						X
Pharmacological/Therapeutic Class						X
Warning/Hazard Statements						X
Artwork Details						
Storage Conditions	X	X	X	X	X	X
Product/Established Name	X					X
Generic/Proprietary Name	X					X
Active Ingredient	X					X
Dosage	X					X
Quantity/Pack Contents	X					X
Tamper Evidence Statement[a]	X					X
Product License Details	X					X
Registration Number	X					X
Barcode	X					X
Contact Information	X					X
Packaging						
Packaging Type	X	X	X			
Item/Part Reference Number	X	X	X			
Item Description	X	X	X			
Date of Receipt	X	X				
Quantity Received	X	X	X			
Supplier Name	X	X				
Supplier Address	X	X				
Examination/Testing Data		X	X			
Acceptance or Rejection Decision		X	X			
Labeling						
Product/Established Name					X	
Generic/Proprietary Name					X	
Dosage/Quantity/Pack Contents					X	
Potency					X	
Lot/Batch Number					X	

	Functional Area					
Process Area/Data Type	**Source Product**	**Receipt**	**Storage**	**Disposal**	**Distribution**	**Marketing Information**
Labeling (Continued)						
Control Number					X	
Date of Expiry					X	
Handling Conditions	X	X	X	X	X	
Storage Conditions	X	X	X	X	X	
Installation Instructions (Medical Devices Only)					X	
Name of the Manufacturer					X	
Manufacturer Place of Business					X	
Name of the Packing Company					X	
Packing Company Place of Business					X	
Name of the Distributor					X	
Place of Business of the Distributor					X	
Inventory of Samples						
Product/Established Name	X	X	X	X	X	
Generic/Proprietary Name	X	X	X	X	X	
Potency	X	X	X	X	X	
Number of Samples Received	X	X	X			
Name of Sample Recipient					X	
Address of Sample Recipient					X	
Date of Sample Distribution					X	
Number of Sample Units Shipped					X	
Date of Sample Disposal				X		
Number of Sample Units Disposed				X		
Supplier Details						
Name of Supplier	X	X	X	X		
Supplier Address	X	X				
Address Goods Shipped from	X	X				
Address Goods Shipped to	X	X				
Purchase Order Number	X	X				
Supplier Batch Number	X	X	X	X	X	
Supplier Control Number	X	X	X	X	X	
Quality Approval	X	X				
Lot/Batch Information						
Lot/Batch Number		X	X	X	X	
Lot/Batch Status		X	X	X	X	
Control Number		X	X	X	X	
Date of Expiry		X	X	X	X	
Date of Receipt		X	X			
Quantity		X	X	X	X	
Potency		X	X			
Conversion Factors		X	X			
Batch Notes[a]		X	X	X	X	
Item						
Item Number	X	X	X	X	X	
Item Description	X	X	X	X	X	
Item Notes[a]	X	X	X	X	X	

	Functional Area					
Process Area/Data Type	**Source Product**	**Receipt**	**Storage**	**Disposal**	**Distribution**	**Marketing Information**
Item (Continued)						
Location			X	X	X	
Type	X	X	X			
Quality	X	X	X			
Shelf Life	X	X	X	X		
Retest Days		X	X			
Barcode		X	X	X	X	
Purchase Order						
Purchase Order Number	X	X				
Supplier	X	X				
Purchase Order Date	X	X				
Purchase Order Quantity	X	X				
Date of Receipt		X				
Quantity Received		X				
Unit of Measure	X	X				
Supplier Batch Number		X	X	X	X	
Bill of Materials						
Item Number		X				
Item Description		X				
Quantity		X				
Units of Measure		X				
Conversion Factors		X				
Work Centers		X				
Potency		X				
Yield Factors		X				
Critical Process Parameters		X				
Approval		X				
Process Order						
Process Order Number		X				
Quantities		X				
Receipt Date		X				
Transaction		X				
Inventory Receipt						
Supplier Name		X	X			
Purchase Order Number		X				
Purchase Order Quantity		X				
Quantity Received		X	X			
Quantity Outstanding		X				
Units of Measure		X	X			
Conversion Factors		X	X			
Carrier Name		X				
Customer Orders						
Customer Order Number					X	
Customer Name					X	
Customer Address					X	
Quantity Ordered					X	
Quantity Supplied					X	
Address Goods Shipped from					X	

	Functional Area					
Process Area/Data Type	Source Product	Receipt	Storage	Disposal	Distribution	Marketing Information
Customer Orders (Continued)						
Address Goods Shipped to					X	
Item Number					X	
Item Description					X	
Item Notes[a]					X	
Lot/Batch Number					X	
Control Number					X	
Distributor Details						
Distributor Name					X	
Distributor Code Number					X	
Distributor Address					X	
Address Goods Shipped from					X	
Address Goods Shipped to					X	
Date Collected					X	
Quantity Collected					X	
Item Number					X	
Item Description					X	
Item Notes[a]					X	
Lot/Batch Number					X	
Control Number					X	
Recipient Details						
Customer Order Number					X	
Customer Name					X	
Customer Address					X	
Shipping Address					X	
Shipping Notes[a]					X	
Date of Dispatch					X	
Date of Receipt					X	
Returned Goods Shipment						
Customer Order Number		X			X	
Return Goods Note Number		X			X	
Customer Name					X	
Customer Address					X	
Address Collected from		X			X	
Address Returned to		X			X	
Shipping Notes[a]		X	X		X	
Date of Return		X	X		X	
Date of Receipt		X	X		X	
Quantity Returned		X	X		X	
Lot/Batch Number		X	X		X	
Control Number		X	X		X	
Date Quarantined		X	X		X	
Reason for Return		X	X		X	
Stock Adjustment						
Number			X	X		
Item Description			X	X		
Item Notes[a]			X	X		
Lot/Batch Number			X	X		

	Functional Area					
Process Area/Data Type	Source Product	Receipt	Storage	Disposal	Distribution	Marketing Information
			Stock Adjustment (Continued)			
Supplier Batch Number			X	X		
Quantity Disposed			X	X		
Date Quarantined			X	X		
Date of Disposal			X	X		
Reason for Disposal			X	X		
			Inventory Transfer			
Item Number			X			
Item Description			X			
Item Notes[a]			X			
Lot/Batch Number			X			
Date of Transfer			X			
Quantity Transferred			X			
Unit of Measure			X			
Transfer from Location			X			
Transfer from Warehouse			X			
Transfer to Location			X			
Transfer to Warehouse			X			

[a] Batch, Item, or Shipping Notes or other data type headings such as Warning or Hazard statements are often free text fields within an application and are purposely created to capture any comments, instructions or special conditions that need to be associated with the particular batch or item at all times. Care should be taken when validating the use of free text fields because these fields do not normally have any mandatory system verification placed on them to confirm that that data entered is of the correct data type (e.g., text or numeric values) before it is committed to the database. The only verification that can usually be tested is that data can be added, amended, and deleted from these fields and that the modifications made on one screen are correctly reflected in any subsequent screens. This factor is particularly important where large blocks of text are being entered, for example, text entries for medical information systems supporting products released to market. If a site's particular usage of the application is determined to be GxP-critical, then the system security programs controlling the integrity of any GMP or GDP data should also be subject to local validation activities, as shown in the table above.

37 Case Study 19: IT Infrastructure and Associated Services

Barbara A. Mullendore, Watson Pharmaceuticals
Chris Reid, Integrity Solutions

CONTENTS

IT infrastructure comprises all computer systems with their associated hardware, operating software (other than software applications), and networks used to run the business. Communication networks include servers used to transmit data between computers, as well as the computers used to manage the network. IT infrastructure therefore encompasses (see Figure 37.1):

- Mainframes, desktop, and laptop (mobile computing) environment
- Data centers and service management
- Storage devices
- Operating systems, software tools
- Network cabling, hardware, and communications software

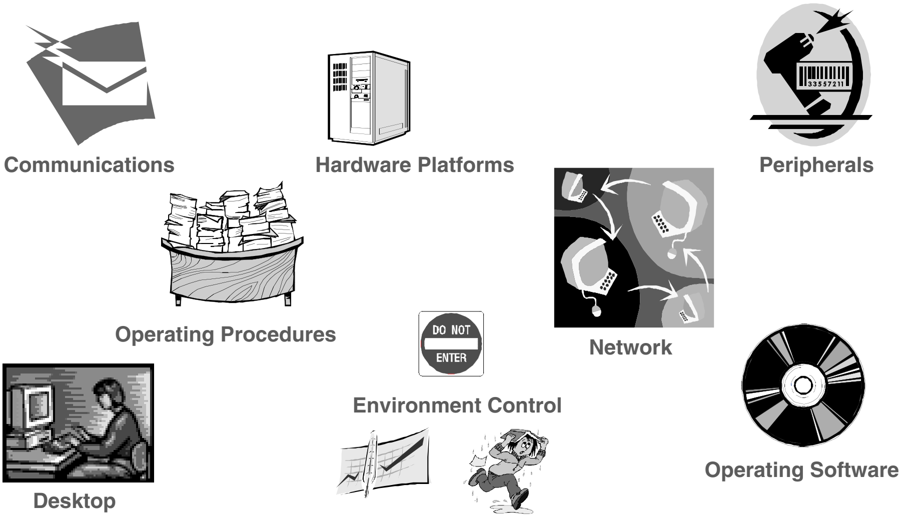


FIGURE 37.1 Elements of IT Infrastructure.

IT infrastructure typically evolves over a period of time in response to demand for application support, services, and storage. As such, IT infrastructure development does not follow a traditional life cycle development approach; rather, it is subject to management controls. Such management controls ensure that the introduction, modification, and disposal of software and hardware components is evaluated, managed, and verified in order to ensure that performance and the integrity of critical applications and data are maintained.

Validated computer system applications used by pharmaceutical organizations can no longer be viewed in isolation. These systems are increasingly being interconnected by means of a communications infrastructure based on both local and wide area data networks. IT infrastructure and associated support services must be suitably quality assured to support validated applications. Pharmaceutical organizations that do not align their IT departments with current good practices run the risk of undermining the significant application qualification undertaken by Users and Project Teams.

Until recently, infrastructure compliance was not a significant regulatory issue however; recent inspections have indicated the importance of developing and maintaining a compliant IT infrastructure. Regulatory observations of noncompliance to date are on the themes of validation and the role of QA, network documentation, and operational controls.

The consequence of an IT infrastructure outage or regulatory citation should not be underestimated. Depending on the scope and severity of a regulatory observation, an entire drug research, development, manufacturing, or distribution site or geographic region could be brought to a standstill while the noncompliance is resolved.

MANAGING INFRASTRUCTURE

End Users

End users usually have good knowledge of GxP requirements; however, end users are not typically engaged in the management of IT infrastructure. IT infrastructure is typically the responsibility of Information Systems or Computer Services Groups who traditionally are not familiar with regulatory expectations. As such, it is essential that the IT department quality awareness and culture is raised.

Quality Assurance

The scope for quality assurance embraces all components within the Local Area Network (LAN), e.g., the bridges and routers and the interface to the public telephone and telecommunications carrier (PTT) but not beyond this. The public domain infrastructure, e.g., Internet that connects site LANS, is not within the control of the pharmaceutical organization. The PTT is entrusted to pass the signal without error to the router at the destination network interface. It is not recommended to qualify the PTT!

Setting Priorities

IT infrastructure supports the whole organization which means that both GxP and non-GxP applications and data will reside on the infrastructure. Typically, a risk-based approach is taken to the management of applications and data. The rigor applied to the management of GxP applications and data would usually be significantly higher than the rigor applied to less critical applications and data. However, with respect to IT infrastructure, care must be taken as the distinction is less easy to make and modification to a non-GxP aspect of the infrastructure may have an impact on a GxP aspect. As such, careful design is required in order to appropriately partition GxP and non-GxP aspects of the infrastructure.

Network infrastructure is rarely implemented from scratch as a project. Normally, it evolves as actual and predicted needs require. For all but the smallest enterprises, a methodical phased planning approach must be adopted. Risk assessment should be used to identify components of the infrastructure that pose greatest business and compliance risk. Determining the probability of a failure, as well as the potential cost of that failure, in terms of business, drug product quality, or research and development data integrity, helps to determine where the more rigorous controls should be applied.

IT groups should initially focus on the following baseline activities:

- Develop high-level overview document(s) of the infrastructure.
- Establish logical topology drawings and configuration specifications.
- Generate master list of significant infrastructure components.
- Issue operating procedures covering scope of services.
- Conduct risk analysis of the entire infrastructure including a graded identification of potential points of failure.
- Establish document management controls.
- Qualify critical components and services.

Subsequently this state should be maintained with:

- Configuration Management
- Change Control
- Operational Management Control

The effort to achieve these three aspects of compliance should not be underestimated or intentionally made overly complicated.

CRITICAL CONTROLS

Client Environment

Procedural controls should be established in order to manage the distribution of client software and associated configuration. Traditional methods of distributing software to the client (e.g., installation from distribution media such as CD ROM, DVD, and floppy disk) are being surpassed by automated deployment methods that distribute applications to the client on mass. It is important that processes are established to manage and verify such automated deployment processes. Suitable records should be available to demonstrate successful application deployment and configuration management.

Conflict/compatibility testing should be conducted in order to ensure that applications are able to coexist on the client PC. A typical example of conflict/incompatibility is where two applications installed on a client PC utilize the same Dynamic Link Library (DLL) files but different versions. Controls should be implemented to prevent two applications using the same DLL files or to evaluate and test the impact of an application using a version of a DLL for which it was not designed.

Logical security controls should be applied to all client PCs containing GxP applications and electronic records. Security controls should conform to electronic record and electronic signature requirements. Such controls include (but are not limited to):

- Access limited to authorized users
- Security code issue and reissue
- Manually locking PCs
- Automated locking of PCs after a defined period of inactivity
- Disabling of user accounts following a defined number of failed login attempts

The standard client configuration should be documented and critical aspects (components that could impact GxP application operation, functionality or data authenticity and integrity) of the client subjected to change control and configuration management. Acceptance testing of standard client configurations can be based on statistical sampling where the process of setting up the configuration is quality assured. Organizations may want to consider segregating all GxP-regulated applications onto a standard client so that qualification efforts can be focused. Processes should be established in order to evaluate the upgrade of client components on resident GxP applications, e.g., Database Access Software such as MDAC, Security Patches, File Readers such as Adobe Acrobat, etc. The effort required to assess the impact of client component upgrades on resident applications should not be underestimated. Often IT personnel do not have knowledge of application criticality and users do not understand the technical impact of the client components on their applications. Bringing users and IT together in a global organization is not always straightforward.

Processes should be established for the distribution of virus protection software and the maintenance of virus detection databases in order to maximize the company's ability to detect and eradicate viruses.

It should be noted that it is not always possible to apply standard configurations to all clients. For example, many networked laboratory clients cannot be validated against a standard client configuration. Nonstandard clients, however, should still be subject to the principles set forward by this chapter.

Servers

Servers containing GxP applications or data should be subject to controls that ensure that the application and data integrity is maintained.

Servers should be subject to configuration management and change control procedures. Configuration records should be in place for server hardware and software.

Servers should be backed up in accordance with appropriate SOPs. The ability to restore multiple and single files should be tested and documented.

Procedures should be in place to archive GxP data at defined intervals and to ensure that such data can be readily retrieved for the duration of the retention period. The integrity of backup and archive media should be verified at appropriate intervals throughout the retention period.

Backup and archive media should be stored in secure locations subject to appropriate environmental controls. Backup and archive media should be adequately labeled to enable clear identification when required.

Backup and archive services are often outsourced to third-party organizations. Where this is the case, the third-party organization should be assessed in order to ensure that they have appropriate facilities and processes in place. Service requirements should be stated in appropriate contract documents or Service Level Agreements (SLAs).

Servers should be located in secure locations subject to appropriate environmental controls and protected against risks of flooding, fire, etc. Business Continuity Plans and Disaster Recovery Plans should be in place to manage catastrophic events. Such plans should be periodically tested.

Configuration records should be in place for server hardware and software. Organizations may wish to consider established servers dedicated to GxP-regulated applications so that qualification can be segregated.

Networks

Specifications and diagrams should be in place that describe the LAN and identify critical physical and logical components of the network. Specifications and diagrams should identify key entry points to the network and how security is managed. Such specifications should be reviewed and approved.

Network specifications and diagrams should be in place for Wide Area Networks (WANs) to show interrelationship between company LANS. Remarkably, there are no expectations to document the interconnection of LANs via the Internet.

Network diagrams should identify all critical equipment, e.g., servers, routers and bridges, and should show connections between LANs and WANs.

The boundaries between open and closed networks should be documented along with methods for protecting data such as access controls, firewalls, and cryptographic techniques where employed.

Critical network configuration should be subject to installation controls, testing, ongoing monitoring, change control, and configuration management.

Communication protocols should be documented, e.g., Transmission Control Protocol/Internet Protocol, SNA, DecNet, etc.).

Middleware (communication software used to link different applications) should be installed according to defined procedures and tested as part of the computerized system it supports. Configuration parameters should be documented and verified.

Distinct data import and export features built into computerized systems are not middleware and should be validated as part of the application.

Tools should be used to monitor network operation and performance and security breaches. Typically, the network backbone should be monitored in order to detect deterioration in traffic throughput.

Appropriate naming conventions should be used for networks and devices.

Remote Access to the infrastructure by staff working from home or third-party support organizations should be carefully managed. Security features such as "Call Back," "Secure ID," and control of temporary connections to the network should be managed (e.g., only enable a connection when required).

Service Management

Procedural controls should be established in order to manage the ongoing support and maintenance of the IT infrastructure. This procedural framework is critical in keeping the IT infrastructure in a state of control, the objective being the maintenance of a continuous state of compliance as the IT infrastructure evolves to meet business requirements.

Typically, service provision is defined within SLAs between the IT function and the business. It is essential that service levels are monitored and any service shortfalls are monitored. Trends of service performance should be established in order to demonstrate delivery of consistent services, repeated problems, and where appropriate, service improvement. Typical service monitoring and reporting might include:

- Network performance and outage (network and storage capacity utilization)
- Problems reported and resolved
- Help desk performance
- Backup performance and backup failures (e.g., overruns)
- Security and virus issues
- Incident resolution

DOCUMENTATION AND INFORMATION MANAGEMENT

Paper-based documentation is not always suitable for management of volatile environments such as IT infrastructure. The time taken to update drawings, specifications, and other documentation can often be more significant than the time taken to implement the change. Further, new changes may be required before the documentation associated with the previous change has been updated. As such, it is often more appropriate to use service management tools or configuration management tools to improve the efficiency of information update. Such tools should be appropriately quality assured (and in some cases validated) and should provide suitable audit trail capability to enable tracking of changes.

TABLE 37.1
ISO/OSI Reference Model

Layer	Layer Title	Layer Description
7	Application	Interfaces directly with the application programs running on the network. This layer provides services such as file access and transfers, peer-to-peer communications, and resource sharing.
6	Presentation	Translation of data formats to enable computers using contrasting languages to communicate. Data encryption is handled in this layer.
5	Session	Establishes bidirectional communication between applications using conversational techniques or dialogues.
4	Transport	Ensures reliable message delivery and the control of data between systems in a flow of packets.
3	Network	Standardization of the addressing mode between multiple linked networks and services to ensure packets of information arrive at the correct destination.
2	Data Link	Defines the control of communication between two devices directly linked, together, and the packet and framing methods.
1	Physical	Defines the mechanical components, type of medium, transmission method, and rates available.

Infrastructure Architecture

The ISO/OSI reference model defines network infrastructure in terms of seven discreet layers, each describing a certain logical function in the transfer of data in a network. Not all of these seven layers need necessarily be present in any given network. The layers are described in Table 37.1.

It may be beneficial to document the infrastructure in accordance with the structured layers described in Table 37.1. Such documentation should contain:

- Drawings, including network topography
- Equipment and infrastructure software inventories (including revision status)
- Critical software and hardware configuration
- Standard Operating Procedures (SOPs), all routine operational and maintenance activities

Documentation Requirements

Documentation requirements shall be dependent on the nature and criticality of IT infrastructure software and hardware. Command files and script files should also be considered when defining documentation requirements. Typical documentation requirements for IT Infrastructure components are defined in Table 37.2.

The software above is typically validated *in situ* with an application that exploits it and therefore dedicated validation of the software is not required. Further, these components are widely used with pharmaceutical and other industries, software faults are readily publicized and fixes made available. It would not be normal practice to audit suppliers of the software outlined in Table 37.2.

The relationship between Project Documentation and current infrastructure status documentation needs to be considered. Projects delivering new applications will often develop Infrastructure Specifications defining new and modified infrastructure components. The information from Infrastructure Specifications must be migrated into operational documentation that shows the overall and current status of the infrastructure; otherwise it will only be possible to demonstrate the current status from a chronological history of specifications and change control records.

ESTABLISHING BASIC CONTROLS AND DOCUMENTATION

Formal plans and specifications are required to describe the extent of the current infrastructure and the plans for future improvements. This information has two purposes: (1) to provide a basis for

TABLE 37.2
Typical Documentation Requirements against ISO/OSI Model Components

Example	Typical GAMP Category	Documentation Requirement
For operating systems such as VMS, HPUX, AIX, and for network operating systems (e.g., Microsoft Windows NT®, Novell Netware®) relating to layers 5 and 6 in the ISO/OSI Model.	1	Operating system name, supplier, and version number recorded. Patch status recorded. Application dependency recorded.
For firmware-controlled devices (e.g., intelligent bridges and routers), relating to layers 1, 2, 3, and 4 in the ISO/OSI Model.	2	Configuration settings for baud rates should be documented. Part numbers or version numbers for hard encoded firmware should be documented.
For standard software packages (e.g., NetDirector® for networks) relating to layer 7 in the ISO/OSI Model.	3	Software version numbers should be documented.

the controlled management of the infrastructure and (2) to be able to provide evidence of control during a regulatory inspection.

As a starting point, basic documentation and procedural controls should be established. LAN and WAN (in the sense of relationships between LANs and not the Internet) diagrams are a useful entry point to infrastructure documentation. Such diagrams define:

- Logical organization of infrastructure
- Major cable routing
- Major components of the infrastructure, e.g., Servers, Routers, Bridges, Storage Devices
- Computer Rooms
- Organization of logical domains

Diagrams may be organized into geographical areas, platforms, or other logical parts. It should be possible to drill down from such diagrams into more detailed specifications of infrastructure components and inventories.

Each hardware and software component of the infrastructure may be defined as a "configuration item." Each configuration item may be then be categorized in terms of business and regulatory risk from which the level of control and information required to manage the configuration item can be determined.

Quality Manual or Quality Plan

Quality Manuals or Quality Plans should be established to manage corporate IT infrastructure. The Quality Manual or Plan shall collate references to all key management components including:

- Corporate IT strategies, policies, and standards
- High level overviews and status of computer rooms, servers, networks
- Organization of the IT group
- Development life cycle for infrastructure
- Detailed inventories and configurations
- Computer operations procedures
- Service delivery procedures
- Service support procedures

Documents describing the above should be approved, controlled, and available during an inspection.

Baseline Assessment

Although regulatory compliance is relatively new to IT organizations, this does not mean that there is a total void of processes and documentation. However, often such processes and systems are based on a "patchwork quilt" type of approach, i.e., individuals see a gap in processes and systems and plug it without taking a holistic approach.

When working within a global organization it is beneficial to conduct a Baseline Assessment in order to determine the current practices, systems, and capability in place at each site. The Baseline Assessment is normally conducted against corporate standards and industry practice.

The Baseline Assessment will enable current best practices and shortfalls to be identified and prioritized. Best Practice should be shared among the organization in order to minimize effort required to raise standards and to quickly bring the organization to a common platform.

Shortfalls should be reviewed and prioritized based on the severity of the shortfall and the number of sites affected. Cross-site and regional teams should be formed in order to provide a consistent solution across the organization. This approach further enables cultural differences and regional approaches to be addressed by any delivered solution.

Appendix 37A provides a questionnaire for assessing quality and compliance practices associated with infrastructure. In addition to asking specific questions as described in the questionnaire it is also useful to conduct interviews with key people from the organization including:

- Technical Architects (Designers)
- Service Managers
- Service Delivery
- Outsource Contract Managers
- Business Representatives (Customers)
- Security Managers
- Quality Assurance

Interviews will generally help to identify particular quality issues that are causing frustration within the organization.

DEVELOPMENT AND QUALIFICATION LIFE CYCLE FOR INFRASTRUCTURE

A disciplined and well-documented approach should be used when managing IT infrastructure. Documented evidence is needed during the life-cycle stages of planning, specification, selection, design and testing, installation, qualification, and operation.

Planning

Projects are normally initiated in response to a need for a new or modified applications, infrastructure or services, with one or more project teams being involved. It is essential that all activities are planned in order to ensure that:

- Appropriate activities are undertaken
- Activities (in particular between different project teams) are synchronized
- Accountabilities exist for all involved (Project Manager, Infrastructure Design, Application Design, Suppliers, Quality Assurance, Users, etc.)

- Relevant documentation/information is established
- Appropriate review and approvals can be organized
- Service interruptions can be planned
- Processes and procedures are determined for use
- Training and Development requirements are in place
- Standards (in particular internal IT standards) will be applied

Supplier Selection

Corporate IT places considerable reliance on many suppliers for infrastructure products and services. Where possible, the infrastructure should be constructed from proven standard components from approved suppliers. Caution is required when introducing new or novel technologies as they will be relatively unproven by industry and potentially contain undetected faults.

In the rare circumstance that bespoke software or hardware is developed, the potential consequence of failure should be evaluated and consideration given to auditing the supplier where there is a high risk.

Typically, supplier selection processes shall take account of:

- Commercial implications
- Reputation of supplier
- Support capability
- Industrywide use of supplier products

Technical Specification

The Technical Infrastructure and Network Specification[7] shall document the logical and physical architecture of the infrastructure components to be installed. The Technical Specification should define:

- Logical organization and relationship of servers, network components, storage devices, printers, etc.; the size of the network (number of user nodes) and topology diagram of the network (including interfaces to other networks) need to be specified
- Redundancy requirements
- Data integrity verification — ensure network hardware and software include error checking, handling, and correction measures commensurate with the applications the network supports (e.g., parity checking, checksum and cyclic redundancy checks and transaction roll-back after network failure should be facilitated; critical data files may be stored in duplicated separate locations)
- Cabling requirements
- Physical location of components
- Operating system requirements
- Middleware components, e.g., communication requirements, database access software
- Performance requirements
- Storage requirements
- Deployment of applications and databases
- Backup requirements
- Virus protection requirements
- Security configuration, e.g., logical access, domains

Some of the requirements stated above may already be defined in local or corporate IT standards. Where appropriate, the Technical Specification should reference such standards. The Technical

Specifications should be developed or, as a minimum, reviewed by a competent technical authority from the IT department.

Installation Plan

Installation plans should be created for the infrastructure hardware and software. They should identify the configuration settings and any dependencies or constraints. Where infrastructure has been standardized, standard installation plans may have already been established.

Qualification

Qualification will demonstrate that all software and hardware has been installed and configured correctly. Installation and Operational Qualification should address:

- Verification of hardware installation
- Verification of software installation
- Verification of hardware and software configuration and addressing
- Incorporation of servers and other storage devices within the backup and virus protection regimes
- Server start-up and shutdown
- Service start-up and shutdown
- Confirmation of updates to existing SOPs
- Testing of security settings
- Update to Operational Documentation and Service/Configuration Management Systems
- Verification of inventory update
- Application conflict/compatibility testing/verification

Performance Qualification of Infrastructure is not conducted in the traditional sense. Rather, an ongoing monitoring program should be established in order to ensure that the network and associated components provide adequate performance and data security. Ongoing monitoring should include:

- Network traffic (collision rates, throughput rates)
- Storage capacities
- Network diagnostic checks
- Unauthorized software installation
- Security and virus alerts

Many infrastructure components will be subject to standard build specifications, e.g., Print Servers, Application Servers, Domain Servers, etc. As such, it may be possible to develop standard, reusable Qualification protocols that can be reexecuted each time a new server is built and installed.

Operational Procedures

Operational procedures should be developed for:[4]

- Adding and maintaining network components
- Connecting to a network
- Removing network components
- Controlling network security
- Network management between sites

- Configuration management
- Disaster recovery plans/procedures
- Virus protection
- Backup and Restoration
- Security Management

Logs

Audit trail and log files should be enabled wherever appropriate in order to provide reporting of software and hardware installation and run time failures.

SPECIFIC DESIGN AND QUALIFICATION CONSIDERATIONS

The following sections illustrate some specific qualification requirements associated with network components.

Computer Rooms and Data Centers

Specific considerations for computer rooms include:

- Segregated areas for production equipment and work-in-process equipment
- Adequate areas for tape storage and rotation in a controlled manner
- Appropriate charts/drawings (data center layout, racking, cabling and wiring, electrical, etc.)
- Environmental specifications and controls and evidence of adherence to them
- Monitoring programs and maintenance programs
- Physical and logical security specifications and controls and evidence of adherence to them
- Automated alerts and evidence of testing
- Raised floors and antistatic floor covering
- Redundancy and backup systems (e.g., RAID, Automatic Changeover Systems)
- Flood and fire protection
- Entry logs

Good housekeeping practices should be exercised at all times. Examples of Good Housekeeping Practices include:

- No food or drink near sensitive computing equipment
- Organization of files and storage locations
- Labeling procedures for tapes and disks
- Storage facilities (e.g., fire- and water-resistant safes) archives, master media, backups, etc.
- Storage areas for operation and administration manuals
- Visitor supervision and access logs
- "Zero-tolerance" policy against "posted" passwords
- Clean clothing (i.e., no muddy boots)

Networks

Specific considerations for networks include the following:

- Hardware installation checks, including visual inspection of components against
  - Design specification

    - Standards
    - Statutory requirements
    - Manufacturer's recommendations
- Check that all equipment and materials are undamaged, clean, new and correctly installed (refer to installation records)
- Checks for hazardous area requirements
- Capacity testing
- Software versions checked
- Electrical supply and interference testing
- Installation diagnostic testing
- Power on-off testing (blackout testing)
- Configuration/system testing (each user port tested for connection to network)
- Simulated communication between network points
- Cable specification and definition
- Cable routing and redundancy, e.g., dual backbone
- Security

Client (Desktop and Laptops)

Specific considerations for Clients:

- Test application conflicts, e.g., runtime and DLLs
- Document standard client configuration
- Test with a combination of validated applications
- Document and verify application scripting and automated deployment processes
- Restrict installation of unauthorized software and downloads from the Internet
- Install virus protection
- Provide locked facilities
- Evaluate impact of client component upgrades on resident applications
- Document group policy settings
- Connecting to the Internet
- Installation of unauthorized software

Tools are now available to assist the management of clients, specifically supporting the installation qualification and change control. One important application is the rapid distribution of virus control software. A well-known example of such a tool is Microsoft's Systems Management Server (SMS) whose facilities include collating hardware and software inventories of clients, software auditing (including version checking), software distribution, and helpdesk/troubleshooting functions. These services are dependent on local scripts being resident on the client. Such scripts must be specified and tested before deployment. Testing is also required to verify that the application of the tool operates as intended. When using such auditing tools care should be taken as they do not always work when a client is switched off.

Additional controls should be applied to mobile clients, e.g., the laptop computers that are used away from the office. In certain cases, policy and procedural controls might be required as the technical security features provided by the IT infrastructure may not be available, e.g., securing the system clock. Typically, policies and procedures might include:

- Use of remote access security when connecting from a remote location (e.g., Call-back, Secure ID, Virtual Private Network [including PC firewall])
- Changing (or not!) system clock
- Private use laptop

SERVERS

Specific configuration for servers:

- Start-up and shutdown
- Virus protection
- Operating system version and patch status
- Configuration settings
- Switch and address settings
- Backup and Restoration
- Application conflicts, e.g., shared DLLs
- Security configuration
- Folder security settings

ELECTRONIC RECORDS AND ELECTRONIC SIGNATURES

Many electronic records are associated with the applications that reside on Infrastructure servers and storage devices. These records are defined and the requirements for the specific management are defined in other chapters of this book. Needless to say, IT department System and Database Administrators will have access to such electronic records. The following controls should be applied to System and Database Administrators:

- Should not modify or delete data from databases without appropriate User and Quality Assurance authorization.
- Should only hold access privileges commensurate with their current duties.
- Only a limited number of IT personnel should hold Super User or Administrator Access privileges.
- Appropriate security access control administration procedures should be in place.
- Users should be notified when backup operations fail.

Typically, records associated with IT operations will not be directly attributable to product quality or research and development data integrity. However, IT records are subject to regulatory inspection and therefore subject to the electronic records and electronic signature controls in U.S. FDA regulatory citations. Typical electronic records may include:

- Security configuration
- Electronic configuration management records
- Infrastructure specifications
- Electronic change control records
- Electronic LAN and WAN diagrams
- Standard Operating Procedures and Work Instructions
- Electronic diagnostic and fault reports (including virus alerts and security violations)
- Installation and deployment records
- Qualification records
- Specifications

Electronic signature requirements are dictated by the applications residing on the infrastructure and the processes operated by the IT department. Typical electronic signatures that could be of interest within the IT department include:

- Electronic approval of change control records
- Electronic authorization of security configuration and access rights
- Electronic approval of specifications, protocols, and drawings

It is important that the IT infrastructure is included in Electronic Records and Electronic Signature Assessment programs. In addition to those challenges that might be made against applications there are some challenges specific to infrastructure. An assessment should be completed for each platform, network, or architecture as appropriate. Sample challenges for an FDA 21 CFR Part 11 compliance gap analysis of infrastructure (as well as the associated references from the regulation) are outlined in Appendix 37B.

Once the responses to the challenges are known, the nature of the areas of noncompliance (if any) as well as the remedial action plans should be documented in a summary and conclusions document specific to the given platform/network/architecture.

Summary, conclusions, and overall compliance status for the given platform/network/architecture should be documented, as well as a detailed description of each area of noncompliance. For each noncompliance, the nature of the remediation as well as the associated estimated completion date should be documented. If the remediation is longer term, the interim stopgap measure that will be taken should be documented. Finally, and perhaps most importantly, the person accountable for the remediation action must be documented.

These summary and conclusions documents can then be "rolled up" into an overall Infrastructure Part 11 Compliance Plan. Progress against this plan should be tracked as part of the overall Part 11 compliance program until full remediation is complete.

A degree of caution should be sounded when considering electronic records and electronic signatures with respect to IT infrastructure management records. Many IT organizations are still trying to manage IT infrastructure information using traditional paper systems. As such, documentation is not updated as it is perceived as being burdensome. However, automated service and configuration management tools that would bring about efficiency gains and improved data integrity are not implemented due to limited electronic records and signature compliance. Regulatory compliance within IT infrastructure departments is relatively new and therefore it is recommended that any steps that improve compliance should be taken rather than constraining implementation of solutions because they are not fully compliant with electronic records and electronic signatures regulation. Such solutions can adopt a hybrid approach.

OPERATIONAL/PROCEDURAL CONTROLS

The compliance status and integrity of the infrastructure can be preserved only by the effective execution of documented processes addressing the following. Table 37.3 identifies some typical procedural controls that should be implemented by the IT infrastructure department.

Training

All IT personnel must be trained in relevant procedures. All training must delivered by competent personnel and should be documented in appropriate training files.

In addition to procedural training, IT personnel should be trained in relevant regulatory requirements. All training should ensure that regulatory requirements are appropriately translated to make them specific and relevant to IT infrastructure services.

INTRANET, INTERNET, AND EXTRANET ISSUES

The qualification requirements for Internet/intranet environments are largely dependent on the use made of this environment by a system or application. GxP operations such as procedures, complaints,

TABLE 37.3
IT Infrastructure Procedural Requirements

Area or Aspect	Processes Requiring SOPs	Minimum Deliverable Documentation
	General Management	
Roles and responsibilities	• Management processes and allocation of responsibilities	• Organization chart • Job descriptions
Training	• Organization • Delivery • Assessment of effectiveness	• CVs • Training records • Records of the acquisition of competencies
SLAs, operating level agreements, underpinning contracts	• Management of suppliers and third-party relationships • Establishing formal agreement, including definition of responsible representatives • Maintenance of the agreements and contracts	• Contractual document
License management	• License usage monitoring • Monitoring of authorized software	• Licenses • Monitoring logs and corrective actions
	Data Center Management	
Data center management	• Procedural controls on data center activities • Physical security access	• Operating procedures • Description of security management
Computer room controls	• Computer systems operating environment (UPS, RFI, EMI, humidity) • Fire protection and safety management	• Performance qualification • Description and periodic check records
Capacity and performance management	• Monitoring service loading and performance against performance capacity	• Periodic service reports
	Systems Management	
System hardware and software installation and changes (including servers and peripheral equipment)	• Physical installation and qualification of new hardware and software; of changes to existing hardware and software • Adjustment of configuration parameters • Description of redundancy features (disk mirroring, RAID devices)	• Installation qualification • Change control reports • Parameter change control records • Descriptions of hardware redundancy features
Client installation and changes	• Establishment of initial standard client • Evolution of standard client • Distribution of software • Upgrades • Maintenance of virus protection	• Installation qualification • Parameter change control records • Virus signature update records

TABLE 37.3 (Continued)
IT Infrastructure Procedural Requirements

Area or Aspect	Processes Requiring SOPs	Minimum Deliverable Documentation
Hardware and software maintenance (including servers)	• Preventative and reactive maintenance • System or application software patch installation	• Maintenance plan • Maintenance logs
Service start-up and close-down	• Start-up and shut-down • Implementation of service restrictions (e.g., TCP/IP, e-mail, databases access)	• Event logs • Access control lists
Job scheduling	• Assignment of batch job priorities • Ensuring proper completion of batch jobs and reprocessing when necessary	• Priority lists for applications • Change control logs • Deviation reports on failures
System monitoring, event/problem logging, problem tracking and reporting	• Capacity management • Establishment of performance metrics • Escalation • Help desk call management and resolution	• Capacity and performance reports • Event/exception handling reports • Help desk call records
	Network Management	
Organizational network hardware and software installation and changes (for cabling, bridges, routers, etc.)	• Physical installation and qualification of new hardware and software; of changes to existing hardware and software • Adjustment of configuration parameters • Network documentation maintenance	• Installation qualification • Communications operating system configuration records • Change control reports • Parameter change control records • Documentation change control records
Third-party networks	• Use of WANs • Interfacing of LANs to WANs	• Network topology diagrams
Hardware and software maintenance	• Preventative and reactive maintenance • System or utility software patch installation	• Maintenance plan • Maintenance logs
Service start-up and close-down	• Start-up and shut-down • Implementation of service restrictions (e.g., TCP/IP, e-mail, database access)	• Event logs • Access control lists
Service monitoring, event/problem logging, problem resolution tracking and reporting	• Capacity management • Establishing of performance metrics • Prioritization and escalation • Help desk call management and resolution	• Capacity, network usage, and performance reports • Network availability reports • Event/exception handling reports • Help desk call records
	Security Management	
Physical security	• Means of access to all system and network components (computer rooms, network rooms/cabinets, cabling, etc.)	• Access control logs • Security monitoring reports especially unauthorized access attempts

TABLE 37.3 (Continued)
IT Infrastructure Procedural Requirements

Area or Aspect	Processes Requiring SOPs	Minimum Deliverable Documentation
Logical security	• User account management • Password management including functionality rules, changes and related event reporting • Digital signature certificate management • Access rights maintenance • Detect and investigate security breaches • Domain configuration • Firewalls	• Logs of creation, deletion, transfers of responsibilities • Logs of password renewals, deletions, suspensions • Log security breaches • Partitioning network
Virus protection	• Installation of virus software • Maintenance of signature library • Handling of virus alerts and infections • Firewalls	• Installation qualification • Signature library change log • Virus infection reports
	Data Management	
Data back-up and restore	• Back-up scheduling, logging, recorded data verification, problem detection, deviation reporting • Media labeling and storage (on-site, off-site) • Restore process (including authorization to restore)	• Back-up logs • Restoration logs • Risk analysis reports • Event logs
Long-term data archiving	• Data management (e.g., in-house or devolved, data deletion from active directories, data restoration from archives, archived data expiry, and destruction)	• Archiving and restoration logs • Data deletion logs • Authorization records
	Quality Management	
Quality assurance	• Compliance with standards and SOPs • Implementation of corrective actions • Process improvement participation • Service Level Agreement performance monitoring • Internal audit schedule	• IT operational standards • Audit reports • Process Evaluations • Performance reports
	Configuration and Change Management	
Configuration management (where not otherwise covered)	• Maintenance of current and historical configuration	• Inventory reports (contemporary and historical)
Change management	• Logging, risk assessment, management, approval/rejection, tracking, implementation, and closure of change requests	• Change control reports

TABLE 37.3 (Continued)
IT Infrastructure Procedural Requirements

Area or Aspect	Processes Requiring SOPs	Minimum Deliverable Documentation
	Business Continuity	
Disaster recovery	• Continuance of service provision in event of catastrophes	• Disaster recovery plan
Contingency	• Continuance of service provision in event of less serious contingencies	• Business continuity plan

and labeling supported by Internet/intranet environments will attract more regulatory attention than non-GxP operations such as price listing for products. Regulators will typically consider applications using Internet environments as "open systems" requiring more controls than applications using intranet environments which they consider as "closed systems." The regulatory position is that Internet applications are managed by a third party with limited contractual obligations to the users while intranet applications are completely under the pharmaceutical organizations control.

Some specific issues associated with intranet/Internet applications are described in the following sections.

Electronic SOPs

Many pharmaceutical manufacturers are considering moving away from traditional paper-based SOP management systems to electronic SOPs with intranet-facilitated review/approval and distribution. In taking such a move it is reasonable to anticipate that regulatory authorities would check that:

- The review and approval of electronic SOPs are controlled and managed.
- Audit trails between versions are maintained.
- Electronic SOPs accessed by users are the current version.
- Electronic SOPs are provided to users in a read-only format to prevent unauthorized change.
- Electronic SOPs printed at point of use are not modified while the paper copy is in use.
- Other security measures to protect the integrity of electronic SOPs.

If electronic SOPs are managed through a validated Electronic Document Management System (EDMS), it could be argued that it is not necessary to validate the intranet environment as the EDMS qualification would have effectively qualified it. The intranet would only have to be validated in its own right where its functionality is not limited to library viewing and read-only access. The qualification approach to EDMS is discussed in Chapter 34: Case Study 16.

Some useful features that can be built into electronic SOP access systems include:

- Use of watermarking to differentiate printed copies from master SOPs
- Use of warning messages indicating the period of validity of printed SOP
- Disabling of copy, cut and paste, and save and save as functions from the viewing tool

Internet Data Entry

Consider the receipt of labeling artwork from a supplier through an Internet/intranet environment. The temptation to use this medium is great as it offers much improved process velocity and customer response times. Blindly accepting electronic artwork on the basis of some sort of assurance from the supplier that the content and format is correct at the point of receipt would almost certainly prompt an expectation for qualification. The exchange of artwork by fax after all generally requires a final confirmation signature on the printed artwork sent by post. If incoming electronic artwork is inspected to verify that it is acceptable then there is less reliance on the Internet/intranet environment.

Wireless Networks and Wireless Device Data Entry

Implementation of wireless networks is on the increase. In simple terms the fundamental concepts of this chapter will apply. However, with traditional network cabling it is clear how physical cables are generally secured from tampering within physical buildings. However, such physical protection is of limited use when transmitting GxP information via a wireless interface. Pharmaceutical organizations must ensure that they can demonstrate authenticity, integrity, and where appropriate, confidentiality of data transmitted via wireless networks through the implementation of security features, Public Key Infrastructure, and other techniques as appropriate.

Handheld digital wireless devices are now becoming available for remote communications access to intranet and Internet applications. Examples include mobile phones, pagers, two-way radios, and smart-phones. Communications are facilitated through a new standard called Wireless Application Protocol (WAP). Applications are written using Wireless Mark-up Language (WML). Data entry through such devices needs to confirm the identity of the user and their authorization before accepting data input. Equally, an audit trail of electronic records often needs to start with the wireless device. Similar issues exist with remote data capture devices used in the production and laboratory environments.

Extranet

Some pharmaceutical manufacturers are implementing extranet web-enabled applications established with suppliers and business partners. Extranet includes Virtual Private Networking (VPN) and offer cheaper Web solutions. Remote node access is achieved using a client with a browser connected to a corporate Web address or Universal Resource Locator (URL). The challenge is ensuring security across the Internet link. A secure session or tunnel is established between the VPN server and the end user workstation. Internet Service Providers (ISPs) and telecommunications carriers are endeavoring to provide a managed extranet/VPN service to corporate subscribers. A major benefit of Web-enabled applications is the ability to recover from disaster scenarios. Business Continuity Plans can actively make use of such applications.

Web Sites and Web-Based Applications

Web sites need to be tested as per any other application. It is important that links are maintained and that old Web site links are deleted once the new Web site is introduced. Data entry via the Web interface needs to be secure and tested.

Web-based applications used in GxP-critical processes are no different than other GxP applications and should be validated and subject to appropriate operation compliance procedures. However, Web-based applications are in principle readily accessible by the general public and therefore must be subject to secure user access controls. Further, having gained access to the particular application, it must not be possible to gain access to other applications located on the infrastructure.

Enterprise User Directory

Some organizations are moving toward a single digital signature for all applications running on integrated infrastructure. Such systems overcome the need to have multiple user accounts with different user IDs and passwords and are generally easier to manage. However, there are a number of issues associated with such an approach that need to be managed including:

- Suspension of access to systems once the person leaves the company
- Management of access to systems when people change their roles
- If the signature is disclosed to a third party are intentionally or otherwise, access is provided to multiple systems
- Access rights must still be managed for each application as access levels for one system may not be appropriate for all systems
- Need to be able to track which systems a user has access to for security investigation purposes

Using E-Mail

The use of e-mail is often taken for granted; however, it is important to consider how e-mail is used to support GxP operations. The qualification of such e-mail systems poses fundamental problems around the lack of audit trails, administration, robustness, and security, particularly with data passing from open to closed systems.

Where e-mail is routinely used to communicate authorization and approvals, regulatory authorities will expect to see evidence that the authorization and approval mechanism is secure and robust. In such cases Public Key Infrastructure (PKI) techniques and use of Digital Signatures will bring additional security necessary to ensure robustness and security of transfers.

IT departments often use e-mail to authorize users. E-mail can be used in this situation if the password is sent in two parts between users and their line manager so that users cannot attain the whole password until authorized by their line manager who would authenticate the users.

Where the e-mail system is only used to share information and any GxP information is securely maintained by another validated system, it can be argued that the e-mail system need not be qualified. It is important that policies and procedures are in place to ensure that information transmitted by e-mail is suitable for its intended purpose and that it is only used that purpose. Records sent via e-mail for information purposes should not be used to demonstrate regulatory compliance or to fulfill regulatory requirements.

The FDA recently announced it was using secure e-mail to communicate with pharmaceutical companies whereby e-mail messages are encrypted and decrypted by senders and receivers. Such e-mail messages should incorporate text indicating that it has been securely encrypted.

Care should be taken to avoid automatic purging of e-mails when transmitting electronic records for regulatory purposes.

OUTSOURCING

Many pharmaceutical organizations are now considering or implementing outsourcing contracts for their infrastructure services for a variety of reasons. First and foremost is to focus internal resources on more specialized activities contributing directly to the research, development, manufacturing, and distribution of drug product. This can be a good option, as long as those responsible for implementing the outsource agreement and the service provider understand the specific quality and compliance requirements of the pharmaceutical industry.

It is also important to include Quality Assurance in the process leading up to the decision to outsource, the assessment and selection of the service provider, the development of quality-related aspects of the contract, and the implementation and monitoring of the agreement. This arrangement should not be executed within an "IS/IT only" scope.

Regulatory Compliance Position

There may be some concern based on FDA 21 CFR Part 11's definition of "open" systems and the additional requirements surrounding them when considering outsourcing agreements. However, a solid argument can be made as to why outsourcing does not automatically change the status of a company's computer systems from "closed" to "open" status.

One of the key principles of FDA 21 CFR Part 11 is the differentiation between closed and open systems and the requirements associated with each in order to comply. The FDA defines the two types of systems as follows:

> *Closed system: An environment in which system access is controlled by persons who are responsible for the content of electronic records that are on the system.*
>
> *Open system: An environment in which system access is NOT controlled by persons who are responsible for the content of electronic records that are on the system.*

The key to claiming that systems remain closed when an outsource organization is used is that the outsource organization is seen as an extension of the pharmaceutical organization and are working to strict contractual obligations. This position is further supported by the clear definition of the pharmaceutical company's and outsource organization's roles and responsibilities. A system that is accessible by people other than the users and local IS can still be considered closed, as long as those people are within or working on behalf of the client company.

Where a system becomes open is when it contains a component somewhere along the data flow that is solely controlled by a third party, without the ability for the company to know or control access to the records. This is typically not the case with the outsourcing of infrastructure services.

Some considerations that might help ensure closed system status include the following:

- Avoid directly interfacing outsource and client company networks.
- Ensure that only designated personnel from the outsource organization have access to the client company network.
- Ensure that security access procedures are reviewed and require client company authorization of system network and system access. Audits should be conducted to ensure that security procedures are adhered to.
- Ensure that adequate controls are built into the governance structure/arrangements to demonstrate that the client company is in control of data and records.
- Ensure review, acceptance and transparency of outsource organization's processes, systems, and records.

The view on outsourcing of infrastructure as indicated in a conversation with Paul Motise of the FDA is that the systems could still be considered closed provided that the "wording in the contract is very specific about who has the ultimate control over system/data access and controls (must be the customer to consider systems closed)."

Due Diligence Process

For obvious reasons, a company should not enter into an agreement with an infrastructure service provider without undergoing a robust due diligence process. This helps ensure a clear understanding of the prospective provider's policies, practices, and work processes within the provider's own facilities, but also an understanding from the client company perspective of:

- How client company infrastructure management practices and roles are affected by the outsource agreement

- What the challenges have been and how they were overcome
- How the level of service and compliance coming from the provider has been
- How knowledgeable and responsive to the unique needs of the pharmaceutical industry the provider has been

These and other points can be determined by talking and/or visiting with the existing client companies. It may be surprising but most companies are willing to share their knowledge on these matters. These interviews should also be combined with a thorough on-site assessment at the prospective provider's facilities. This assessment should include such items as:

- Document management practices
- Work processes and personnel
- Facility compliance
- Security practices

The Contract

With respect to compliance, quality and security, the pivotal portion of the contract should be an appendix that covers the ongoing conditions and controls by which the client company can demonstrate ongoing compliance and robust security measures, especially with respect to FDA 21 CFR Part 11 requirements. This will hold the service provider (and the client company) accountable for implementing and sustaining these controls, and thereby sustaining compliance over the life of the contract. This appendix should also make provision for process improvement and/or additional measures to be implemented in the future, should any changes in requirements arise. Table 37.4 summarizes subjects suggested in Appendix 37A.

It is recommended that both client company and service provider appoint Regulatory Compliance Officers and Information Security Officers accountable for maintaining a state of regulatory compliance and secure operations.

Aside from the contract, every effort should be made to include compliance and security service levels in the overall Service Level Agreement (SLA). Although difficult to quantify, it is important

TABLE 37.4
Suggested Contract Content from a Compliance, Security, and Quality Perspective

Definitions
Organizational Structure
Compliance with Client Policies (e.g., GxP, qualification, electronic records and signatures)
Changes to Client Policies
Provider Personnel and Training
Roles & Responsibilities for Regulatory Compliance
Strategy and Planning for Regulatory Compliance
Roles and Responsibilities for Security
Strategy and Planning for Security
Account Authorization and Administration
Incident Reporting and Investigation
Time Synchronization
Networks
Compliance/Regulatory Training Program
Audit Requirements
Other Relevant Subjects

to ensure via SLA or other means that the provider understands that unacceptable levels of compliance and security are detrimental to its success in the arrangement.

Transition Planning

Depending on the scale of the outsource agreement, it may be advantageous to phase the handover of the services to the outsource organization in order to ensure that all quality objectives are met before moving onto the next service. It is important that the contract or associated SLAs define the success criteria to demonstrate that both the client and outsource company are meeting their defined obligations and business benefits are being realized.

Transition plans should define what needs to be accomplished, by whom, and by when. These plans should be jointly agreed to by both the client company and the service provider, and should include such items as:

- SOP modifications (to reflect the arrangement and any roles and responsibilities and/or process changes)
- Relevant teams and forums including scope and remit for each
- Conduct of assessments
- Issue escalation routes and reporting schemes

It must be possible to demonstrate to regulatory authorities that control is being maintained throughout the transition to the outsource organization. At any point in the transition, it must be possible to demonstrate that roles and responsibilities are clearly understood and are being executed by the appropriate organization. Further, it should be possible to demonstrate that GxP compliance and in particular system security and data integrity is being maintained at all times.

Points to Consider When Outsourcing

Table 37.5 identifies provides an indication of some real issues that might be encountered when outsourcing IT services. A significant investment in terms of expertise may be required to support the outsource partner, but this is likely to be a lot cheaper and easier than having to change outsource partners because of an unacceptable level of service. This support may come directly from the pharmaceutical manufacturer's own staff or through a third-party consultancy.

CULTURAL CHANGES

Quality Assurance within an IT environment is as much a cultural issue as a technical one. Change management skills are vital to successfully establish and maintain a quality culture. A formal quality assurance program is likely to be seen as an unnecessary and expensive add-on with little or no return on cost. It will not be readily understood that quality assurance programs significantly avoid or reduce costs that are due to infrastructure failures and their resolution.

A modern IT department supporting computer rooms, networks, and the client environment is a very exciting place to work. There is a proliferation of new technologies and innovative ideas in both hardware and software. The evolution of IT technologies has led to changes in the structure and culture of support organizations and the new skills needed by individuals to design, maintain, and operate these systems.

Members of IT departments are not often familiar with the concepts of regulatory compliance. Similarly Quality Assurance groups require an appreciation of the technologies and processes within the IT group and an understanding that within the operating environment decisions have to be made in a timely manner to provide an uninterrupted service.

Before embarking on a cultural change program it is essential to first determine what a quality culture will look like when it is achieved and the starting point. The change program must be

TABLE 37.5
Points to Consider When Outsourcing

What will happen to existing service levels (a fall in service level is not always detrimental to GxP)?
What are the implications of different business risks across the infrastructure, e.g., GxP and non-GxP?
Whose processes, procedures, and systems will be used?
How will client and outsource company's processes, procedures, and systems be interfaced?
Will the outsource organization's processes, procedures, and systems be transparent to the client organization?
Which organization is responsible for defining, reviewing, authorizationing, and implementing changes to the outsource organization's processes, procedures, and systems?
Whose documentation standards will apply?
Who will own documentation? Are there any issues with shared documentation management responsibilities?
What will be the impact on client staff? Will they transition to the outsource company? What might the impact be? Will they be reluctant to hand over services?
How will local site issues be managed and prioritized within a global contract framework?
How are outsourced services accessed?
How will outsourced service quality be measured and reported?
What are the implications of a global outsourcing agreement involving multiple client company businesses, e.g., manufacturing vs. research and development and different regional approaches?
How will the outsource organization work with different client processes procedures and systems across regions and sites?
Who is accountable for the outsource organization's service performance, local sites, or contract managers?
How will the outsource organization interface with local business groups — directly, via local IS/IT, or otherwise?
How will total dependence on outsource company's processes, procedures, and systems be avoided? In particular, how will the client company regain control of processes, documentation, and information in the event that a new service provider is selected or services are taken back inhouse?

carefully designed to enable the IT department to recognize the needs of the new environment within which they are working and to ensure that the resultant changes are owned by the IT department rather than imposed by external groups.

Some key indications of cultural change may be:

- Appointment of an IT Quality Manager
- Quality demand reflected in IT budgets
- Internal development of pragmatic, compliant, and value-adding processes
- Increased consultation between IT and users
- Increased recognition of quality issues in decision making processes
- Ongoing improvement of processes and systems, driven by management and technical staff

Responsibility

Responsibility for qualification of the infrastructure should be jointly shared between the IT group and Quality Assurance functions. All qualification or requalification plans should be subject to formal review and sign-off prior to implementation by IT. Internal IT organizations are advised to establish their own quality assurance arrangements to facilitate this sharing of responsibility.

Role of the Supplier

Suppliers of software and firmware (embedded in hardware) incorporated into the infrastructure are obliged to demonstrate that their products are fit for their intended purpose. Unlike suppliers of application software used in pharmaceutical research, development and manufacturing processes, it unusual to audit suppliers of infrastructure software such as Operating Systems and Communications software. The quality of such software is quality assured *in situ* with the applications with which they are integrated.

SUMMARY

Pharmaceutical organizations today are totally dependent on accurate data stored and manipulated by validated business applications. IT Infrastructure is the platform for business applications and hence must be qualified and managed following approved procedures.

IT departments have traditionally not been directly worked in compliant environments and hence there is a lack of understanding in current good practices. Expectations from user and quality assurance groups will not be achieved unless there is a training program and culture change within the IT department.

To meet compliance requirements, fitness for purpose must be demonstrated through specification, installation, qualification, procedures, and trained personnel.

The consequences of not meeting current regulatory requirements are significant. However, compliance of the corporate IT infrastructure can only be achieved through careful planning, organization, communication, and management commitment.

REFERENCES

1. Gindin, S. (1999), *Guide to E-Mail and the Internet in the Workplace*, Bureau of National Affairs (www.info-law.com/guide.html).
2. Wingate, G.A.S. (2000), *Validating Corporate Computer Systems: Good IT Practice for Pharmaceutical Manufacturers*, Interpharm Press, Buffalo, IL.
3. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (www.ispe.org).
4. Crosson, J.E., Campbell, M.W., and Noonan, T. (2000), Network management in an FDA-regulated environment, *PDA Journal of Pharmaceutical Science and Technology*.
5. D'Eramo, P. (2000), Computer Qualification in an Internet Age, GAMP Concepts and Case Studies, ISPE Conference, Zurich, September 18–21.
6. GAMP Forum (2001), *Good Practice and Compliance for Electronic Records and Signatures: Part 2 — Complying with 21 CFR Part 11, Electronic Records and Electronic Signatures,* published by ISPE and PDA (www.ispe.org).
7. FDA (1999), Off-The-Shelf Software Use in Medical Devices, Guidance for Industry, FDA Reviewers and Compliance, September 9.
8. OECD (1995), GLP Computer Qualification (1995) Section 3(b)(ii).

APPENDIX 37A
BASELINE QUALITY ASSESSMENT

Ref.	Challenge
A	**IT MANAGEMENT AND ORGANIZATION**
A1	**Roles and Responsibilities**
A1.1	Are IT Management roles and responsibilities defined (e.g., remit, job description)?
A1.2	Are IT Quality roles and responsibilities defined (e.g., remit, job description)?
A1.3	Is the organization documented (e.g., organization charts)?
A2	**Capability and Competency**
A2.1	Are training plans in place for IT personnel?
A2.2	Have IT personnel received training in regulatory expectations (where appropriate)?
A2.3	Are training records in place to demonstrate that training has been delivered?
A2.4	Do training records document: • Description of training? • Date of training? • Instructor? • Evidence of attendance?
A2.5	Do training records demonstrate that the attendee understood the training?
A3	**Internal Organization Interfaces**
A3.1	Are interfaces between IT and other infrastructure organizations defined?
A3.2	Are service agreements in place between internal infrastructure organizations?
A4	**External Support Organizations**
A4.1	Are contracts and/or service agreements in place for all external service/support organizations?
A4.2	Have external service/support organizations been assessed (e.g., audited) against contract requirements?
A4.3	Is service performance monitored against defined service levels?
A4.4	Have service providers been trained in your company's procedures where relevant?
A4.5	Have service providers been trained in your company's security policy?
A4.6	Are there controls in place to ensure that only authorized personnel from the service organization have access to your network and files?
B	**QUALITY SYSTEMS**
B1	**General**
B1.1	Are IT projects managed in accordance with life cycle project management systems that meet the requirements of industry standards or internal policies?
B1.2	Is there an overview document (e.g., Quality Manual) describing the Quality Management System?
B1.3	Is the Quality Management System periodically reviewed for its effectiveness?
B1.4	Are Quality Metrics in place to enable measurement of Quality System performance?
B1.5	Are Documentation and Records Management processes, systems and/or procedures in place?
B1.6	Are automated support systems compliant with regulatory and company requirements, e.g., SOP systems, configuration management, change control, etc.?
B1.7	Are infrastructure documentation standards in place including: • Planning • Requirements Specification • Design Specification • Development • Installation Testing • Functional Testing • Report Requirements
B1.8	Does the QMS address operational processes, e.g.: • Change Management • Security Management • Backup and Restoration

Ref.	Challenge
	• Disaster Recovery • Archive and Retention • Help Desk • Client Management • Configuration Management • System Performance Monitoring • Maintenance • Problem Investigation • Decommissioning
C	**COMPUTER ROOMS AND DATA CENTERS**
C1	**Environmental Conditions**
C1.1	Are computer rooms and data centers environmentally controlled? Environmental conditions include: • Temperature • Humidity • Vibration • Radio Frequency Interference • Electro Magnetic Interference • Electro Static Interference
D	**INFRASTRUCTURE SPECIFICATION**
D1	**Hardware**
D1.1	Are inventories of hardware components in place?
D1.2	Are specifications, diagrams or other documentation in place to describe the Site Local Area Network including: • Complete network layout of the site showing the backbone cable path and location of main network objects, e.g., hubs, servers, etc. • For each area or building, the location of each network component and cable path • Network access points
D1.3	Are documented configuration specifications in place for each network component (e.g., Mainframes, Servers, Storage Devices, Transceivers, Repeaters, Bridges and Routers, etc.): • Manufacturers' details • Location • Addressing • System performance (processor speed, memory, disk space, BIOS, etc.) • Cards within component (including address) • Configuration settings
D2	**Network Organization**
D2.1	Are network trusts, domains, etc., documented (including access controls)?
D3	**Software and Configuration**
D3.1	Is there an inventory of all network control and monitoring software/tools? • Operating Systems • Communication protocols • Performance monitoring software • Virus protection • Backup and restoration • Software deployment tools (e.g., SMS)
D3.2	Is there an inventory of all applications and data storage areas within the network?
D4	**Cable Infrastructure**
D4.1	Are (internal or external) standards used to define cable requirements?
D4.2	Are cabling diagrams or specifications in place?
D4.3	Are cables tagged or labeled to aid identification?
D5	**Control of External Connections**
D5.1	Are connections to WANs defined?
D5.2	Are controls in place to ensure that only authorized users can access the system remotely (e.g., Secure ID or callback)?

Ref.	Challenge
D5.3	When a remote access link is terminated, is the user automatically logged off the network?
D6	**Electrical Supplies**
D6.1	Are backup power supplies (e.g., UPS) in place to guard against power loss to critical components?
D6.2	Do electrical supplies conform to earthing, loading, filtering, and safety standards?
D7	**Redundancy and Fault Tolerance**
D7.1	Have redundancy requirements been assessed, e.g., disk mirroring, RAID?
D7.2	Have requirements for automatic standby systems been defined?
E	**INFRASTRUCTURE QUALIFICATION**
E1	Are critical hardware components, e.g., servers, storage devices, etc., subject to installation verification?
E2	Are infrastructure tools, e.g., virus protection, backup, performance monitoring, etc., subject to installation verification and operational testing?
E3	Are computerized infrastructure tools, e.g., change control, configuration management, access authorization, etc. used?
F	**NETWORK PERFORMANCE AND FAULT MANAGEMENT**
F1	**Speeds and Capacities**
F1.1	Are procedures or automated controls in place to monitor network performance and capacities including: • Speed? • Bandwidth? • Storage capacities? • Disk performance (e.g., fragmentation, thrashing)? • Address clashes?
F1.2	Are procedures in place for reporting, investigating, and documenting network faults?
F1.3	Are event logs created and maintained in support of service performance monitoring?
G	**DATA MANAGEMENT, DISASTER RECOVERY AND CONTINGENCY PLANS**
G1	**Backup and Restoration**
G1.1	Are procedures in place to assess backup requirements against business and regulatory needs?
G1.2	Have backup restoration procedures been formally tested?
G1.3	Are installed versions of Operating Systems, Communication Protocols, Applications, etc. archived in order to facilitate backup?
G1.4	Do backup procedures address: • Frequency of backups? • Physical labeling of media? • Review and retention of backup logs? • Periodic testing of backups to verify that the backup procedure is functioning? • On-site and off-site storage of media (full backups should be periodically stored off site)? • Rotation of backup media?
G1.5	Do off-site backup storage considerations include: • Location of facility? • Formal processes and controls over physical access to media both on a schedule and “on request” basis?
G1.6	Do restoration procedures adequately address the retrieval of single and multiple files?
G2	**Archive**
G2.1	Are decommissioning processes in place?
G2.2	Are processes in place for management of data deletion?
G2.3	Do processes, systems and/or procedures implement the requirements of company Archive, Records Management, Retention and Disposal policies?
G2.4	Do archive procedures include: • Identification of archive media? • Management of archive media? • Documentation of records to be archived? • Retention periods? • Secure and safe storage of archive media? • Frequency of archiving?

Ref.	Challenge
	• Periodic evaluation of archive media? • Migration following system upgrades?
G2.5	Do archive restoration procedures address: • Authorization to request records from archive? • Procedure for performing restoration?
G3	**Business Continuity Plans**
G3.1	Are contingency plans in place to manage critical processes and maintain data integrity in the event of a failure?
G4	**External Data Management Organizations**
G4.1	Are external organizations managing backup and archive facilities subject to appropriate controls including: • Contact/service definition? • Audit? • Performance monitoring?
H	**NETWORK ACCESS AND SECURITY**
H1	**Security General**
H1.1	Are processes, systems, and/or procedures in place to address the requirements of IS Security Policies?
H2	**Physical Security**
H2.1	Are servers and other critical hardware located in secure areas where access is controlled by key or other security device (e.g., card key)?
H3	**Logical Security**
H3.1	Are responsibilities for security management defined?
H3.2	Are firewalls in place and documented in order to control access to the network?
H3.3	Are procedures in place to ensure that users are restricted to those parts of the network required to fulfill their defined role?
H3.4	Is virus detection software in place?
H3.5	Are controls in place to ensure that unauthorized software and files cannot be loaded into the network?
H3.6	Are procedures in place to detect and investigate potential security violations?
H3.7	Are user IDs two-component and unique?
H3.8	Do user accounts automatically time out after a period of inactivity?
H3.9	Are user accounts disabled after a defined period of inactivity?
H3.10	Are users removed from the system when they leave the company or change jobs?
H3.11	Are there documented rules for password management? These should include: • Passwords should not be written down. • Passwords shall not be shared. • Users should change their password upon logging into an account for the first time or following modification of the password by anyone other than the user. • Prevention of the use of common words. • Passwords should expire on a periodic basis. • Policies in place to discourage reuse of passwords. • Minimum password length should be five characters.
H3.12	Do procedures exist to manage cards and tokens, including? • Issue of temporary and permanent cards and tokens, consistent with the security, account management, and password procedures? • The testing of their correct operation upon issue and periodically thereafter? • Cancellation in the event of loss?
H3.13	Are user access rights documented?
I	**CONFIGURATION MANAGEMENT**
I1	**Physical Controls**
I1.1	Are development, test, and production environments managed in order to ensure that software, hardware, and configuration integrity is maintained?
I1.2	Are GxP and non-GxP areas segregated or are GxP-level controls applied to both?
I2	**Procedural Controls**
I2.1	Are change control procedures in place to manage changes to network hardware, firmware and software, including impact assessment of any application affected by change?

Ref.	Challenge
I2.2	Do change control procedures require testing to be conducted when hardware or software is added, removed, or modified within the infrastructure?
I2.3	Do change control procedures address the management of emergency changes?
I2.4	Are installation plans used to control the installation and verification of new software hardware and software on the system?
I2.5	Are specifications, configuration statements, and other documentation updated following changes to hardware and software?
I2.6	Do configuration statements document the following information for hardware and software installed on the network: • Item name or identifier • Serial number • Model or hardware type • Manufacturer • Item location • Storage devices • Operating system software, including version • Layered products, including version • Relevant application software, including version and the system owner
I2.7	Are controls in place to control access to system documentation?
I2.8	Are retention periods defined for system documentation in line with the site/function record retention schedule?
J	**CLIENT MANAGEMENT (DESKTOP)**
J1	Is the standard client defined?
J2	Are local extensions/configurations to standard client defined?
J3	Are processes in place to management to deployment of client applications?
J4	Are processes in place to audit client configuration?
J5	Is client configuration documented?
J6	Are processes in place to management the build of new clients?
J7	Are processes in place to manage upgrades to the client?
J8	Are processes in place to maintain up-to-date virus protection?
K	**SERVICE MANAGEMENT**
K1	Have service start-up and close-down processes been defined?
K2	Have processes for implementing and communicating service restrictions been defined?
K3	Are facilities in place for fault reporting and tracking (e.g., Help Desk)?
K4	Are support services defined (e.g., first, second, third line support)?
K5	Are escalation procedures in place for management of service shortfalls?
K6	Are continuity plans in place to address critical service outage?
L	**CHANGE MANAGEMENT**
L1	Are change management processes in place?
L2	Do change management processes include risk/impact assessment?
L3	Are IS/QA/User responsibilities defined for change management?
L4	Are patches, configuration changes, etc. subject to change control?
L5	Are changes tested/qualified?

APPENDIX 37B
ELECTRONIC RECORDS AND ELECTRONIC SIGNATURE INFRASTRUCTURE CHALLENGES

Training and Personnel

Is there adequately documented training, including on the job training, for the following groups? 11.10(i)

- System Administrators
- System Developers
- IS/IT support staff

Security

Is access to the system platform/network/architecture limited to authorized individuals, with their details recorded and maintained up to date? 11.10(d)

Is there an identified group responsible for platform/network/architecture security — both logical and physical? 11.300

Documentation Controls

Is there distribution and access control over infrastructure operations and maintenance documentation? 11.10(k)(1)

Is the distribution of sensitive documentation, such as information on system security features, controlled? 11.10(k)(1)

Change Control

Do procedures/documentation exist for the design, installation, qualification, and maintenance of the platform/network/architecture/technical infrastructure components? Are these fully versioned and change-controlled? 11.10(k)(2)

Is System Documentation (e.g., design, installation, qualification, and maintenance documentation) available for the platform/network/architecture? 11.10 (k) (2)

Is there a current inventory of all hardware and software components? 11.10(a)

Is there a change control procedure to ensure that all hardware and software changes are properly documented? 11.10(k)2

Is a change history maintained for this platform/network/architecture? 11.10(a)

Policies

Is there a written policy that makes it clear that individuals are fully accountable and responsible for actions initiated under their electronic signatures in the same way as for their hand-written signatures and has this policy been communicated? 11.10(j)

Is there a procedure requiring a formal investigation into suspected instances of electronic signature falsification? 11.10(j) 11.300(c)

User ID/ID Device and Password Controls

Is the identity of an individual verified before assigning a user ID, card, or token? 11.100(b)

Is there a procedure to periodically check, recall, or revise passwords, user IDs, cards, or tokens and to test that the latter function properly and have not been altered? 11.300(b), 11.300(e)

Are there procedures that address the loss or compromise of user identification devices (cards/tokens, etc.) or passwords including electronic de-authorization, immediate and urgent reporting, and rigorous control of temporary or permanent replacements? 11.300(c), 11.300(d)

Is there a procedure for recalling, as appropriate, a user ID, card, or token in the event an individual leaves the position, the company, or is transferred? 11.300(b), 11.300(c)

Is there a procedure assuring that repeated or serious attempts at unauthorized password usage are reported to organizational management? 11.300(d)

Is it assured that the electronic signature is unique to an individual and cannot be used by anyone else — including system administrators? 11.100(a)

Are user IDs assigned in such a way that they are never reused? 11/100(a) 11.300(a)

Do passwords periodically expire? 11.300(b)

Remote Access

Is there an additional level of authentication for remote access? 11.200(a)i

Are there additional controls around third-party remote access? 11.10(d)

Clock Settings

Is the time/date stamp applied by the system to any records reliable and can any alterations made to it be readily identified? 11.10(e)

Open Systems

Do the measures for open systems architectures, additional to all the above, assure authenticity, integrity, and required confidentiality of records and signatures? 11.30

Archiving

Do adequate procedures exist for the archiving and retrieving of media and data, which includes (where needed) archiving and storage of obsolete software and hardware needed to retrieve the electronic records through the required period of retention? 11.10(c)

Are these periodically tested through the required period of retention? 11.10(c)

Back-Ups

Are backups performed on a regularly scheduled basis? 11.10(c)

Is there documentation of regular backups?

Is the backup process periodically tested?

Is backup/archived media rotated to prevent degradation?

Disaster Recovery

Do disaster recovery/business continuity plans exist for this platform/network/architecture? 11.10(a) 11.10(c)

38 Case Study 20: Local and Wide Area Networks

Nicola Signorile, Aventis

CONTENTS

Modern IT applications would not be possible if it were not for innovations in communications network technology. Indeed, major IT system vendors and consultancy firms, including Digital Equipment, were saying at the end of the 1980s, “The Network is the System.” Over the past 20 years IT systems have evolved from the development of centralized systems, through main frame systems, to distributed computing environments. Supporting network developments include client/server technology and intranet/Internet technology. These developments have offered users ever more flexibility and functionality but at a price — the network systems supporting IT applications have become more and more complex.

Within the pharmaceutical industry IT applications are increasingly being used to support the manufacture of drug products. These applications must be validated to fulfill GxP regulations, and so, too, should any support networks. Chapters 6 through 10 discussed a range of IT applications and how they might be validated. This case study considers the validation of communication networks. The integrity of data being carried by a network must not be compromised. An IT application may be perfectly functional but as the IT fraternity says, “Garbage In, Garbage Out.” Networks usually have a potential GxP impact on IT applications. Validation of IT applications should take a “systems approach” and not ignore supporting networks.

NETWORK APPLICATIONS

Networks are used to link collections of independent computers and devices (such as printers), providing a shared communications medium over which the computers can transfer information. Prior to the development of networking technology, individual machines were isolated and hence their range of applications limited.

Local Area Networks (LANs) are those networks usually confined to a small geographic area, such as a single building, group of localized buildings, or a site. LANs are not necessarily simple in design; some may link many thousands of systems and service hundreds of users. The development of various standards for networking protocols and media has made possible the proliferation of LANs worldwide for business and manufacturing applications.

Wide Area Networks (WANs) are those networks installed over a wide geographical area, typically linking multiple sites. They can cross national boundaries and join continents. LANs are often connected to WANs to create a communication web. For many multinational companies the combined LAN/WAN topology is akin to the human body’s central nervous system.

This section describes two examples of IT applications that require their supporting networks to be validated. The first example describes a Manufacturing Execution System (MES) based on a LAN and the second example describes an Enterprise Resource Planning (ERP) application that uses a WAN.

A Site MES Application (Manufacturing Execution System)

The architecture of the system is shown in Figure 38.1. The system provides Electronic Batch Record (EBR) functionality interfaced to a Laboratory Information Management System (LIMS).

The system is a typical client/server application that has its own main functions distributed on the LAN. The main functions are:

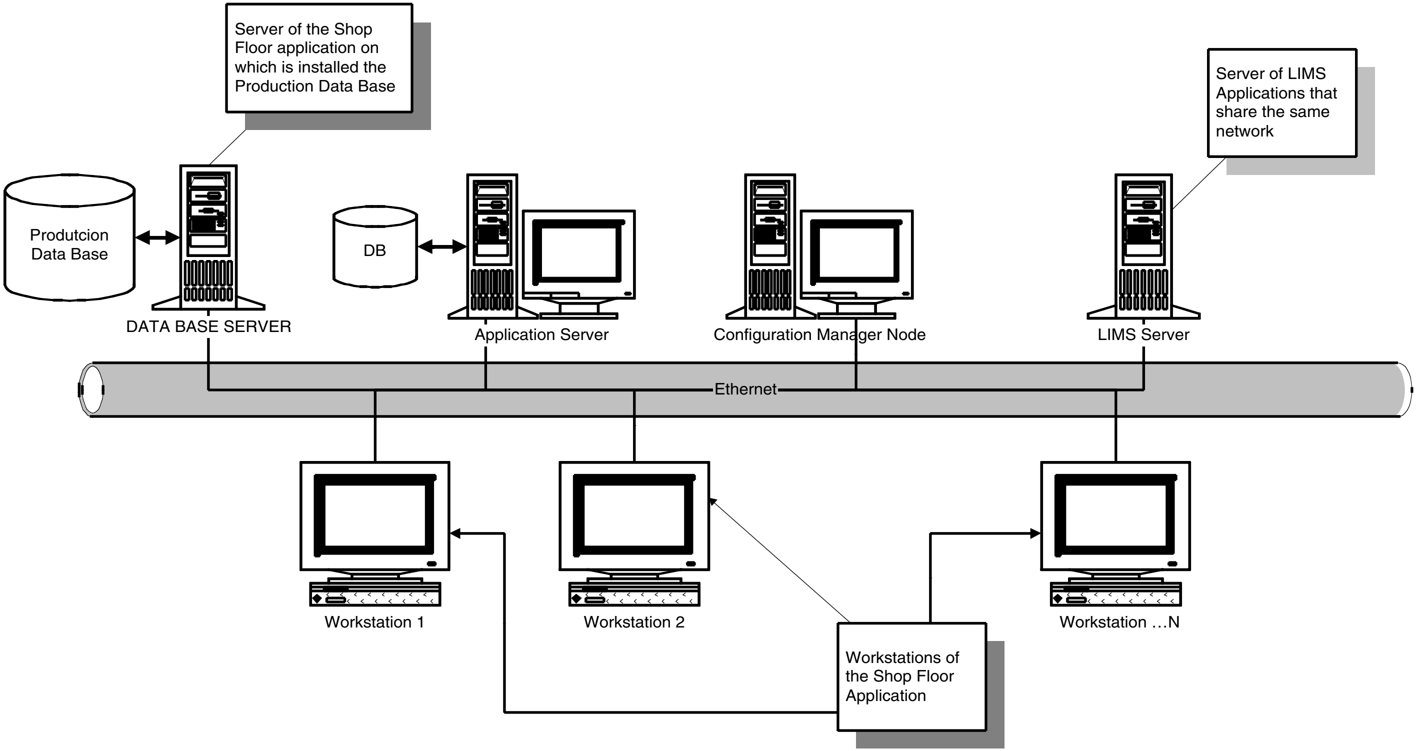

Server of the Shop Floor application on which is installed the Production Data Base
Server of LIMS Applications that share the same network
Produtcion Data Base
DB
DATA BASE SERVER
Application Server
Configuration Manager Node
LIMS Server
Ethernet
Workstation 1
Workstation 2
Workstation …N
Workstations of the Shop Floor Application

FIGURE 38.1 Site Manufacturing Execution System.

- *Data Base Server*
  Unix-based computer with an Oracle database that manages “manufacturing” data (all data used at, and coming from, the shop floor, and all the specification data such as bills of materials, specifications, batch records, etc.)
- *Application Server*
  An OS/2-based computer that manages the software that is distributed to all the application’s workstations. It takes care of communication between workstations and data base server.
- *Configuration Manager Node*
  An OS/2-based computer with a DB/2 database that manages “application data” — i.e., all data specifying the set-up of each workstation and associated security functions.
- *Workstations*
  OS/2-based computers that provide the user interface to the application (e.g., display operating instructions, mimics and alarms/messages to the operators). About 100 workstations are distributed on the shop floor and connect to equipment such as scales and Programmable Logic Controllers (PLCs).

The system is interfaced with a local LIMS and data is exchanged back and forward between the systems. In this example, the GxP nature of the data managed by the MES and LIMS systems and data exchange between the systems mean that both systems and the LAN must be validated.

In this example, and generally speaking in all cases in which more application share the same LAN, it is convenient to proceed with a separate network validation project. In this way all the applications that need to be validated can refer to the validation package of the network, avoiding duplicated work during validation of the IT applications.

A Multisite ERP Application (Enterprise Resource Planning)

A multisite ERP application is presented in Figure 38.2. The ERP application is based on a single SAP R/3 production instance installed on a computer located at the company’s HQ offices. The HQ users access the SAP R/3 application, as well as users from four different manufacturing sites geographically distributed at different locations. All users access the application through the WAN.

There are about 700 application users with an average of 300 concurrent users, and four MESs systems that exchange data with the SAP R/3. Each manufacturing site installed a different MES system based on the topology described in the Site MES Application example above. The ERP application passes GMP relevant data (including production orders, bills of materials, materials allocation in the warehouse, materials consumption data, and materials status) back and forth with the MES systems.

The client/server nature of the SAP R/3 product means that the client-side SAP Graphical User Interface (GUI) must be aligned at the server version. To guarantee that all the 700 users of the application distributed on the five sites receive client version upgrades at the same time and align with the server version concurrently, an application has been installed to automatically distribute the software through the network (Microsoft SMS application).

To validate the application SAP R/3 it is necessary to validate the WAN infrastructure on which the application is built.

NETWORK COMPONENTS

Common network terminology will now be introduced for those who are unfamiliar with the components that constitute a LAN or WAN.

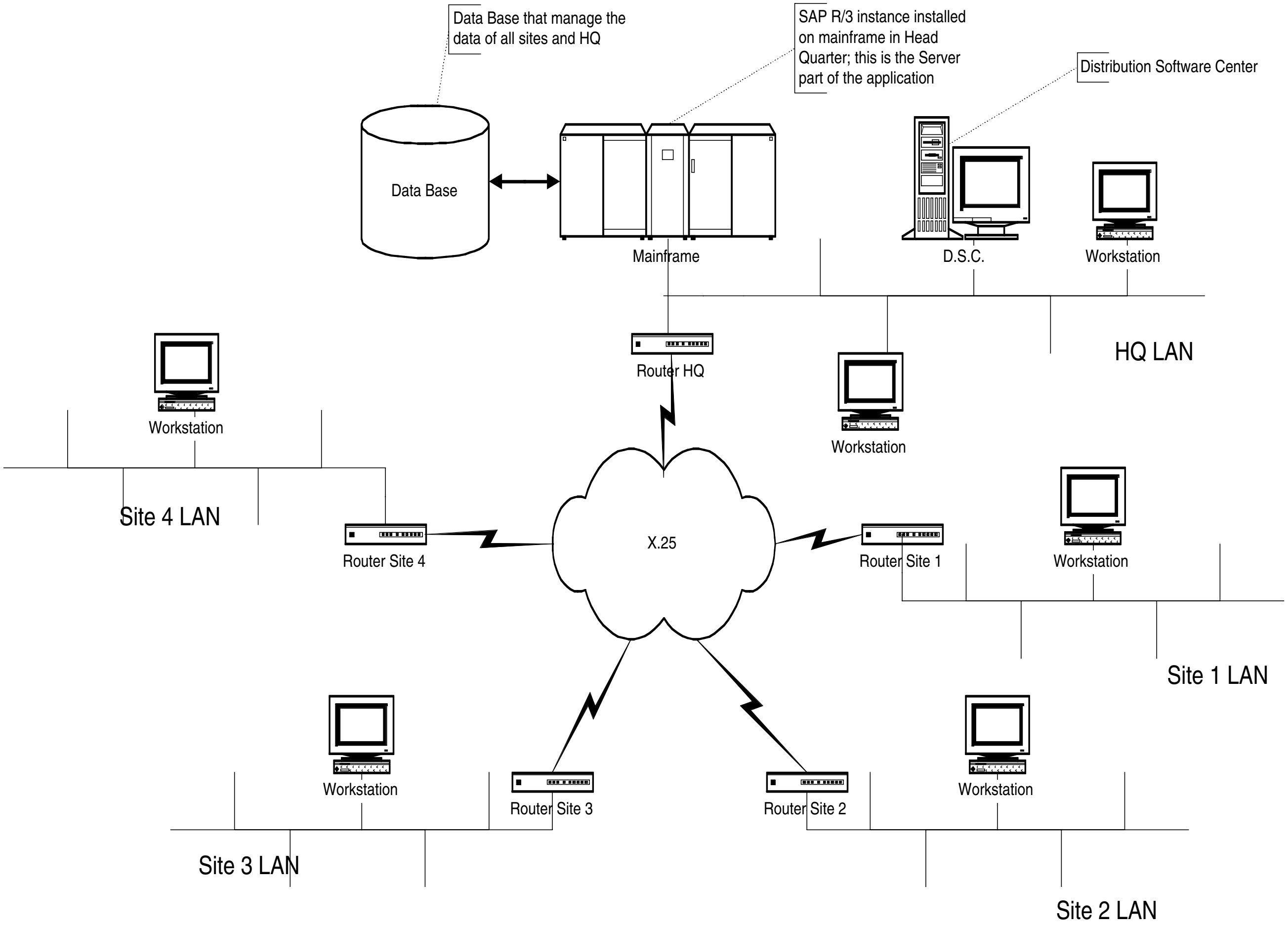


FIGURE 38.2 Example Multisite ERP Network Architecture.

PROTOCOLS

Network protocols are standards that define how computers communicate. A typical protocol defines how computers should identify one another on a network, the form that the data should take in transit, and how this information should be processed once it reaches its final destination. Protocols also define procedures for handling lost or damaged transmissions. Transmissions are sometimes described as "packets" of information. The most common network protocols include TCP/IP, LAT, IPX, AppleTalk, and DECnet.

Network protocols use physical cabling in exactly the same manner, allowing protocols to peacefully coexist. This concept is known as "protocol independence," meaning that the physical network does not need to concern itself with the protocols being carried. The network builder can use any of the protocols supported by an item of equipment. The final choice may depend on personal preference, a defined operating philosophy, or perhaps more arbitrary criteria.

ETHERNET

Ethernet is the most popular LAN technology in use today. Other LAN types include Token Ring, Fiber Distributed Data Interface (FDDI), and LocalTalk. Each has its own advantages and disadvantages. Ethernet strikes a good balance between speed, price, and ease of installation. These strong points combined with wide acceptance into the computer marketplace, and the ability to support virtually all popular network protocols makes Ethernet the perfect networking technology for most computer users today.

The Ethernet standard is defined by the Institute for Electrical and Electronic Engineers (IEEE). IEEE Standard 802.3 defines rules for configuring an Ethernet, and it specifies how elements in a network interact with one another. Networks, equipment, and network protocols that utilize and adhere to the IEEE standard will operate in the most efficient manner.

MEDIA AND TOPOLOGIES

An important part of designing and installing an Ethernet is selecting the appropriate Ethernet medium for the problems at hand. There are four major types of media in use today: ThickWire, Thin Coax, Unshielded Twisted Pair, and Fiber Optic. Each type has its strong and weak points. Careful selection of the appropriate Ethernet medium can avoid recabling costs as your network grows.

Ethernet media can be divided into two general configurations or topologies: "bus" and "point-to-point." These two topologies define how "nodes" are connected to one another. A node is an active device connected to the network, such as a computer or a piece of networking equipment, for example, a repeater, a bridge, or a router.

A bus topology consists of nodes strung together in series with each node connected to a long cable or bus. Many nodes can tap into the bus and begin communication with all other nodes on that cable segment. A break anywhere in the cable will usually cause the entire segment to be inoperable until the break is repaired.

Point-to-point media link only two nodes together. The primary advantage of this type of network is reliability. If a point-to-point segment has a break, it will only affect the two nodes on that link. Other nodes on the network continue to operate as if that segment were nonexistent. Obviously, connecting only two computers together makes for a very limited network. Repeaters may be used to bind groups of point-to-point segments together. (See the section on repeaters for more information on how to connect both point to point and/or bus segments together to make larger, more useful, networks.)

THICKWIRE

ThickWire, or 10BASE5 Ethernet, is generally used to create large "backbones." A network backbone joins many smaller network segments into one large LAN. ThickWire makes an excellent backbone because it can support many nodes in a bus topology and the segment can be quite long. It can be run from workgroup to workgroup where smaller departmental networks can then be attached to the backbone. A ThickWire segment can be up to 500 m long and have as many as 100 nodes attached.

ThickWire, as the name suggests, is a thick, hefty, coaxial cable, and can be expensive and difficult to work with. A thick coaxial cable is used because of its immunity to common levels of electrical noise, helping to ensure the integrity of the network signals. The cable must not be cut to install new nodes; rather nodes must connect by drilling into the media with a device known appropriately as a "vampire tap." Nodes must be spaced exactly in increments of 2.5 m apart to prevent signals from interfering with one another. Due to this combination of assets and liabilities, ThickWire is best suited for, but not limited to, backbone applications.

Thin Coax

Thin Coax, or 10BASE2 Ethernet, offers many of the advantages of ThickWire's bus topology with lower cost and easier installation. Thin Coax coaxial cable is considerably thinner and more flexible than ThickWire, but it can only support 30 nodes, each at least 0.5 m apart. Each segment must not be longer than 185 m. Subject to these restrictions, Thin Coax still can be used to create backbones, albeit with fewer nodes.

A thin coax segment is actually composed of many lengths of cables, each with a BNC type connector on both ends. Each cable length is connected to the next with a "T" connector wherever a node is needed. Nodes can be connected or disconnected at the "T" connectors as the need arises with no ill effects on the rest of the network. The low cost of Thin Coax, its reconfigurability and bus topology make it an attractive medium for small networks, for building departmental networks to connect to backbones, and for wiring a number of nodes together in the same room, such as a computer lab.

Twisted Pair

Unshielded twisted pair cable (UTP) offers many advantages over the ThickWire and Thin Coax media. Because ThickWire and Thin Coax are coaxial cables, they are relatively expensive and require some care during installation. UTP is similar to, if not the same as, the telephone cable that may already be installed and available for network use in your building.

Unshielded twisted pair cables come in a variety of grades, with each higher grade offering better performance. Level 5 cable is the highest most expensive grade, offering support for transmission rates of up to 100 Mbps (megabits per second). This grade of cable is unnecessary for ordinary 10 BaseT applications with 10 Mbps. Level 4 and Level 3 cables are far more popular for current 10 BaseT configurations; Level 4 cable can support speeds of up to 20 Mbps and Level 3 up to 16 Mbps. Level 2 and Level 1 cables are the lowest grades and least expensive wire, designed primarily for voice and low speed transmissions (less than 5 Mbps); these should not be used in the design of 10 BaseT networks.

A UTP, or 10 BaseT Ethernet, is realized with a point-to-point topology. Generally a computer is located at one end of the segment and the other end is terminated in a central location with a repeater or hub. Since UTP is often run in conjunction with telephone cabling, this central location can be a telephone closet or other area where it is convenient to connect the UTP segment to a backbone. UTP segments are limited to 100 meters, but UTPs point-to-point nature allows the rest of the network to function correctly if a break occurs in a particular segment.

Fiber Optic

Fiber Optic, or 10 BaseFL Ethernet, is similar to twisted pair. Fiber optic cable is more expensive, but it is invaluable for situations where electronic emissions and environmental hazards are a concern. The most common situation where these conditions threaten a network is in LAN connections between buildings. Lightning strikes and current loops due to ground potential differences can wreak havoc and easily destroy networking equipment. Fiber optic cables effectively insulate networking equipment from these conditions since they cannot conduct electricity. Fiber optic cable can also be useful in areas where large amounts of electromagnetic interference are generally present, such as on a factory floor.

The Ethernet standard allows for fiber optic cable segments up to 2 km long. Remote nodes and buildings that otherwise would not be reachable with LANs can be brought into the fold.

An investment in fiber optic cabling can be a wise one. As network technologies evolve and demands on the network increase, FDDI and other technologies faster than Ethernet can be run on the same cable, avoiding major rewiring.

Transceivers

Transceivers are used to connect nodes to the various Ethernet media. Transceivers, also known as Media Attachment Units (MAUs), attach to the Ethernet cable and provide an Application User Interface (or AUI) connector for the computer. The AUI connector consists of a 15-pin D-shell type connector, female on the computer side and male on the transceiver side. Virtually all Ethernet-compatible computers provide such an AUI connector. The transceiver is generally attached directly to the computer's AUI connector, or the transceiver may be attached to the computer with a specially shielded AUI cable which must be less than 50 m long. In addition to an AUI connector, many computers also contain a built-in transceiver, allowing them to be connected directly to Ethernet without requiring an external transceiver.

Repeaters

Repeaters are used to connect two or more Ethernet segments of any media type. As segments exceed their maximum number of nodes or maximum length, signal quality begins to deteriorate. Repeaters provide the signal amplification and retiming required to connect segments. Splitting a segment into two or more segments with a repeater allows a network to continue to grow. A repeater connection counts in the total node limit on each segment. For example, a Thin Coax segment may have 29 computers and 1 repeater, or a ThickWire segment can have 20 repeaters and 80 computers.

Ethernet repeaters are invaluable with point-to-point media. As pointed out earlier, a network with only two nodes is of limited use. A twisted pair repeater allows several point-to-point segments to be joined into one network. One end of the point-to-point link is attached to the repeater and the other is attached to the computer with a transceiver. If the repeater is attached to a backbone, then all computers at the end of the twisted pair segments can communicate with all the hosts on the backbone.

Repeaters also monitor all connected segments for basic characteristics necessary for an Ethernet to run correctly. When these conditions are not met on a particular segment, for example when a break occurs, all segments in an Ethernet may become inoperable.

Repeaters limit the effect of these problems to the faulty section of cable by "segmenting" the network, disconnecting the problem segment and allowing unaffected segments to function normally. A segment malfunction in a point-to-point network will generally only disable a single computer, whereas the same problem in a bus topology would disable all nodes attached to that segment.

Just as the various Ethernet media have segment limitations, larger Ethernets created with repeaters and multiple segments have restrictions. These restrictions generally have to do with timing constraints. Although electrical signals inside the Ethernet media travel close to the speed of light, it still takes a finite time for the signal to travel from one end of a large Ethernet to another. The Ethernet standard assumes it will not take more than a certain amount of time for a signal to propagate to the far ends of the Ethernet. If the Ethernet is too large, this assumption will not be met and the network may not perform correctly. Timing problems must not be taken lightly. When the Ethernet standard is violated, packets will be lost, network performance will suffer, and applications will become slow and may even fail.

The IEEE 802.3 specifications describe rules for the maximum number of repeaters that can be used in a configuration. The maximum number of repeaters that can be found in the transmission path between two nodes is four. The maximum number of network segments between two nodes is five, with a further restriction that no more than three of those five segments may have other network stations attached to them (the other segments must be interrupter links which simply connect repeaters). These rules are determined by calculations of maximum cable lengths and repeater delays. Networks that violate these rules may still be functional, but they are subject to sporadic failures or frequent problems of an indeterminate nature. Bridges are recommended for networks where many repeaters are required; they can limit the amount of traffic on each segment and improve performance.

Bridges and Routers

An Ethernet may eventually become too large. It may not be possible to add additional nodes without violating the Ethernet standards, or traffic on the network may cause such a high load that performance suffers. In such cases it may be necessary to split the Ethernet into two or more separate Ethernets with a bridge or a router.

Each of the resulting smaller Ethernets can be expanded with more repeaters and segments because the 802.3 specifications then apply to each of the new Ethernets, not both Ethernets combined. Bridges and routers allow hosts on these two new and distinct Ethernets to talk to one another by using a technique known as “store and forward.”

In store-and-forward devices, packets are gathered off one Ethernet and then saved in memory. When the bridge or router senses that the other Ethernet is available, it transmits the packet. To each of the two Ethernets, the bridge or router looks just like any other host since the bridge or router obeys all the same rules for accessing the Ethernet. Note that the major difference between a bridge/router and repeater is that repeaters do not store packets; they simply clean up the signal on the network and send the signal out all other ports.

Bridges and routers can reduce network load if used intelligently. Bridges listen to all traffic on the network, “learning” where various hosts reside. If a bridge detects a packet on one Ethernet destined to a host on another Ethernet, it will forward the packet to the Ethernet to which the destination host is attached. If a bridge detects a packet on an Ethernet destined to a host on that same Ethernet, it does not bother to forward it. Thus, the second Ethernet is spared from receiving the packet, which was not of any use to any of its hosts, and overall load is reduced. Bridges are protocol independent, they can store and forward packets for any network protocol type without regard for the information they contain.

Bridges read an entire packet before they compare it to their address list; this is done so that short or illegal packets, packets with bad CRCs, or packets with late collisions may be automatically filtered out of the network. Obviously, this means that there will be some small delay factor between the time the bridge finished reading a packet and the time it takes to forward it on. For the benefit of having any bad packets filtered out, most bridge users are willing to incur the very small delay for full packet examination.

A new class of bridging devices, called Ether-switches, offers users another option. Ether-switches read only enough of a packet to determine the source and destination addresses for filtering purposes and then send on the packet at that point. This process speeds throughput but does not filter out illegal or bad packets unless the problem is evident in the first few bytes. The speed advantage of these devices must be weighed against the need to filter.

Routers work in a similar fashion to bridges, except routers are protocol dependent. Routers know about the inner workings of the protocols that they support. This intimate knowledge allows routers to do sophisticated packet forwarding and can provide a great reduction in network traffic by filtering extraneous packets. The price paid for this intelligent forwarding capability is usually additional configuration and cost.

Some routers offer bridging services as a supplement to their primary capabilities; these routers are referred to as "B-routers." B-routers offer such standard bridge features as source/destination address filtering and automatic filtering of bad packets in addition to their protocol-specific routing functions.

Terminal and Print Servers

As their names suggest, terminal servers and print servers support the use of terminals and printers on networks. They support modems and other devices as well. The primary difference between them is that terminal servers are bidirectional devices while print servers have been unidirectional devices, at least as far as data transmissions are concerned. Unlike transceivers, repeaters, or port multipliers, terminal servers and print servers are intelligent devices which have their own network addresses and perform more than just a physical connection or signal forwarding function.

REGULATORY REQUIREMENTS FOR THE NETWORKS

As pharmaceutical manufacturers increasingly integrate manufacturing operations on a national and international basis their reliance on networks increases. Pharmaceutical manufacturers are required to validate these networks. Validation in this context has been defined by the FDA as "establishing documented evidence which provides a high degree of assurance that a specific process will consistently produce a product meeting its predetermined specifications and quality attributes."[1]

The validation computer networks are therefore clearly important, but how do we go about validating them and how much detail is required? The FDA gave the following advice in 1983 on the topic of computer networks supporting manufacturing operations:[2]

If the firm is on a computer network it is important to know:

1. What output, such as batch production records, is sent to other parts of the network
2. What kinds of input (instructions, programs) are received
3. The identity and location of establishments that interact with the firm
4. The extent and nature of monitoring and controlling activities exercised by remote on-net establishments
5. What security measures are used to prevent unauthorized entry into the network and possible drug process sabotage

It is possible under a computer network for manufacturing operations conducted in one part of the country to be documented in batch records on a real-time basis in some other part of the country. Such records must be immediately retrievable from the computer network at the establishment where the activity took place.

In relation to computer networks, the FDA cites clause 180 in CFR 211, which deals with records and reports for manufacturers of finished pharmaceutical products. The concern is that the computer network must maintain the integrity of data passed through the network. This links to the recent issue of 21 CFR 11 in 1997 which deals with electronic records and their security. Further information on this topic can be found in Chapter 16.

The GAMP Forum provides some advice in its latest guide.[4] Basically the same validation methodology should be followed as for other automation and IT systems: categorize software components and follow a "V-Model" life-cycle approach. The GAMP Guide identifies five categories of software:

- *System Software:* Record version of software.
- *Firmware:* Record configuration.
- *Standard Software:* Validate application.
- *Configurable Software:* Consider audit, validate application, and any bespoke code.
- *Bespoke Software:* Audit supplier and validate complete system.

Networks are largely made up of standard components (system software, firmware, and standard software); there is little bespoke programming other than configuration and perhaps some specialist interfaces. Supplier audits for Commercial Off-The-Shelf software are not normally required, as discussed later.

The application of the FDA and GAMP guidance is discussed in the following sections of this chapter and is based on the practical experience of validating networks within an international pharmaceutical manufacturing company.

Examples of Network Inspection Findings

The Wide Area Network … is used to connect network applications to local area networks. The [AAAA] and the [BBBB] run both the [XXXX] and the [YYYY] network application at each site … Both the [AAAA] and the [BBBB] documentation were not included in the [XXXX] and [YYYY] validation efforts and therefore lacked adequate documentation controls.

The firm utilizes a Wide Area Network (WAN) to connect all Local Area Networks (LANs). The WAN is not validated as described below:

- The Quality unit has failed to ensure that procedures are in place, which define all system definition documentation, which must be maintained for the WAN.
- The Quality unit has failed to ensure that complete WAN system definition documentation is included in WAN documentation. For example, the Quality unit has failed to ensure that the WAN validation documentation includes WAN site diagrams.
- When requested, the firm could produce no approved WAN site diagrams. The Quality unit has failed to put in place procedures which define that WAN site diagrams are maintained.

Local Area Networks (LANs) connect local manufacturing, testing and warehouse departments at each site on the WAN …. The LAN is not validated as described below:

- The Quality unit has failed to put in place procedures which ensure that LANs for each site are controlled.
- Complete system definition documentation has not been maintained. For example, the firm produced no approved LAN diagrams identifying all sites/equipment on the LAN.
- LAN site listings have not been maintained or controlled; equipment listings, which were presented as system definition documentation for the XXXXXX LAN, were not procedurally defined or controlled.

The network … which can only support up to four [XXXX] systems, had up to five [XXXX] systems connected. There was no validation showing this configuration to be acceptable.

… original reports … which were sent via electronic mail to the Quality Assurance Management differed significantly from the versions included in the Quality Assurance Management's official reports.

To date the firm has failed to generate and approve sufficient design control documentation for complete definition of the network (i.e., high-level diagrams identifying all sites/equipment …)

The firm has failed to document all sites, departments, or connections on the network using the [application]. The … program communicates across the network with various other programs external to the [system]. The firm has failed to document external program interfaces … in controlled documentation …

The firm's IT staff currently uses the Visio database as documentation for the network. The Visio documentation is maintained as electronic records and not in hardcopy. These electronic records are not reviewed or approved (i.e., no electronic signatures of review or approval).

The … program runs across the LAN … The firm presented a wiring diagram in support of the validation status for this LAN. The diagram provides a graphical representation of the current I/O wiring (node lists) for each of the various devices on this LAN. Regarding this diagram:

- The diagram lacks review by the Quality Unit.
- The diagram has not been maintained following established document control procedures.
- The diagram has been produced using I/O data contained within the nonvalidated Excel node list database, which … is not a controlled record.

There was no validation data to show that [data] could not be inserted by the corporate WAN into the LAN.

Incremental and full backups of lab data/results were done from the WAN. There were no validation data to demonstrate that an authorized user of the corporate WAN did not have access to analytical data on the laboratory's LAN.

The HP OpenView computer system is currently used by the firm's IT staff to manage the computer network. The firm currently has no procedures defining the use of the HP OpenView system.

VALIDATION STRATEGY FOR THE NETWORKS

To define a validation strategy we have first to consider the current status of the system. Basically, if the system has already been installed, the validation will be *retrospective*. Otherwise, if it is a new network we can proceed with *prospective* validation. The chapter will consider a prospective validation; however, most concepts can be used also in case of retrospective validation. It should be noted here that retrospective validation is usually a much more expensive and timely task compared to prospective validation. It has been suggested that retrospective validation can be in excess of five times more expensive than prospective validation.[3]

Now that the kind of validation that we are going to perform on our system is defined — prospective or retrospective — it is necessary to clarify the scope of the system. Agreeing that the scope is extremely important for two reasons. First, we can adopt the appropriate variant of the V-Model life cycle, depending on the use of different categories of software. Second, it makes it possible to establish what is part of our system and what is not part of our system — substantially, what we are going to validate and what we do not validate.

What is in and out of scope must be clearly visible to the user. If a network server or network control system are, for example, determined to be out of the scope of one project, then they must be covered by another validation exercise. Other network scope issues might include who is responsible for network interface cards or who is responsible for the firewalls between interconnected networks. It is quite common for networks to fall between projects and not to be validated until a regulatory inspection identifies this as a GxP nonconformance. It is in the interest of the pharmaceutical manufacturers to avoid the embarrassment of a regulator identifying absent validation and applying any consequential official warning or sanction. Determining who is responsible for validating a particular network can be assisted by identifying what data are transferred over the network and who is responsible for that data.

VALIDATION AND SYSTEM LIFE CYCLE

A life-cycle approach should be adopted when validating networks. In a simple form this might consist of a "cascade" development methodology, forcing the definition and approval of each document produced in the prior phase before proceeding with the next phase. The cascade approach is certainly applicable to the development of network systems but more usually an "incremental" approach is adopted. The incremental approach facilitates the construction of complex network systems from configurable software and hardware equipment. Configurable packages provide a means of easily modifying a network system by reconfiguring software and/or equipment to reflect any changing requirements without the need for the development of an entirely new replacement network systems.

Figure 38.3 shows the phases within a cascade approach and incremental approach. Each box presents a different phase of the life cycle. The descending arrows mark the passage between one

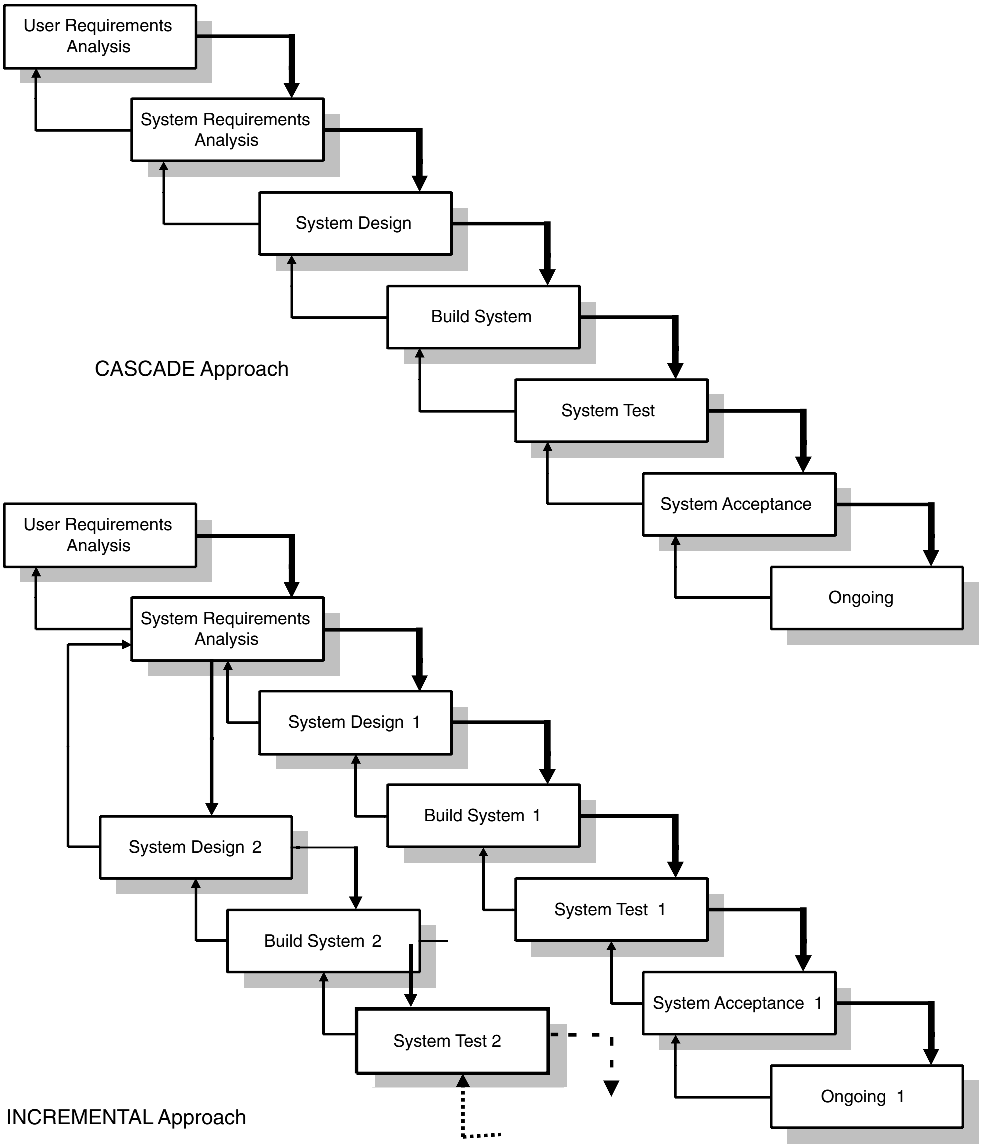


FIGURE 38.3 Cascade and Incremental Approaches.

phase and the next (enabled only after the approval of "deliverables" related to the prior phase), the ascending arrows identify the system acceptance step by step (after the execution of the tests related to the specific phase). Note that the same life-cycle phases are still valid inside those different development methodologies. As noted by other practitioners, documents and activities can be combined.[3] This is especially so where extensive use is made of COTS products.

The V-Model commonly used in the pharmaceutical and healthcare industry for computer systems validation is presented in Figure 38.4. The model illustrates the cascade approach and the relationship between specifications and testing. This V-Model can now be developed to fit the incremental approach (see Figure 38.5).

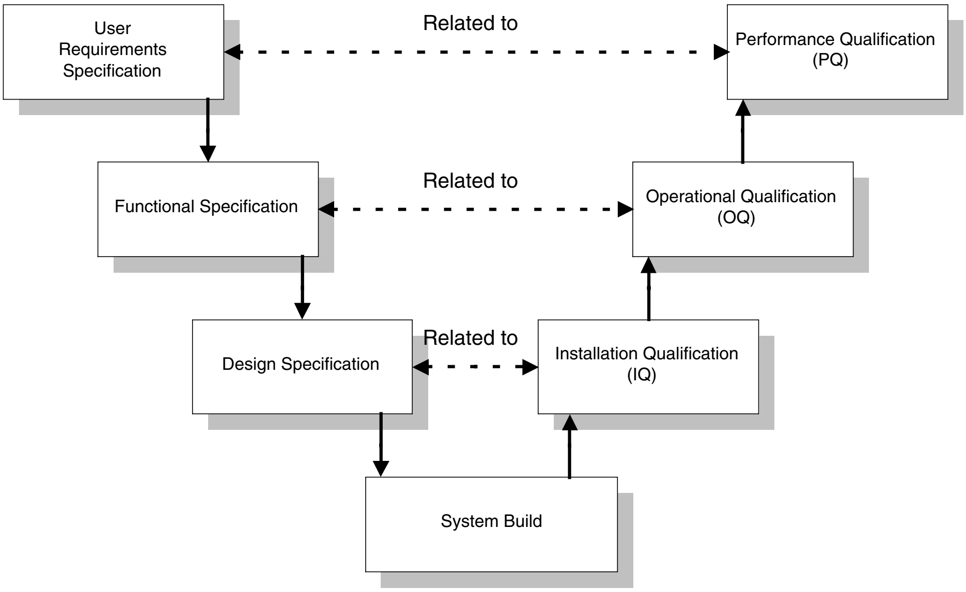

User Requirements Specification
Related to
Performance Qualification (PQ)
Functional Specification
Related to
Operational Qualification (OQ)
Design Specification
Related to
Installation Qualification (IQ)
System Build

FIGURE 38.4 Relationship between Specifications and Testing.

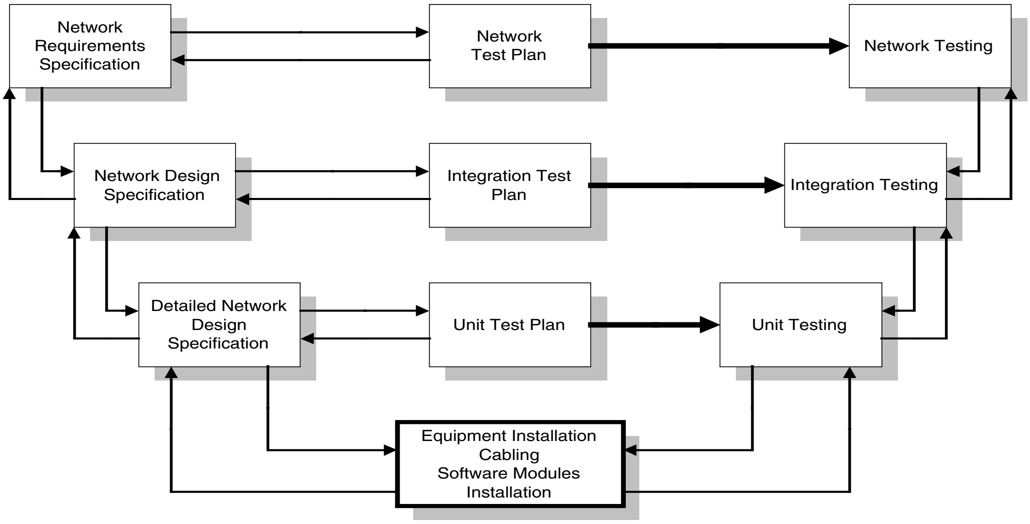


FIGURE 38.5 Relationship between Specifications and Testing in Incremental Approach.

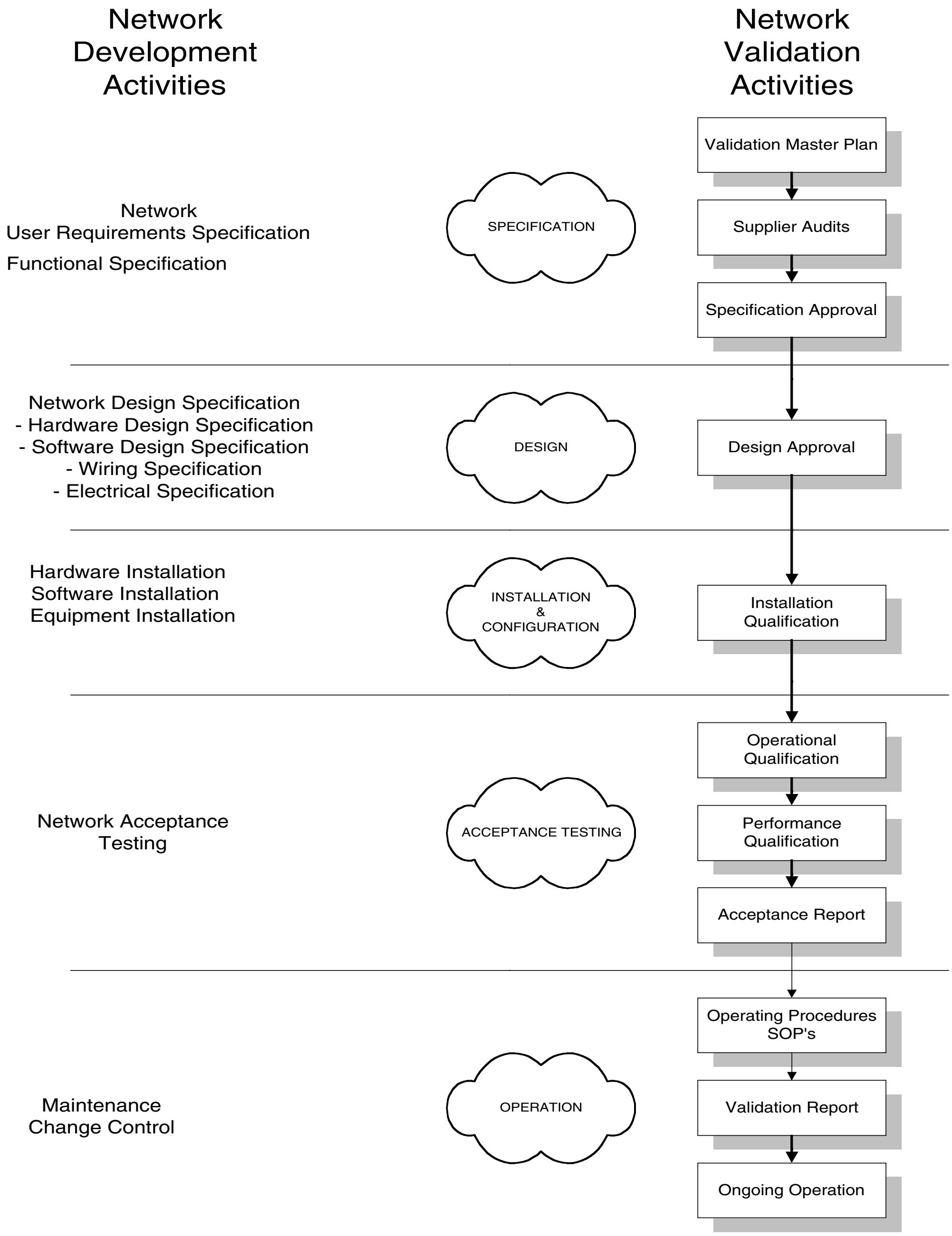

Network Development Activities
Network Validation Activities
Validation Master Plan
Network User Requirements Specification
Functional Specification
SPECIFICATION
Supplier Audits
Specification Approval
Network Design Specification
- Hardware Design Specification
- Software Design Specification
- Wiring Specification
- Electrical Specification
DESIGN
Design Approval
Hardware Installation
Software Installation
Equipment Installation
INSTALLATION & CONFIGURATION
Installation Qualification
Operational Qualification
Network Acceptance Testing
ACCEPTANCE TESTING
Performance Qualification
Acceptance Report
Operating Procedures SOP's
Maintenance
Change Control
OPERATION
Validation Report
Ongoing Operation

FIGURE 38.6 Validation Activities for a Network.

The validation activities that should be executed during a development life cycle of a network are described in Figure 38.6. The right side of the diagram lists the network development activities, and for each activity or group of activities, the related validation activities.

Figure 38.7 identifies the documentation that should be produced during the development of a network system. Of course, the specification and testing stages could be different depending on complexity of network design. For instance, the network wiring and electrical design specification (Document A), the network hardware design specification (Document B), and the network software

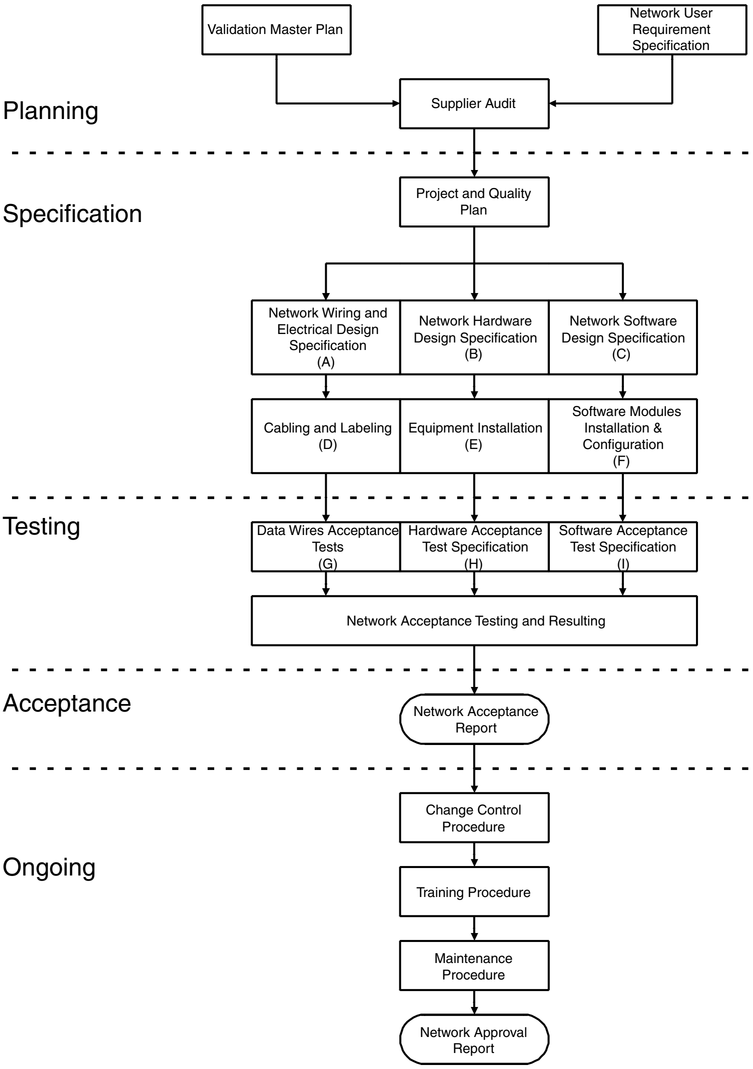

Validation Master Plan
Network User Requirement Specification
Planning
Supplier Audit
Specification
Project and Quality Plan
Network Wiring and Electrical Design Specification (A)
Network Hardware Design Specification (B)
Network Software Design Specification (C)
Cabling and Labeling (D)
Equipment Installation (E)
Software Modules Installation & Configuration (F)
Testing
Data Wires Acceptance Tests (G)
Hardware Acceptance Test Specification (H)
Software Acceptance Test Specification (I)
Network Acceptance Testing and Resulting
Acceptance
Network Acceptance Report
Change Control Procedure
Ongoing
Training Procedure
Maintenance Procedure
Network Approval Report

FIGURE 38.7 Documentation in the Life Cycle.

design specification (Document C) should be merged in one document divided into three sections. The cabling and labeling specification (Document D) in reality should be a set of drawings that shows how the data and electrical cables are physically located. The cabling and labeling specification will include piping/ducting layouts and the location of the network equipment. The equipment installation details (Document E) and software module installation and configuration details (Document F) could also be merged in one document. The data wires acceptance testing, hardware acceptance testing and software acceptance testing (Documents G, H, and I), could again be merged in one document that collates all the tests that must be executed on the network. Separate documents can nevertheless be appropriate when project logistics require the installation and testing of electrical supplies, network equipment, and network software at different times.

SPECIFICATIONS OF NETWORKS

The following specifications are based on the GAMP Guide[4] available from ISPE.

Production of a Local Area Network Design Specification

The contents of a LAN Design Specification are outlined below. It is strongly recommended that the design specification include a map of the network and the systems it interconnects.

Introduction Section

This section shall contain information on who produced the document, under which authority, and for what purpose. The relationship to other documents should also be reported.

Design Overview Section

This section should briefly describe the design of the LAN. It should introduce the basic concepts used in the design, and discuss the rational of the proposed solutions. The following subsections should be included:

- Site/Area Description
  It may contain a drawing of a simplified site layout (area on which the LAN will be installed), identifying building and area classifications (e.g., manufacturing units, warehouse, laboratories, etc.). For each area a description of the other network systems should be available (areas or building that require wireless sub-LAN, or areas that are required to sustain a large number of high bandwidth connections simultaneously because they are dedicated to videoconferencing and multimedia devices, etc.).
- A summary of peculiarities of a site LAN environment
  - Temperature
  - Humidity
  - External interference
  - Physical security
  - Radio-frequency, electromagnetic, and UV interference
- The design/solution
  Demonstrate how design requirements meet or do not meet the URS. Include how operational environment needs are fulfilled.

LAN Architecture Section

This section shall describe briefly the design of the LAN.

- **General Description**
  This subsection shall briefly describe the design of the LAN. It may include a drawing of logical schema of the LAN with the major network components and how they interact with the environment. A description of these topics should be listed:
  - Connectivity
  - Network redundancy
  - Routing capability
  - Equipment/device naming conventions
- **Detailed Description**
  This subsection shall contain a list and description of all LAN components. Those components may be classified as:

  Higher Components: Components that falls into Application Presentation and Session OSI layer, e.g., NOS (Network Operating Systems). All application services should be listed.

Network Components: Components that falls into Session and Transportation OSI layer, e.g., Routings, Transportation Protocols.

Physical Components: Components that falls into Data Link and Physical layers, e.g., Terminal servers, hub management cards, etc.

Cabling Infrastructure: List of all wiring standards that must be taken in account, e.g., wiring standard (EIA/TIA-568). In this subsection may be specified, for each area, the related wiring concept, e.g., horizontal wiring, backbone wiring, working area wiring, etc. All types of cables that must be used should be listed. The following requirements should be considered:

- Screening and shielding
- Labeling
- Tools and equipment

Electrical Supplies: In this subsection all electrical supply requirements for the LAN will be addressed. Elements to be considered include:

- Earthing
- Loading
- Filtering
- Uninterruptable power supply (UPS)
- Disconnection by fault
- Electrical safety

Network Management: In this subsection the network control system and its functions should be defined, if part of the design.

Security: In this subsection the security requirements, physical and logical access, should be defined.

LAN Detailed Design Sections

- **Exact configuration of each component of the network**
  In this subsection the exact number of each component of the network should be defined. It shall consider those elements for each hub/working group:
  - Reference with geographical location on layout
  - Number type of equipment/cards
  - Number/type of connections gates available
  - Hardware/software parameter to be used
  - Address for each network component
- **A series of drawings that shows the exact location of all network equipment and cables**
  In this subsection these drawings should be attached, as a minimum:
  - Complete layout of the site showing the backbone cable path and location of main network objects, e.g., hub's end working groups.
  - A detailed drawing showing, for each area building, the location of each network component and the cable path from the hub to the faceplate on the walls.

Production of a Wide Area Network Design Specification

Below is the definition of which sections shall be included in the WAN Design Specification. Due to the particular objective of the design, the diagrams and drawings are strongly recommended to define WAN.

Introduction Section

This section shall contain information on who produced the document, under which authority, and for what purpose. The relationship to other documents should also be reported.

Network Overview Section

This section shall contain a general description of the WAN. It may describe, eventually, different implementation phases of the WAN, the interconnection points, and their geographic location. This section shall contain the following information:

- Connectivity
- Services
- Access Points
- Security and Network Management
- Network Dimensioning and Performance

The WAN Architecture Section

This section should describe briefly the design of the WAN.

- **General Design**
  It should introduce the basic concepts used in the design of the WAN. It may include a drawing of logical schema of the WAN with the major network components and how they interact with the environment. A description of these topics should be listed:
  - Network Features and Characteristics
  - Connectivity
  - Network Redundancy
  - Dynamic Routing Capability
- **Detailed Network Design**
  This section should contain a description of each component of the network, both hardware and software. It shall describe how those components are connected and configured; these topics may be listed:
  - Hardware and Software Requirements
  - Protocol Supported
  - Configuration

Access Points Architecture Section

This section should describe the access points architecture, in terms of users and technology, that must be used to connect to the WAN. It may address these topics:

- Different Remote Access Categories
- Cabling Configuration

Application Services Section

This section should describe the application layer services supported by the WAN. It shall describe the follows services, e.g., Electronic Messaging Systems, Simple Mail Transfer Protocol (SMTP), etc.

Network Management Section

This section should describe all the management functionality available on the WAN such as:

- Configuration Management
- Performance Management
- Fault Management
- Security Management

Security Controls Section

This section should describe all security issues involved in the operation of the WAN. It may address the following topics:

- Authorization Granting of Rights to Access Resources
- Access Control Mechanisms
- Access Control Policies
- Routing Control Mechanism
- Security Protocols

QUALIFICATION OF NETWORKS

A network test specification must be written and approved before testing can begin. The test protocol should not introduce any new specification details but, instead, reference the network design specification. Raw data should be collected during testing to provide evidence of test outcomes. This evidence should be retained with the test specification and a test report summarizing the results of individual tests, listing test discrepancies and failures, and identifying corrective actions and any necessary or recommended repeat testing. It is this test report that will conclude whether or not the network is fit for purpose.

Test Protocol for Network Qualification

The contents of a network test specification are outlined below.[2]

Introduction Section

This section should reference the validation plan, network design specification, and the validation procedure being used for testing.

Scope Section

Define the scope of the qualifications program to be undertaken, including:

- Visual check of components:
  - Against design specification
  - Against standards
  - Against statutory requirements
  - In accordance with manufacturers' instructions
- All equipment and materials undamaged, clean, new, and correctly installed (refer to installation records)
- Any requirements for hazardous areas are met:
  - Capacity testing
  - Software versions checked
  - Electrical supply and interference testing
  - Manufacture diagnostic testing
  - Power on-off testing

- Operational environmental
- Configuration/system testing (each user port tested for connection to network)

Test Plan Section

Describe the overall testing philosophy. The following issues should be addressed:

- Specific areas not tested, and why
- Any logical grouping or ordering of tests
- Personnel required for test groups

Testing Prerequisites Section

- Hardware requirements (systems(s) set-up)
- Test equipment requirements (including simulation tools)
- Test data requirements
- Reference document (such as operating manual, vendor data sheets, etc.)

Test Procedure Section

Details of all the test cases. Each test case should be on a separate page. The test case should collectively provide 100% coverage of the network design specifications.

PRACTICAL ISSUES

Network validation is only necessary where that network is used to convey controlling instructions or GxP data.[5] This said, it is important to recognize that networks will often be installed without a validation requirement and then later in their life be requested to support GxP applications. It can be very difficult to retrospectively validate a network. It is much better from the outset of installing a network to establish good IT practice and keep appropriate records detailing work done. Then, if the use of a network changes, there is some documentary evidence in place that can be used to supplement validation. In particular, good IT practice for networks should include:

- Configuration Management
- Installation Qualification
- Change Control

The content of networks specification and test protocols will depend on the complexity of the system. For small systems it may be possible to incorporate the design of the network as a special section within the Systems Functional Specification, and similarly incorporate the network test cases as a special section within the systems. Operational Qualification Supplier audits for network hardware and software components are not usually necessary as these are normally industry standard. Further discussion in regard to supplier audits can be found in Chapter 7.

Once a network is validated, care must be taken to maintain its validation status. Tests should be formally recorded when new systems are connected to a validated network even if the new systems themselves do not require validation. The addition of a new system to a network will alter traffic loading and hence could compromise the responsiveness of the network. Security, too, could be compromised. Networks often employ firewalls to protect sensitive portions of a network from interference or abuse.

Another issue that has practical implications on validation is the use for third parties to maintain and support networks. In these circumstances contracts must be established with the suppliers defying what procedures will be used by contract staff and what records will be maintained and/or handed over to the pharmaceutical manufacturer. The pharmaceutical manufacturer, not the supplier, is accountable to the GxP regulatory authorities for validation. The operational terms of the contract are usually defined in a document called a Service Level Agreement (SLA).

U.S. Code of Federal Regulations Title 21 Part 11 on electronic records impacts the use of networks. A detailed assessment of this regulation is outside the scope of this case study. A discussion on 21 CFR Part 11, however, can be found in Chapter 15.

REFERENCES

1. FDA (1995), *Glossary of Computerized and Software Development Terminology*, U.S. Food and Drug Administration, Rockville, MD, August.
2. FDA (1983), *Guide to Inspection of Computerized Systems in Drug Manufacturing*,Technical Report, Reference Materials and Training Aids for Investigators, U.S. Food and Drug Administration, Rockville, MD.
3. Benson, J., Smith, M., Mole, D., and McDowall, R.D. (2002), Computer Network and Infrastructure Qualification and Validation of Associated IT Applications: A Case Study, *Journal of Validation Technology*, 9(1), November.
4. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (www.ispe.org).
5. Wingate, G.A.S. (2000), *Validating Corporate Computer Systems: Good IT Practice for Pharmaceutical Manufacturers*, Interpharm Press, Buffalo Grove, IL.

39 Case Study 21: Web Applications

Ludwig Huber, Agilent Technologies

CONTENTS

Web applications are increasingly used for all types of businesses including healthcare. Two main applications are the World Wide Web for all types of on-line transactions and e-mails for exchanging messages with and without attachments. An example where the Internet can play an important role is shown in Figure 39.1. A pharmaceutical company outsources part of its clinical studies or laboratory analyses to a contract laboratory. The sample is sent by FedEx to the contract laboratory, analyzed, and the data sent back to the sponsor by e-mail with reports attached.

Other examples for using the intranet or Internet in the healthcare business are:

- Release of batch approvals
- Approval and release of validation life-cycle checkpoints and validation and reports
- Electronic artwork transfer
- Remote approval of Certificates of Analysis at contract laboratories
- Updates, exchange and approval of training records and SOPs
- Administration of electronic patient records
- Billing Information exchange between healthcare provider and insurance
- Tele Medicine (remote surgery, diagnostics, imaging)
- Drug prescription online

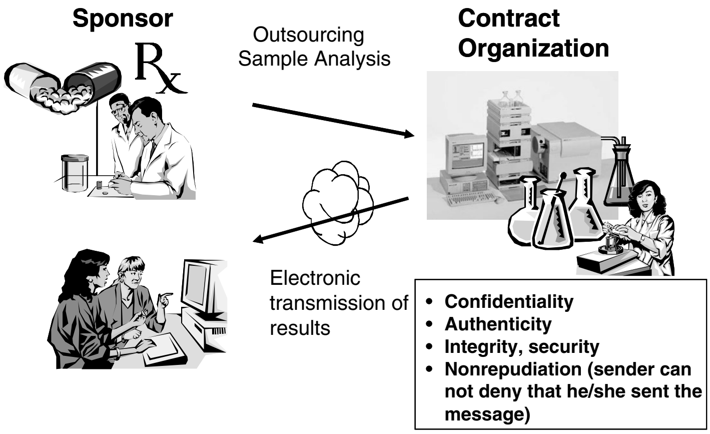

Sponsor
Rx
Outsourcing Sample Analysis
Contract Organization
Electronic transmission of results
• Confidentiality
• Authenticity
• Integrity, security
• Nonrepudiation (sender can not deny that he/she sent the message)

FIGURE 39.1 Example — Case Study.

- Electronic patient card
- Centralized and local patient data administration

When transporting clinical studies or any other data as mentioned above, data traffic needs to adhere to:

- *Confidentiality:* The contents of the data should only be accessible by authorized persons.
- *Integrity:* The data should be the exactly same at source and destination computers.
- *Authenticity:* The authenticity of the sender of the data must be guaranteed.
- *Nonrepudiation:* Sender and recipient of the data cannot deny sending/receiving the data.

FDA regulation 21 CFR Part 11 on electronic records and signatures requires records to be trustworthy,[1] a word that combines all requirements as mentioned above.

The Internet by its nature is an insecure and unreliable environment and therefore without special precautions is not compliant with the above-mentioned requirements. For example, the TCP/IP Internet communication protocol was not originally designed to accommodate security commands. Almost daily we hear in the news about hackers, viruses, scam artists, and on-line predators. For example, the Open Web Application Security Project (OWASP) published a 27-page report about the top ten Web application security vulnerabilities.[2] They include:

1. Unvalidated parameters
2. Broken access control
3. Broken account and session management
4. Cross-site scripting flaws
5. Buffer overflows
6. Command injection flaws
7. Error handling problems
8. Insecure use of cryptography
9. Remote administration flaws
10. Web and application server misconfiguration

Should we neglect the advantages of the Internet because of the many problems that have been reported? The answer is "no."

The FDA recognizes the increasing use of the Internet and gives recommendations on how it can be used in an FDA-regulated environment. This has been spelled out in an FDA draft guidance on validation:[3]

> *We recognize the expanding role of the Internet in electronic recordkeeping in the context of part 11. Vital records, such as clinical data reports or batch release approvals, can be transmitted from source to destination computing systems by way of the Internet.*[3]

There are a lot of security tools and technology available and offered as part of browser software, such as Microsoft Internet Explorer and ISPs. There are also tools available that help comply with other requirements such as authenticity, data integrity, accuracy of data transfer, and nonrepudiation. This is also recognized by the FDA:

> *The Internet can nonetheless be a trustworthy and reliable communications pipeline for electronic records when there are measures in place to ensure the accurate, complete, and timely transfer of data and records from source to destination computing systems.*[3]

Availability of tools does not necessarily mean that everybody takes advantage of them. This is where this case study will help. Its aim is to give guidelines on the steps to take to make the use the Internet trustworthy as required by regulations, such as the FDA's 21 CFR Part 11. Before we do this we would like to introduce readers to some of the basic terms and technologies for better understanding. These include open vs. closed systems as defined by 21 CFR Part 11, FTP/IP protocols, cryptography, digital signatures, digital certificates, Public Key Infrastructures (PKI), and Secure Multipurpose Internet Mail Extensions (S/MIME). We cannot go into too much detail without exceeding the scope of a book chapter. However, there is a lot of reference material available. Danda[4] gives a very good overview on security, privacy, and data integrity for on-line applications in a textbook.

Extensive information on hash calculations can be found on RSA's Web site.[5] A working group of the Institute of Electrical and Electronics Engineers (IEEE) has developed standard specifications for public key cryptography.[6] The American Bar Association has developed guidelines for digital signatures[7] and information on Public Key Infrastructure can be found in references,[8–12] for example, from RSA[9] and the PKI Forum.[12] Guidelines on the validation of computerized systems came from GAMP[13] and from Huber.[14] The qualification of network infrastructure and validation of networked systems is documented in Huber.[15]

Because of the dynamic nature of this topic references may have to be updated frequently. Therefore, we recommend readers of this chapter to visit a specific Web site that has been established especially for Internet compliance: www.networkcompliance.com/internet. The site includes links to tools and publications as well as reference materials such as SOPs, templates, and checklists. which help readers get a better and more in-depth understanding of the topic. Frequent updates ensure that visitors can always learn about the most recent technologies.

OPEN VS. CLOSED SYSTEMS

The Internet is a classic example of an open system in the definition of 21 CFR Part 11: "Open system means an environment in which system access is not controlled by persons who are responsible for the content of electronic records that are on the system."[1] As illustrated in Figure 39.2. ISPs have access to data, which means the persons who are responsible for the content cannot control access to any data transferred through the Internet.

> *Section 11.30 of Part 11 specifies requirements for open systems: Persons who use open systems to create, modify, maintain, or transmit electronic records shall employ procedures and controls designed*

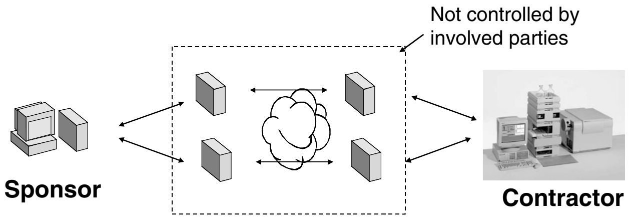


FIGURE 39.2 The Internet as an Open System.

> *to ensure the authenticity, integrity, and, as appropriate, the confidentiality of electronic records from the point of their creation to the point of their receipt. Such procedures and controls shall include those identified in Sec. 11.10, as appropriate, and additional measures such as document encryption and use of appropriate digital signature standards to ensure, as necessary under the circumstances, record authenticity, integrity, and confidentiality.*

Encryption and digital signatures are the keywords here. We will elaborate further on these two techniques later in this chapter.

DATA TRANSFER THROUGH THE INTERNET

Data are transmitted through the Internet by using TCP/IP communication protocols. A specific function of TCP/IP protocols is so-called packet switching. When a data file is sent through the Internet it is not sent in one piece. Instead, the file is broken into packets that can be routed separately through the Internet, a process that is called packet switching. This is illustrated in Figure 39.3.

At the sending computer, the files are broken into packets. They are sent through a LAN or modem and gateway through routers to the receiving computer.

The receiving computer reassembles the packets into a single file that is identical to the original file. The advantage of this concept is that each packet can find the fastest way through the Internet. When one way becomes overloaded, packages broken down from one file can be directed through different lines.

Computers on the Internet are identified through IP addresses. IP addresses of the sending and receiving computers together with some other information are included in a header created for each packet. This is the reason why all packets from one data file find their way to the same computer

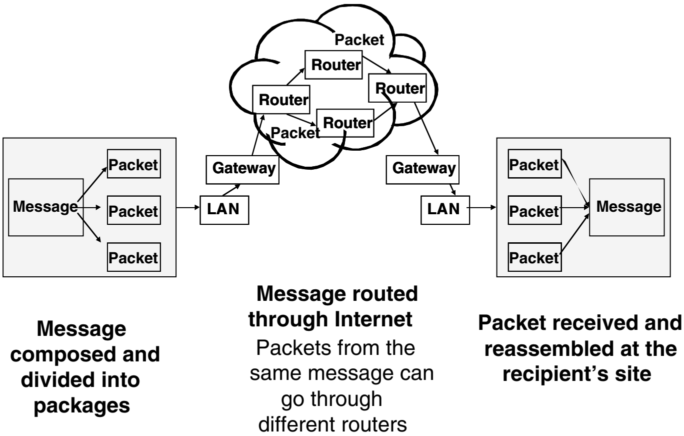

Packet
Router
Router
Router
Router
Packet
Message
Packet
Packet
Packet
Gateway
LAN
Gateway
LAN
Packet
Packet
Packet
Message
Message composed and divided into packages
Message routed through Internet
Packets from the same message can go through different routers
Packet received and reassembled at the recipient's site

FIGURE 39.3 Data Transfer through the Internet.

and can be reassembled again. The routers read the message headers and forward the packets to either another computer or to the gateway at the destination site.

CRYPTOGRAPHY AND DIGITAL SIGNATURES

As already mentioned, the Internet is a classic example of an open system. TCP/IP communication protocols were not designed for security, and file information can be accessed by the ISP personnel. To prevent the information from being read by people who are not authorized to do so, the information must be scrambled. The tool to help do this is cryptography, a word that comes from the Greek for "secret writing." Only a person who knows how to unscramble the information can read and understand it. Cryptography technology is used for on-line shopping to make credit card information invisible. It has been used over the last 2000 years mainly by the military to protect instructions sent from headquarters to the front. The principle is very simple and can best be illustrated by an example.

If we want to encrypt the word "test," all we have to do is to convert each letter to the next letter in the alphabet. After doing this the word reads "uftu" and unless the reader knows the encryption mechanism, he/she does not understand the meaning. Of course, in practice the encryption algorithms are much more complex, otherwise they would be easy to guess.

We can use this example to explain some terms used in cryptography:

- "Test" is the plain text.
- "Ciphertext" is the text after encryption, in our example "uftu."
- "Cipher" is the cryptographic algorithm; in our example we use subsequent letters in the alphabet.
- "Key" is the incremental step; in our example it was one.

Cryptography can be divided into two groups: symmetric and asymmetric encryption. In symmetric encryption the sender and receiver use the same key. This method is very fast but the sender needs to send not only the message but also the key, otherwise the receiving party cannot read the message.

Asymmetric encryption is also called "public key encryption." Two keys are required: a private key and a public key. Usually the sender encrypts the data with the public key and the receiver decrypts the data with a private key but it can also be the other way around. Public keys are frequently located on the Internet. Private keys are located in a secure area of the owner's computer.

HASH FUNCTIONS TO ENSURE DATA INTEGRITY

Hash functions are used to check data integrity. A hash function is an algorithm that takes a variable-length string of any length as the input, and produces a fixed-length binary value (hash) as the output (fingerprint). The tricky part is to make this process irreversible, that is, finding a string that produces a given hash value should be very difficult. It should also be difficult to find two arbitrary strings that produce the same hash value. Because of its irreversibility this is called a one-way hash. Examples are MD4, MD5, and SHA-1. MD4 and MD5, invented by Ron Rivest for RSA Security, Inc., produce 128-bit hash values. SHA-1 (also known as simply SHA) was designed by NIST and NSA and produces 160-bit hash values. Hash calculations are not only used for Internet applications but also for verifying accuracy of file copies and proper installation of software packages from CDs to hard disks.

The sender's computer calculates the hash value and attaches the value to the message. The receiving computer uses the same algorithm, recalculates the hash value, and compares the result with the value as attached to the message. Obtaining exactly the same hash means the file is the same. The probability that two different records generate the same message digest is one in 10^{87}, which is quite high.

Neither cryptography nor hash values ensure authenticity of the sender over the Internet. To do this we need digital signatures that combine one-way hash calculations and cryptography using a person's private key. A digital signature is an encrypted message digest that is appended as ciphertext to a message. The receiver uses the sender's public key to decrypt the message and calculates the message digest using the same hash function that the sender was using. When correctly implemented, digital signatures are equivalent to handwritten signatures on paper and the sender cannot deny legal responsibility for the content of the message.

DIGITAL CERTIFICATES OF PUBLIC KEYS

With cryptography, hash calculations, and digital signatures as described above, there is still one open question: How can you be sure that the person whose name is in the message is really the sender? To ensure that a public key is assigned to an individual or organization, we need certificates. These verify the identity of a person. They are also the basis for secure electronic transactions. The information that should go into a digital certificate is standardized in a protocol called X.509. The information usually includes a person's name, the public key, information on the CA, the expiration date, and a serial number. When implemented correctly, both sender and receiver can trust each other. The trust typically is based on a third part, the so-called Certification Agency (CA). An example of an organization that issues digital certificates is Verisign (www.verisign.com).

E-MAILS THROUGH S/MIME

Sending a text message via e-mail on the Internet is similar to mailing a picture postcard. The mailing process is inexpensive and quick, but your message is public and open for everyone to see. Anyone who happens to see the card can pick it up and read your message. E-mail is similar in that it can be easily read by a variety of off-the-shelf hacker applications. This leaves your sensitive data vulnerable.

For secure e-mail transactions the format of Secure/Multipurpose Internet Mail Extensions (S/MIME) was developed. It uses most of the technologies as described in previous sections of this chapter. E-mails can send encrypted e-mail messages including attachments, and the system allows users to use digital signatures. It uses the RSA public key algorithm for handshake and is the standard default secure message format for Outlook 2000. The life of an e-mail message is illustrated in Figure 39.4.

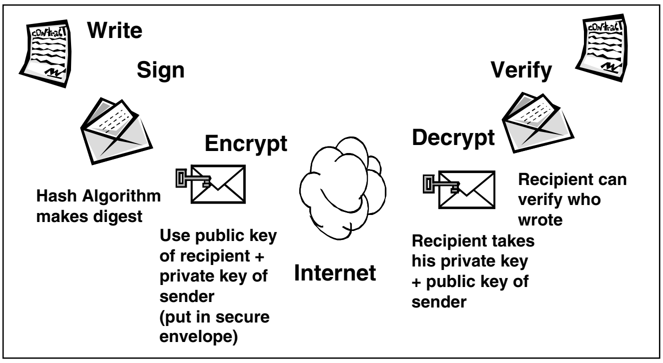

Write
Sign
Hash Algorithm makes digest
Encrypt
Use public key of recipient + private key of sender (put in secure envelope)
Internet
Decrypt
Recipient takes his private key + public key of sender
Verify
Recipient can verify who wrote

FIGURE 39.4 Life of a S/MIME E-Mail Message.

The sender writes the text and can attach documents. The sending computer makes a digest of the message using MD5 hash algorithm. Using either the private or public key of the sender, the message and the attachments are encrypted, and sent over the Internet to the receiving site. The recipient takes his/her private key plus the public key of the sender and decrypts the message. Using the sender's digital signature and certificate the recipient can verify who wrote the message before he/she reads the message and attachments.

Users of Outlook can practice the use of digital signatures free of charge for 3 months. Select "tools," "options," "security," and "get digital certificates" under "digital IDs." If you choose Verisign as a Certification Authority you get a free digital signature for 3 months or you can purchase the digital certificate for an annual fee of $20 to $30 (Status 2003).

VIRTUAL PRIVATE NETWORKS

Usually we want to send confidential information not only to third parties, as shown in the example in Figure 39.1, but also within our organization. To do this we use our corporate network. Access to a corporate network can be very well controlled as long as the company uses transfer lines inside its firewall. However, there is also a desire to communicate confidential data across the globe, e.g., to remote sites, business travelers, home offices, and trusted business partners.

In the past, companies leased private telephone lines to build private networks. This is very expensive, but at that time there was no other possibility to protect their data. With modern technology as described in previous chapters, companies can now achieve very much the same level of security and confidentiality using public telephone lines and Internet Service Providers (ISPs). This concept is called Virtual Private Network (VPN) and is illustrated in Figure 39.5. As soon as data travel outside the firewall they are encrypted. Users outside the firewall may be employees working in branch or home offices, business travelers, or business partners. They can log into the system through all types of Internet connections, for example, modems or high-speed Internet connections (DSL). VPNs can provide secure connection between computers over the Internet and they are cheaper than private networks.

Access security is the major concern of VPNs. Static passwords are not secure enough. Access to VPNs typically requires dynamic password control through tokens. These are credit card-size devices that a user must physically have when entering the system. When accessing the user enters

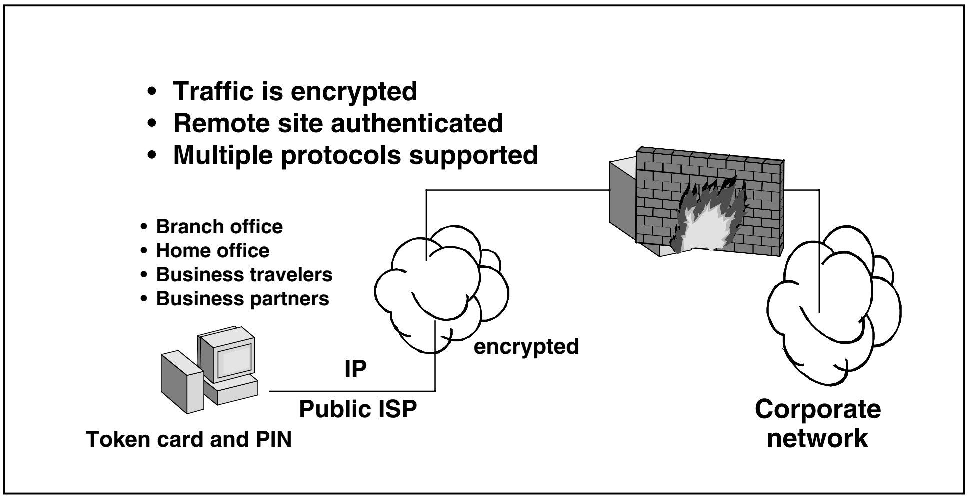


FIGURE 39.5 Virtual Private Network.

a PIN and the token generates a password that is displayed on the token and expires after a relatively short period of time, e.g., 1 minute. The password is synchronized with the authentication system of the target system.

VALIDATION

All types of computer hardware and software are used for Internet communication. Because these are typically used in regulated environments, we could assume that all such computer hardware, peripheral devices, and software should be validated. However, because of the nature of the Internet this is unrealistic.

For example, the FDA's 21 CFR Part 11 validation guidance[3] states:

> *We recognize that the Internet, as computer system, cannot be validated because its configuration is dynamic. For example, when a record is transmitted from source to destination computers, various portions (or packets) of the record may travel along different paths, a route that neither sender nor recipient can define or know ahead of time. In addition, entirely different paths might be used for subsequent transfers.*

The guidance also states that computers at the source and destination should be validated:

> *Validation of both the source and destination computing systems (i.e., both ends of the Internet communications pipeline) should extend to those measures. We therefore consider it extremely important that those measures are fully documented as part of the system requirements specifications so they can be validated.*

In addition, the guidance recommends digital signatures to verify data integrity as well as some kind of confirmation that data have been received:

> *Use of digital signature technology to verify that electronic records have not been altered and that the sender's authenticity is affirmed.*
>
> *Delivery acknowledgements such as receipts or separate confirmations executed apart from the Internet (e.g., via fax or voice telephone lines).*

The Open Web Application Security Project (OWASP)[2] has identified unvalidated paramters as the number one reason for vulnerability of Web applications. Validation of an Internet application comprises six parts:

- Configuration management and documentation of all hardware and software at sending and receiving site.
- Validation of applications on source and destination computers.
- Test correct browser functionality and user interface.
- Usability testing (during development).
- Verification of correct file transfer.
- Security testing.

CONFIGURATION MANAGEMENT

The objective of configuration management is to have detailed information on the system initially and after any changes. Wrongly configured Web and application servers are among the ten most frequently found Web vulnerabilities.[2] SOPs should be available for both initial configurations as well as for planned and unplanned changes. Worksheets are useful to document initial configurations.

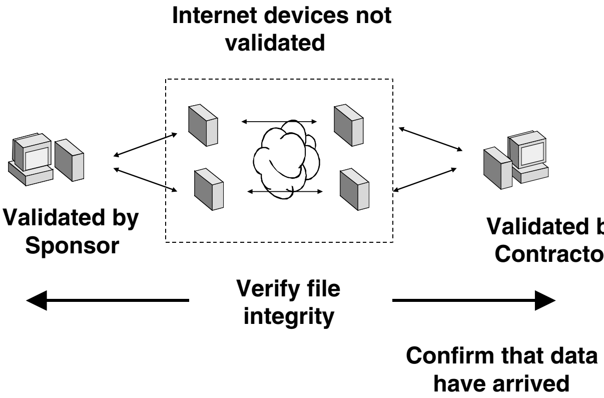
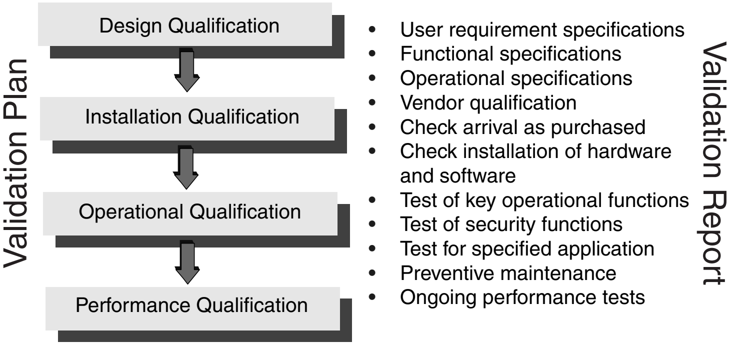


FIGURE 39.6 Validation Activities of Internet Applications.


Validation Plan
Design Qualification
Installation Qualification
Operational Qualification
Performance Qualification
Validation Report

- User requirement specifications
- Functional specifications
- Operational specifications
- Vendor qualification
- Check arrival as purchased
- Check installation of hardware and software
- Test of key operational functions
- Test of security functions
- Test for specified application
- Preventive maintenance
- Ongoing performance tests

FIGURE 39.7 Four Q-Model Life Cycle Concepts for Internet Computers at Source and Destination Site.

This includes computer hardware, operating software with product name and revision number, application software with product name and revision number, network devices with product name, software and firmware revision, cables, documentation such as user manuals and configuration settings. Any changes should be documented following documented change control procedures.

Validation of Computer Applications at Sending and Receiving Site

For the validation of computer applications at the sending and receiving site we recommend applying any life cycle concept, e.g., the V-Model according to GAMP 4[13] or the four Q-Model as illustrated in Figure 39.7. The general recommendation is to follow well-accepted computer validation practices that have been described in the literature.[14] Individual steps are illustrated in Figure 39.7. Validation activities should be well planned and documented in a validation project plan with validation activities, owners, and time schedule. The validation approach and results should be documented in a validation report.

When we validate Web applications we should also look at Internet specifics. For example, special attention should be given to the validation of authorized access. Tests to validate authorized access should be performed during initial set-up and repeated on an ongoing basis.

Testing Browser Functionality and User Interface

User interface of Web applications are browsers such as Netscape's Navigator or Microsoft's Internet explorer. The compatibility of various browsers with the applications should be tested. Testing should include:

- Correct functioning of scroll bars to make sure users can scroll through items and make correct selection from a list of items
- Correct functionality of buttons
- Correct hyperlinks to make sure that the correct application is started
- Correct arrangement and functioning of frames
- Test if all information is visible on the screen at different screen resolutions

Usability Testing

Usability testing is important for any software project and done during the design phase. It is extremely important because Web applications are used by many different users with different skill sets and expectations. Typically user interface prototypes are developed and tested by all types of anticipated users. Easy navigation through screens and applications are most important for Web applications. There are a couple of more considerations specifically for Web applications:

- Is the site visually appealing?
- Is information easily accessible?
- Is there a home button or link at every page?
- If there are frames, are they easy to access?
- Are the fonts big enough?
- If files are offered for download, is the file size displayed?
- If downloads are offered in special file formats, is there a link to tools that can be used to view or print the file content?
- If data entries are accepted for a specific range, is the range displayed and if data are entered outside the range, is an error message displayed?
- Is there contact information, e.g., an e-mail link to get further information or help?

Verification of Correct File Transfer

An essential application of the Internet is exchanging data. The most important validation task is to verify correct data transfer when uploading and downloading files to the Web server and for e-mail communication. Correct file data transfer should be tested not only under normal but also high load conditions. Test variables should include Web browser, time of day, location of destination computer, and file size. For verification of correct file transfer we can use hash calculations, which are also used for digital signatures. Important is development of specifications, for example, maximum file size.

Security Testing

Security testing is most important for Web applications. Users need to be confident that only authorized users can get accesses to confidential data. Access through the public Internet should be encrypted and confidentiality and authenticity should be tested. Test scenarios should be set up to:

- Deny incorrect user-ID/password access for intranet and VPN applications
- Password expirations
- Check authorized access to certain areas
- Check data integrity (see also section on “Hash Functions”)
- Check data filtering at firewalls

DEVELOPMENT AND COMMUNICATION OF PROCEDURES

In previous sections of this chapter, we discussed techniques that can make the Internet trustworthy. However, all this does not guarantee trustworthiness if the users do not develop a culture toward Internet security. This requires development of procedures for good Internet practices and training on how to use the procedures day by day as well as enforcement of the procedures.

While on-line users should think with every click and keystroke about what they are doing and consider its potential impact on privacy and security, they should also have a good understanding of the technology used to protect themselves and the entire corporation.

Such procedures should be available on training along with information on the technologies which were described in previous sections such as using digital signatures and certificates, sending and receiving e-mails with S/MIME, and validation of computers at the sending and receiving site.

In addition, procedures should be available for more normal use of the Internet, whether it is used for regulated applications or not. These are procedures that should help protect the computer and data against accidental or incidental attack from outside.

Such procedures should include:

- Regular update and use of virus programs
- Downloading of data and programs from the Internet
- Configuration of the system for highest security

We recommend adding a section on "Using the Internet in a regulated environment" in the company's validation or compliance master plan.

For example, recommended steps to protect the computer from viruses can be:

- Purchase, install, and regularly run virus scanning software on your computer.
- Keep virus scanning software current.
- Do not open e-mail attachments if you do not know who has sent them.
- Scan attachments of incoming e-mail messages and other new files before opening/using them.
- Before using a floppy, scan it for viruses.
- Disable macros and/or macro features if you do not use them.
- If a file arrives compressed or zipped, check it for viruses before and after unzipping it.

Downloading programs and data should follow an SOP "Policies for Downloading Files from the Internet":

- Download files only from well-known and reputable Web sites (you can typically trust downloads from a software vendor such as Microsoft).
- Check downloaded files for viruses before using them.
- Set security zone on MS Internet Explorer to Medium or High.
- Block cookies through appropriate settings on the MS IE.

Procedures are not of much value if they are not followed. The internal audit program should include procedures on how to use the Internet.

Security procedures are also important but do not fit into the scope of this chapter. We recommend looking at the literature; for example, NIST has published good guidance documents on IT security, the NIST Security Guide for Interconnecting Information Technology System,[16] and the Security Self-Assessment Guide for Information Technology Systems.[17]

Microsoft also has good recommendations for security, for example, on the Windows 2000 security Web site.[18]

REFERENCES

1. Code of Federal Regulations, Title 21: Part 11, *Electronic Records; Electronic Signatures;* Final Rule; *Federal Register*, 62(54), 13429–13466, Rockville, MD.
2. Open Web Application Security Project (OWASP), http://prdownloads.sourceforge.net/owasp/OWASPWebApplicationSecurityTopTen-Version1.pdf?download.
3. FDA (2001), Guidance for Industry: 21 CFR Part 11, Electronic Records; Electronic Signatures Validation (Draft), U.S. Food and Drug Administration, August.
4. Danda, M. (2001), Protect Yourself Online, Microsoft Press, www.microsoftpress.com.
5. RSA, www.rsasecurity.com.
6. Institute of Electrical and Electronics Engineers Working Groups, "IEEE P1363 Standard Specifications For Public-Key Cryptography," http://grouper.ieee.org/groups/1363/index.html.
7. American Bar Association, "Digital Signature Guidelines," http://www.abanet.org/scitech/ec/isc/dsg-toc.html.
8. RFC 2527, "Internet X.509 Public-Key Infrastructure, Certificate Policy, and Certification Practices Framework," http://www.ietf.org/rfc.html.
9. RSA, "Understanding Public-Key Infrastructure (PKI) Technology,"http://www.rsasecurity.com/products/keon/whitepapers/pki/PKIwp.pdf.
10. The PKI Page Web site. http://www.pki-page.org.
11. PKI Forum Web site. http://www.pkiforum.org.2.
12. American Bar Association (2001), "PKI Assessment Guidelines (Draft)," http://www.abanet.org/scitech/ec/isc.
13. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP4), published by International Society for Pharmaceutical Engineering (www.ispe.org).
14. Huber, L. (2002), *Validation of Computerized Analytical and Networked Systems*, Interpharm, Englewood, CO.
15. Huber, L. (2002), Network Quality Package, Labcompliance, www.labcompliance.com/books/network-quality.htm.
16. NIST (2002), Security Guide for Interconnecting Information Technology Systems, http://csrc.nist.gov/publications/nistpubs/800-47/sp800-47.pdf.
17. Security Self-Assessment Guide for Information Technology Systems.http://csrc.nist.gov/publications/nistpubs/index.html.
18. Windows 2000 Security Services Web site, http://www.microsoft.com/windows2000/technologies/security/default.asp.

40 Case Study 22: Medical Devices and Their Automated Manufacture

Guy Wingate, GlaxoSmithKline

CONTENTS

A medical device is an instrument, apparatus, appliance, material, or other article, whether used alone or in combination, together with any software necessary for its proper application, which

1. Is intended by the manufacturer to be used for human beings for the purpose of:
   a. Diagnosis, monitoring, treatment, alleviation of disease
   b. Diagnosis, monitoring, treatment, alleviation of, or compensation for, an injury or handicap
   c. Investigation, replacement, or modification of the anatomy or of a physiological process
   d. Control of conception
2. Does not achieve its principal intended action in or on the human body by pharmacological, immunological, or metabolic means, even if it is assisted in its function by such means.[1]

The definition's reference to software extends the scope of medical devices to include those based on programmable technology. Such devices can be extremely complex and consist of a large

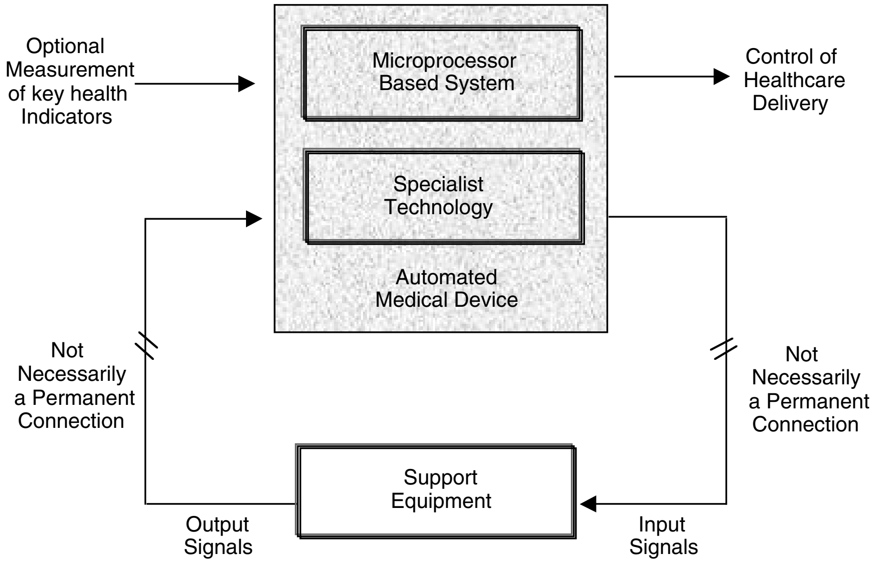

Optional Measurement of key health Indicators
Microprocessor Based System
Specialist Technology
Automated Medical Device
Control of Healthcare Delivery
Not Necessarily a Permanent Connection
Not Necessarily a Permanent Connection
Support Equipment
Output Signals
Input Signals

FIGURE 40.1 Automated Medical Device.

number of programmable elements. Figure 40.1 provides a schematic overview of a medical device's healthcare service: the delivery, control, and monitoring of medical treatment.

Medical devices require both product and process validation. Product validation is necessary to assure that they are designed and assembled consistently to assure a high quality of service. Process validation is necessary to assure that they are produced under a compliant regime of Good Manufacturing Practice (GMP). Particular validation requirements are laid down by European Directive 93/42/EEC[2] and the U.S. Code of Federal Regulations Title 21: Part 820.[3] Some medical device manufacturers in Europe seek an additional CE marking, which is a quality standard given to organizations that successfully pass a quality inspection by a regulatory agency.

Regulatory expectations also require that "when computer or automated data processing systems are used as part of production or the quality system, the [device] manufacturer shall validate computer software for its intended use according to an established protocol."[3] In addition, computer systems that implement part of a device manufacturer's production processes or quality system may be subject to electronic record and electronic signature requirements. Medical devices destined for the U.S., for instance, will be subject to 21 CFR Part 11.[4] Example computer applications that typically require validation include medical device design tools, laboratory testing and analysis, product inspection and acceptance, production and process control, environmental controls, packaging, labeling, document control, and compliant management.[5]

This case study discusses the particular issues affecting the automated manufacture and validation of a medical device involving the management and coordination of a number of suppliers:

- One medical device designer
- One technology development designer
- Two supporting equipment designers
- One medical device manufacturer
- Four manufacturing process lines

The approach to validation is unchanged from that described in this book with one exception. Risk assessments for medical devices should focus on severity of instances of erroneous behavior.

Risk assessments should not take account of the probability of erroneous behavior as any occurrence of erroneous behavior may be critical.

VALIDATION PLANNING

The coordination of suppliers is vital, and, to this end, a Validation Master Plan is often produced, referencing a number of Validation Plans for each element of the automated medical device. Suppliers should be encouraged to produce their own Quality Plans in response to the Validation Plan for their portion of the system so that any inconsistencies and ambiguities can be identified and corrective actions instituted before change becomes too inconvenient and expensive.

The selection of suppliers is often limited because only one or two suppliers will generally have the capability to provide a particular item of technology used in the medical device or in its manufacture. It may not be the case, therefore, that the supplier to be used has any experience in validation. Indeed, many suppliers supporting medical devices are small organizations and have limited opportunity to develop an in-house validation capability. If this is the case, then training should be given to ensure that the supplier has no misunderstandings about the expectations made by GMP regulators. Care must be taken not to always take capability "sales-speak" at face value. The use of external validation consultants and the GAMP Guide for supplier validation can prove useful in training mechanisms.[6] Most suppliers are keen to pick up new skills and will welcome the chance to enhance the competency of their staff.

Because of such uncertainties, each supplier should be audited at the start of the project to establish whether a Quality Management System (QMS) exists that will support the validation of the equipment or process. Where none exists, an agreement must be reached with each supplier as to what quality measures will be used. This should be outlined by each company in their Quality Plan. In our case, we decided that all companies concerned should comply with the GAMP Guide, as this outlines validation documentation that is suitable for the U.S. Food and Drug Administration's (FDA) requirements and may also be used to support the European Union's CE mark accreditation (see Table 40.1). The device was to be released in both the American and European markets.

A particularly important area to be addressed during planning is the consistent use of terminology. The GAMP Guide[6] includes a lexicon of validation terminology that can prove to be a useful reference. In this respect some practitioners prefer to include specific terms associated with a validation project and their definitions in the Validation Plans.

Intellectual property rights (IPR) should also be agreed to as part of the validation planning exercise. Particular elements of the automated medical device or its manufacture may be confidential to individual suppliers. Contracts must clearly define the terms and conditions affecting the supplier's rights.

REQUIREMENTS

Requirements should be developed for the medical device and automated equipment used to support its manufacture. The requirements should clearly state the intended use of software. Areas of special importance include allocation of system functions to hardware/software, operating conditions, user characteristics, and potential hazards.[5]

The FDA recommends that a software requirements specification document is generated. Regulators will deem software unvalidated without predetermined and documented requirements. The scope of requirements should cover:[5]

- All software system inputs
- All software system outputs

TABLE 40.1
CE Marking and Validation Documentation Relationships

EU Directive 93/42/EEC, Annex III, Section 3	Validation Documentation
The documentation must allow an understanding of the design, manufacture, and performance of the product and must contain the following items in particular:	
A general description of the type, including any variants planned	User Requirements Specification
Design drawings; method of manufacture envisaged, in particular, sterilization, and diagrams of components, subassemblies, circuits, etc.	Drawings, Hardware Design and Software Design Specifications
The description and explanation necessary to understand the above-mentioned drawings and diagrams and the operation of the product	Functional Specifications, Operator's Manual
A list of the standards referred to in Article 5, applied in full or part, and descriptions of the solutions adopted to meet the essential requirements if the standards referred to in Article 5 have not been applied in full	Validation Plan, Supplier Quality Plans
The results of the design calculations, risk analysis, investigations, technical test, etc. carried out	Results of clinical trials; Threats and Controls; FMEA, Software Structure Analysis; IQ, OQ, and PQ tests; and Fundamental Science Document
A statement indicating whether or not the device incorporates, as an integral part, a substance as referred to in Section 7.4 of Annex I and data on the test conducted in this connection	Relevant to medical device as a whole; not directly applicable to its automation
This clinical data referred to in Annex X	Clinical trial report
The draft label and, where appropriate, instructions for use	Relevant to medical device as a whole; not directly applicable to its automation

Note: Annex III, Section 3 of the European Union's Medical Device Directive[2] holds the key area that maps CE onto computer validation requirements.

- All functions that the software system will perform
- All performance requirements (e.g., data throughput, reliability, response times)
- How users will interact with the system
- The definition of all external and user interfaces
- The definition of internal system interfaces
- What constitutes an error and how errors should be handled
- The intended operating environment (e.g., hardware platform, operating system)
- Any ranges, limits, defaults, and specified values that the software will accept/reject
- Any potential hazards and design constraints (i.e., safety-related requirements)

Each requirement identified should be evaluated for accuracy, completeness, consistency, testability, correctness, and clarity.[5] The U.S. Quality System regulation requires a mechanism for addressing incomplete, ambiguous, or conflicting requirements (Clause 30c[3]).

In this case study let us consider an insulin delivery system to aid diabetes.[7] Diabetes is a relatively common condition where the human body is unable to produce sufficient quantities of a hormone called insulin. Insulin metabolizes glucose in the blood. The conventional treatment of diabetes involves regular injections of genetically engineered insulin. The problem with this treatment is that the level of insulin in the blood does not depend on the blood glucose level but is a function of the time when the insulin injection was taken. This can lead to very low levels of blood

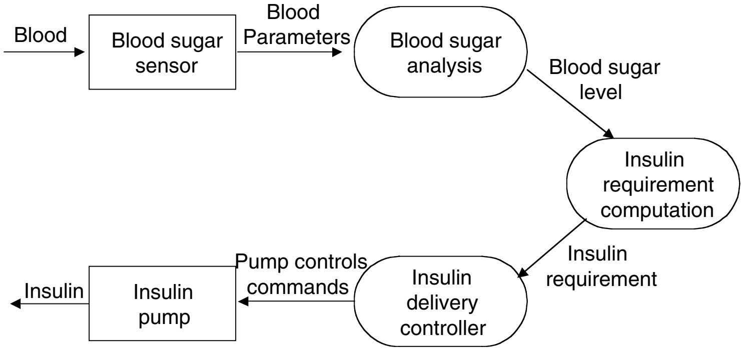


FIGURE 40.2 Example Data Flow for Insulin Delivery System.

glucose (if there is too much insulin) or very high levels of blood sugar (if there is too little insulin). Low blood sugar is, in the short term, a more serious condition as it can result in temporary brain malfunctioning and, ultimately, unconsciousness and death. In the long term, continual high levels of blood sugar can lead to eye damage, kidney damage, and heart problems.

An insulin delivery system might work by using a microsensor embedded in the patient to measure some blood parameter that is proportional to the sugar level.[7] This controller computes the sugar level, judges how much insulin is required, and sends signals to a miniaturized pump to deliver the insulin via a permanently attached needle. Insulin delivery systems are likely to be software controlled. Figure 40.2 is a data-flow model that illustrates how an input blood sugar level is transformed to a sequence of pump control commands.

The requirements for the insulin delivery system would include specific patient-safety needs such as:

- A single dose of insulin shall not be delivered that is greater than the designated maximum dose.
- The daily cumulative dose of insulin shall not be greater than a designated maximum dose.
- An audible alarm shall sound when any device anomaly is detected.
- Diagnostic messages should indicate nature of warning and remedial action required.

A risk analysis is then required to confirm these safety requirements can be met, that the medical device will not malfunction, and that a safe state is maintained in relation to patient health. For the insulin delivery system a safe state is a shut-down state where no insulin is delivered. Over a short period this will not pose a threat to the diabetic's health.[7]

RISK ASSESSMENT (SAFETY CASE)

Risks should be identified that can result in system malfunction or failure. The consequences of failure should be analyzed, along with requirements to mitigate these malfunctions and failures. It has been suggested that risk, which might otherwise be evaluated through likelihood and consequence, should only be factored on consequence because of the social unacceptability of any known harmful impact a medical device might pose. In practice, some allowance must be made for likelihood but with a careful eye also on the probability of detection so that corrective action can be taken.[8]

The process of risk assessment generally involves considering different classes of hazard such as physical hazards, electrical hazards, biological hazards, radiation hazards (where appropriate), and hazards due to service failure. Each of these classes is then analyzed in detail to determine the acceptability of associated risks.

TABLE 40.2
Example Risk Assessment of Identified Hazards

Identified Hazard	Hazard Probability	Hazard Severity	Estimated Risk	Acceptability
Insulin overdose	Medium	High	High	Unacceptable
Insulin underdose	Medium	Low	Low	Acceptable
Power failure	High	Low	Low	Acceptable
Machine incorrectly fitted	High	High	High	Unacceptable
Machine breaks in patient	Low	High	Medium	Unacceptable
Machine causes infection	Medium	Medium	Medium	Unacceptable
Electrical interference	Low	High	Medium	Unacceptable
Allergic reaction	Low	Low	Low	Acceptable

An insulin delivery system, for example, might have the following hazards and associated classes:[7]

1. Insulin overdose (service failure)
2. Insulin underdose (service failure)
3. Power failure due to exhausted battery (electrical)
4. Machine interferes electrically with other medical equipment such as a heart pacemaker (electrical)
5. Poor sensor and actuator contact caused by incorrect fitting (physical)
6. Parts of machine break off in the patient's body (physical)
7. Infection caused by introduction of the machine (biological)
8. Allergic reaction to the materials or insulin used in the machine (biological)

The hazards posed to a medical device associated with its manufacture should also be included. Complex arrangements may require multiple phases of hazard analysis.

The level of risk posed can then be determined and its acceptability considered (see Table 40.2). Risks should be designated acceptable or unacceptable. Unacceptable risks require management. Acceptable risks require no further action.

It is rarely possible to completely mitigate a risk other than by somehow taking action to avoid the associated hazard in the first place. Instead, risks need to be reduced so that they become "As Low As Reasonably Practical" (ALARP). Remedial project actions should be specifically documented — this is sometimes referred to as the "Safety Case." Remedial actions may employ hazard avoidance strategies, introduce hazard tolerant design features, or apply specific project management controls, or a combination. Further information on risk management for medical devices can be found in ISO 14971.[9]

In the example, the first two hazards are software related within the medical device and will require attention as part of the design process. The remaining hazards, meanwhile, are not software related but can be countered by self-checking software that monitors the system state and alerts unsafe conditions. Warnings that alert detection of a hazard should be designed to allow an accident to be avoided by prompting some defined remedial action; for instance, power failure and incorrect fitting of the device. Monitoring software itself, of course, is safety-critical and will require validation.

DESIGN

Early on in the design phase it is important to identify and understand the impact of faults on the medical device so that controls can be incorporated as necessary. Fault Tree Analysis is often used

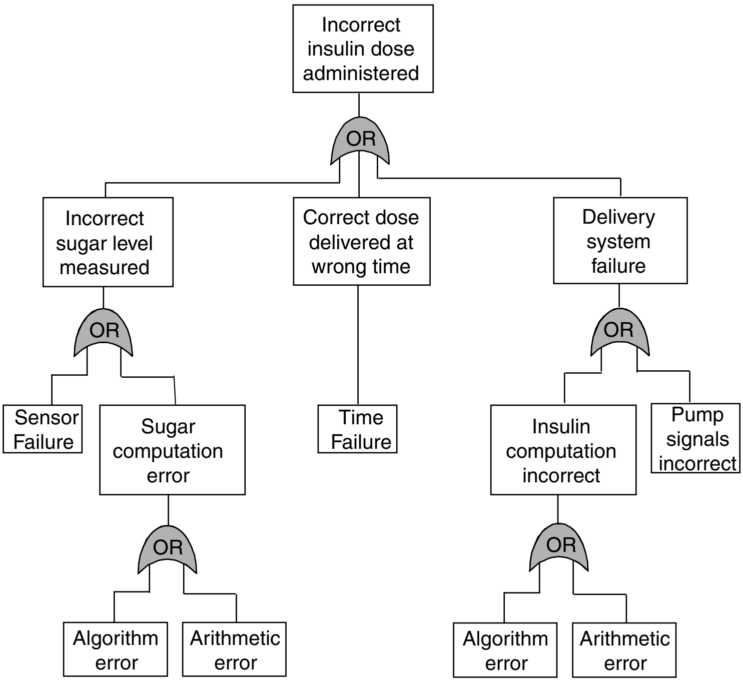


FIGURE 40.3 Example Fault Tree Analysis.

to identify medical device fault scenarios (see Figure 40.3). Recommended design controls including any user procedures should be clearly logged. Arithmetic errors might, for instance, be mitigated though exception handling.

Design documentation will consist of Hardware and Software Design Specifications for equipment and process definitions for process lines. Where a feasibility study has been carried out on a certain technology, a fundamental science document will be generated to summarize the technology on which the medical device is to be based. Fundamental science reports will examine the use of specialist hardware and programming and the ability of these technologies to provide the necessary functionality. The use of particular technologies may be inhibited because they are deemed as unvalidateable or because the validation is too expensive. A justification of the validation approach to be used for different technologies must be documented and made available for inspection by regulatory authorities. It is highly unlikely that regulatory inspectors will be experts in the technologies being used; therefore, a step-by-step argument supported by validation evidence should be developed so that the inspectors can walk through the validation exercise to check its integrity.

DESIGN REVIEW

The FDA recommends that a traceability analysis be conducted from requirements to design, including the risk management documentation, to verify that the design is fit for purpose.[5] A Failure Modes and Effect Analysis (FMEA) should be carried out to confirm that the hardware of the medical device and the supporting equipment cannot fail in an unsafe way. Hazard Analysis and Critical Control Points (HACCP) has also proved useful for some medical device manufacturing processes[10] but may have more limited use directly on software. The use of FMEA and HACCP should be documented, and any recommendations on redesign should be carried out before testing. A check needs to be made that ALARP risk mitigation results in an acceptable residual level of risk (refer back to Safety Case).

SOFTWARE PROGRAMMING

Software written for the medical device and any automated manufacturing equipment should be prepared in accordance with established industry Good Programming Practices. In particular, programs should be well structured and commented, and include headers giving details of version and change Control. A Source Code Review (sometimes known as a Software Structural Assessment) should be conducted to verify the adoption of Good Programming Practices and verify any critical algorithms, such as Fast Fourier Transforms (FFTs) on Digital Signal Processing (DSP) microchips, error handling by the software, and fail-safe or graceful degradation scenarios.

To address concerns that source codes for OTS software may not be available; the FDA will allow "black box testing" as a validation method whenever source codes and design specifications cannot be obtained from suppliers.[5] Alternatively, a Supplier Audit may be used to document that the supplier employed acceptable software programming practices.

ASSEMBLY

It is not uncommon for a company to design a medical device prototype and then have it built by a company specializing in electronic manufacture. Where this happens, as in our case, a number of suitable companies should be considered for the project, and those shortlisted should be audited for capability. Selection would depend on the quality system in place, previous experience in medical device manufacture, control of subcontractors, level of testing supplied, and the amount of in-house technical support available. It is advisable to ensure that a legal agreement, Quality Plan, and User Requirements Specification (URS) are employed to secure production standards for the device.

The circuit board components of some medical prototype devices may have to be reorganized in order to obtain a layout that will allow easy automatic assembly. Any changes must be noted in the Hardware Design Specification, and the new board must be checked against the prototype for functional equivalence. The FMEA report should be consulted to see if there are any critical components on the circuit board. If so, the manufacturer will have to make special arrangements to ensure that component traceability exists from supplier to circuit board to user.

A typical six-step manufacturing process for a medical device would be:

1. Build circuit board. This will usually be done by automatic machines.
2. Check circuit board. The electronic manufacturer should have in-house expertise to program an automatic "in-circuit" tester for the board that ensures all components are in the correct place and that measures the correct value.
3. Load a test program into the device and place in a heat cycling oven. The heating profile and testing time should be agreed on between manufacturer and customer. It is important that the oven is temperature calibrated with a supporting certificate.
4. Load the current version of the validated software into the device.
5. Test the device on a suitable automatic test system.
6. Pack and ship to customers.

It is advisable to use a color-coded labeling system to keep track of the test stages. In our case, a yellow spot was added after successful heat cycling with test software loaded, and a green spot half covering the yellow spot was added once the validated software had been loaded and the board tested successfully; no device could be shipped unless both spots are present. The manufacturer will be expected to supply a certificate of conformity for the devices produced and packed with each batch.

Assembly of the medical device may be partially completed by suppliers before final assembly by the device's registered manufacturer. In such situations, the registered manufacturer is entirely responsible for the work and is expected to assign its own Quality Assurance staff to monitor and perhaps witness supplier assembly.

QUALIFICATION

The medical device and all supporting equipment and processes covered by Validation Plans will each require a test specification. These will be generated by reading through the URS, Functional Specification, Threats and Control Analysis, and (where appropriate) the FMEA for each unit and determining which functions require testing. Critical areas should be examined very carefully, and a demanding set of tests drawn up to check each key area. If an automatic testing system is designed to test the medical device during manufacture, as in our case, the tests will need to check both the pass and fail routes of every test by using test medical devices with known hardware faults introduced.

Another point to remember is that where a software test harness is written to help test a medical device, this must be subject to validation (i.e., it must have functional, design, and test documentation to support it).

The tests will be broken into Hardware Acceptance Tests (or Installation Qualification), System Acceptance Tests (or Operational Qualification), and Equipment Tests (or Performance Qualification). According to the size of the unit, all tests may be in one test document or there may be three separate Qualification documents. A test document will define the test philosophy and how the tests should be run. Each test will have a title, a reason for the test, an outline of any test equipment required, a description of the test, data to be recorded, and the test acceptance criteria. Calibrated test equipment must be supported with calibration certificates.

Each supplier will be responsible for generating its own test specifications so it is important to ensure that all suppliers use the same standard for the test document. Reference to a common standard such as GAMP should be considered.

Environmental Tests

Apart from qualification testing, a medical device should undergo a number of environmental tests, as outlined in standards such as BS 2011 (British Standard for environmental testing), CISPR 11 (limits and methods of measurement of electro-magnetic disturbance characteristics of industrial, scientific), ISM (medical radio frequency equipment), and BS4826 (British Standard for packaging electronic equipment for transport).

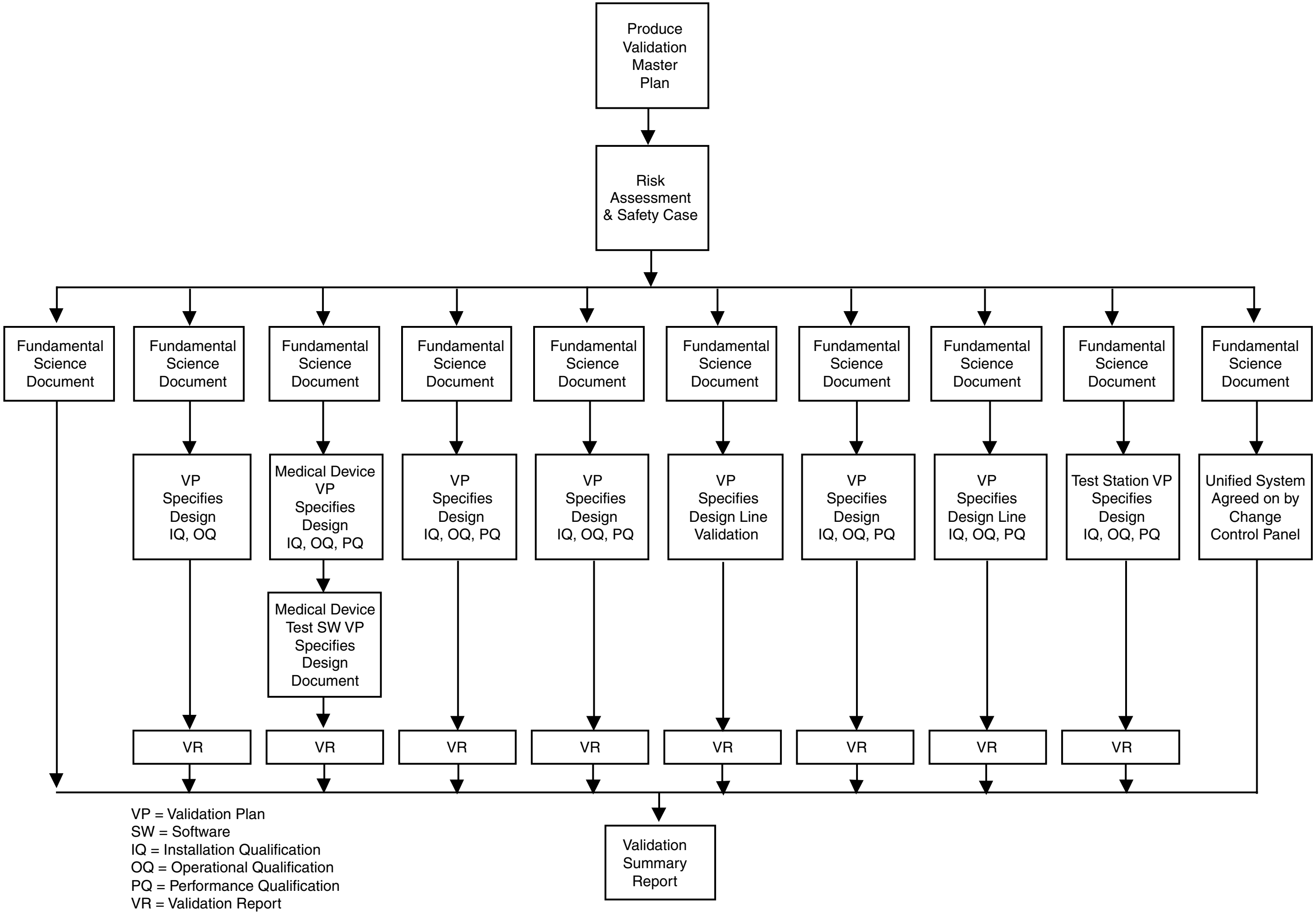
Typical tests that might be carried out on a medical device under environmental testing are as follows:

- Temperature operating range test
- Humidity operating range test
- Pressure operating range test
- Impact test (including drop and bump)
- Vibration test
- Damp heat test
- Transport packaging test
- Radio frequency radiation test
- Radio frequency immunity test
- Reliability test

These tests would be carried out on the final manufactured product by a company specializing in this field.

REPORTING

Potentially, the number of validation documents associated with a medical device can be great because each subsystem of the device is subject to its own validation life cycle (see Figure 40.4).


Produce Validation Master Plan
Risk Assessment & Safety Case
Fundamental Science Document
Fundamental Science Document
Fundamental Science Document
Fundamental Science Document
Fundamental Science Document
Fundamental Science Document
Fundamental Science Document
Fundamental Science Document
Fundamental Science Document
Fundamental Science Document
VP Specifies Design IQ, OQ
Medical Device VP Specifies Design IQ, OQ, PQ
VP Specifies Design IQ, OQ, PQ
VP Specifies Design IQ, OQ, PQ
VP Specifies Design Line Validation
VP Specifies Design IQ, OQ, PQ
VP Specifies Design Line IQ, OQ, PQ
Test Station VP Specifies Design IQ, OQ, PQ
Unified System Agreed on by Change Control Panel
Medical Device Test SW VP Specifies Design Document
VR
VR
VR
VR
VR
VR
VR
VR
Validation Summary Report

VP = Validation Plan
SW = Software
IQ = Installation Qualification
OQ = Operational Qualification
PQ = Performance Qualification
VR = Validation Report

FIGURE 40.4 Validation Documents.

It is very important to collate these documents. They must be managed, otherwise it is likely with the large number of participants that some will be lost, which will severely compromise the validation. In our case, an agreement was reached at the beginning of the project as to what validation documents would be generated by which supplier. This information was recorded in a live Document Tracking Report containing the title of the document, reference, who was responsible, required date, current status, and issue date. It was issued every 4 weeks and proved invaluable with over 100 documents under control. The Document Tracking Report ended with a document library index.

Once the Document Tracking Report shows that all validation documents called for by a corresponding Validation Plan are present, all of the documents are reviewed and a Validation Report is generated, concluding with whether or not the automated medical device is suitable for its intended purpose. Any outstanding issues must be discussed, and either corrective actions are raised or justification for taking no action must be made. The Validation Report should be periodically reviewed to check that the results of any changes, corrective actions, or regulatory requirements have not altered the validated status of the medical device. If additional validation is required, then this should be planned and completed as soon as reasonably practical. Any aspects of the additional validation that directly impact its safety should be addressed immediately.

When the project comes to the end, and all Validation Reports have been completed, each report must be reviewed to ensure nothing is outstanding and that all parts of the project are satisfactory for use. This information is covered by the Validation Summary Report, which should be the last validation document of the validation suite to be written.

CHANGE CONTROL

Change control must be established for the validation project and ongoing support of the medical device. Each supplier associated with the project is likely to have its own particular change control practice. These must either be linked and coordinated or a single change control procedure to be used by everybody must be enforced to ensure effective logging and management of changes. In our case, a single coordinated change control procedure was established with all suppliers. Any supplier can make a change request, but it can only be authorized by all concerned parties. Once authorized, it may be carried out under the quality system of the supplier, with the test results being sent to all parties for approval. The medical device itself must also be subject to change controls. It should be given a serial and version number. All embedded software must also be under version control.

The FDA also requires evidence revalidation whenever a change is made to the software to determine the extent and impact of the change on the entire software system.[5] Baseline validation will be allowed for low-risk devices.

MAINTENANCE AND DECOMMISSIONING

The robust operation of a medical device must be maintained throughout its operational service. Upgrades must be validated prospectively using the same basic life cycle as described earlier for the original medical device. Distribution records need to be maintained so that any medical device upgrade or product recall can be effectively conducted. Such records are also required when notifying withdrawal of support for a medical device and any decommissioning that might be involved.

INSPECTION FINDINGS

A selection of observations taken from a number of different FDA Warning Letters issued in 2001 that reference software is provided below. It is not a comprehensive listing but rather it has been collated in support of the validation approach proposed by this case study.

Failure to validate computer software used to ensure the software will for its intended use as required by 21 CFR 820.70(i).

Failure to validate computer software for its intended use according to established protocol when computers or automated data processing systems are used as part of production or the quality system as required by 21 CFR 820.70(i).

Failure to validate computer software for its intended use according to an established protocol when computers or automated data processing systems are used as part of production or the quality system as required by 21 CFR 820.70(i). For example: your firm's XXXX is computer controlled. It uses software programs to record data from measurements of the radius of curvature and corneal refraction of the eye. However, your firm has not validated the software and computer system used to record this data for its intended uses. Your firm has no documentation to assure that they perform as intended. Also, there is no validation and documentation of subsequent changes to the software.

Failure to validate processes that cannot be fully verified by subsequent inspection and test, as required by 21 CFR 820.75(a). For example, the complaint handling software program, ultrasonic sealing procedure, leak testing procedure, and injection molding procedure have not been validated.

Your firm failed to validate several computer databases that are used for quality functions, including your Access database, your [redacted] software, and your MS Excel spreadsheet program as required by 21 CFR 820.70(i).

Your organization failed to document the selection and design specification of the catheter testing equipment, including the computer system, software, data acquisition hardware, and meters.

Failure to maintain procedures to ensure all purchased or otherwise received products and services conform to specified requirements [21 CFR 820.50]. For example, your firm failed to ensure that the supplier of the main computer board documented all of the required test results to indicate the supplier's quality acceptance of the computer boards manufactured and delivered to your firm.

Your firm failed to establish and maintain procedures to control the design of the device in order to ensure that specified design requirements are met, as required by 21 CFR 820.30. For example, the software designed by your firm was developed without design controls

Failure to validate computer software for its intended use according to an established protocol prior to approval and issuance, and document the results of these validation activities, as required by 21 CFR 820.70(i). For example:

1. The associated computer hardware and software used to identify incoming devices.
2. Software used to control the production and assignment of work orders and the control of master SOPS.
3. The software and hardware used to print labeling.

Failure to maintain procedures to ensure that the device design is correctly translated into production specifications. For example, the [redacted] software source code version 1.6 did not go through a formally documented design transfer process. The source code's electronic file transfer to the master chip before production release was not documented and the approved source code version 1.6 (hardcopy or electronic file) was not retained under Document Controls.

Software validation report not reviewed, approved, and signed.

No documented corrective and preventative action for software bugs found during retrospective validation. Validation testing revealed several responses that were unexpected and may potentially adversely affect the performance of the telemetry device. Yet these responses were not evaluated and addressed. These unexpected responses include the software acceptance of a new patient under an existing patient's identifier without displaying an error message and four other unexpected responses documented in the validation document.

Failure to validate processes with a high degree of assurance where the results cannot be fully verified by subsequent inspection and testing, and have those processes approved and documented according to established procedures, as required by 21 CFR 820.75 (a). Specifically, revalidation of the microprocessor software used in the Palm Pump has not been completely performed following an engineering change in April 2001. Additionally, computer and/or automated data processing system software used in production and quality systems, including the use of electronic signatures, has not been validated.

Failure to address and correct problems with software bugs/errors and defects identified during your retrospective software validation and retrospective risk assessment. You indicate these defects will be reviewed in September 2001. However, you provide no justification to support your continued marketing of these products until such time as these defects and deficiencies are corrected or otherwise resolved. Please explain your reasoning in this matter and provide whatever documentation supports your position that these devices are safe to market.

CONCLUSION

Managing a large number of suppliers increases greatly the administrative complexity of validation. This case study has identified some of the important issues. Other issues will arise with particular projects. The key to success is establishing a partnership between suppliers and the device's registered manufacturer.

It is acknowledged that development activities may be dispersed, occurring at different locations being conducted by different organizations. Regulatory authorities such as the FDA hold the device manufacturer ultimately responsible for ensuring validation is conducted and sufficient, regardless of the distribution of tasks, contractual relations, source of software components, or the development environment.[5]

Above all, a team effort is required. Without it, validation is extremely difficult, if not impossible, to achieve.

ACKNOWLEDGMENTS

The author would like to acknowledge the contribution of Bob Paige, formerly of ICI Eutech Engineering, in preparing the original version of this case study,[11] which was subsequently updated and extended.

REFERENCES

1. United Kingdom Medical Devices Regulations (1994), Regulation 2(1): S1 1994 No. 3017. This U.K. regulation implements EU Directive 93/42/EEC.
2. European Directive 93/42/EEC concerning medical devices — class 1lb Parts III and V.
3. U.S. Code of Federal Regulations Title 21: Part 820, *Good Manufacturing Practice for Medical Devices*, U.S. Food and Drug Administration, Rockville, MD.
4. U.S. Code of Federal Regulations Title 21: Part 11, *Electronic Records; Electronic Signatures*, U.S. Food and Drug Administration, Rockville, MD.
5. FDA (2002), General Principles of Software Validation; Final Guidance for Industry and FDA Staff, U.S. Food and Drug Administration, Rockville, MD.
6. GAMP Forum (2001), *GAMP Guide for Validation of Automated Systems* (known as GAMP 4), published by International Society for Pharmaceutical Engineering (www.ispe.org). International Society of Pharmaceutical Engineering, The Hague, the Netherlands.
7. Sommerville, I. (2001), *Software Engineering,* 6th Edition, Addison-Wesley, Reading, MA.
8. FDA (1998), *Guidance for the Content of Premarket Submissions for Software Contained in Medical Devices, Guidance for FDA Reviewers and Industry*, Center for Devices and Radiological Health, Food and Drug Administration, Rockville, MD.
9. ISO 14971 (2000), Medical Devices — Application of Risk Management to Medical Devices.

10. Jahnke, M. and Kuhn, K.D. (2003), Use of Hazard Analysis and Critical Control Points (HACCP) Risk Assessments on a Medical Device for Parenteral Application, *PDA Journal of Pharmaceutical Science and Technology*, 57(1), January/February.
11. Paige, R.A.F. and Wingate, G.A.S. (1997), Validating Medical Devices and Their Automated Manufacture, in *Validating Automated Manufacturing and Laboratory Applications: Putting Principles into Practice* (Ed. G.A.S. Wingate), Interpharm Press, Buffalo Grove, IL.

41 Case Study 23: Blood Establishment Computer Systems

Joan Evans, ABB

CONTENTS

In line with the rest of the pharmaceutical and healthcare industries, the use of computer systems by blood establishments has increased rapidly in recent years. These systems now assist, manage, and, in some cases, control the analysis, creation, and management of critical records for whole blood, blood components, and blood derivatives.

Typical features of computer systems and associated software products encountered in a 21st century blood establishment include:

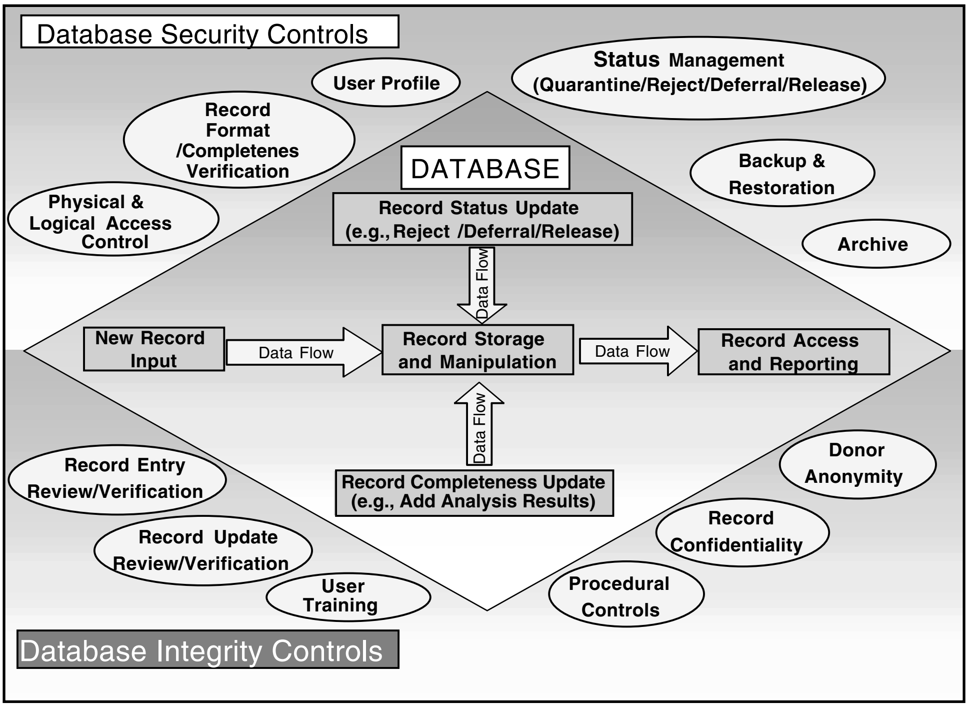

Database Security Controls
User Profile
Status Management (Quarantine/Reject/Deferral/Release)
Record Format /Completenes Verification
DATABASE
Backup & Restoration
Physical & Logical Access Control
Record Status Update (e.g., Reject /Deferral/Release)
Archive
Data Flow
New Record Input
Data Flow
Record Storage and Manipulation
Data Flow
Record Access and Reporting
Data Flow
Donor Anonymity
Record Entry Review/Verification
Record Completeness Update (e.g., Add Analysis Results)
Record Confidentiality
Record Update Review/Verification
User Training
Procedural Controls
Database Integrity Controls

FIGURE 41.1 Blood Product Database Record Security and Integrity.

- Database systems for record management (e.g., Blood Inventory Management systems, Donor Management Information systems)
- Environmental control (e.g., Building Management Systems)
- Manufacturing control equipment (e.g., PLCs on a hematocrit centrifuge)
- PC-based systems on laboratory analysis equipment (e.g., HPLCs, GCs, electrophoresis equipment, and automated blood typing machines)
- MS Excel Analysis spreadsheets
- Laboratory Information Management Systems
- Bar code labeling and container printers and readers

While this case study focuses on database systems to assist in decision making and management of data associated with blood products and their donors, the principles and good practice outlined are applicable to all computer systems handling critical data with potential public health impact.

MEETING REGULATORY EXPECTATIONS

Contemporaneous with technology developments, concern among the public at large has increased about issues such as confidentiality, traceability, and, in particular, the potential for blood contamination. All this has resulted in increasing pressure on blood establishments to ensure the consistency and reliability of their operations and the security and data integrity of their critical records (see Figure 41.1).

Regulatory requirements governing blood establishments stem from three sources:

- As per the **U.S. Public Health Service Act**,[1] blood, blood components, and blood derivatives are defined as biological products; therefore, 21 CFR Part 600,[2] Part 606,[3] and Part 610[4] apply.

- Since blood and blood components are classed as drugs in the Federal Food, Drug, and Cosmetic (FD&C) Act, current **Good Manufacturing Practices (cGMP)** as defined in 21 CFR Parts 210[5] and 211[6] apply.
- Blood bank software products are classed as medical devices; therefore, 21 CFR Parts 800[7] and 820[8] apply.
- Other regulatory requirements are included in CFR 640 "Additional standards for human blood and blood products."[9]
- European Union requirements for blood establishments are to be found in the ***Guide to Good Manufacturing Practice for Medicinal Products***.[10]

21 CFR Part 11 — Electronic Records; Electronic Signatures

No other regulation in the history of the FDA has been as widely publicized and hotly debated as the U.S. 21 CFR Part 11 regulation governing electronic records and electronic signatures. In its focus on data integrity and audit trails to provide traceability it goes to the heart of public health and safety concerns and potential risks concerning blood establishment operations. The recent new guidance from the FDA, which has clarified and narrowed the scope of application of 21 CFR Part 11,[11] has also highlighted the base predicate rule requirements on the reproducibility of critical records throughout their retention period, and the preservation of integrity of their data. These are, obviously, issues that are fundamental to blood establishment operations, particularly to their database systems. Use of nonbiometric methods, e.g., user ID plus password, is becoming increasingly common for signature of blood establishment records on-line, in which case the regulatory requirements for electronic signatures also apply.

Regulatory Reporting Requirements

All blood establishments are required to submit information describing their proposed computer systems along with their Establishment License Application (ELA). Moreover, any "important" proposed change to a computer system or database is also reportable as a supplement to the original ELA.

The gravity of the potential consequences associated with blood establishments' significant record-keeping irregularities resulting from failures to control software adequately, and the seriousness with which the regulatory agencies view such deficiencies, are underlined by a series of FDA Warning Letters. These observations catalog a depressing picture of:

- Software control deficiencies (lack of control on software configuration settings, software put into production with contaminant decision logic package errors, lack of root cause and preventative/corrective action, etc.)
- Inadequate control of installation qualifications and validation
- Failure to maintain adequate documentation

They may also include:

- Nonconcurrent documentation
- Falsification of records cases

All these, despite previous observations, clearly highlighting the urgent need for significant review and supplementation of procedures for software control, records management, and the addressing of a "culture to hide problems" as reported in a document from at least one National Testing Laboratory (NTL).

Other examples of FDA Warning Letter observations include:

- "***Failure*** to maintain records concurrently with the performance of each significant step in the collection, processing, compatibility testing, storage, and distribution of each unit as required by 21 CFR 606.160 and 211.188."
- "***Failure*** to conduct validation studies on the [redacted] software used to control irradiation dose calculation, irradiation timing, product expiration dating, and electronic data reporting [21 CFR 211.100(b)]."
- "***Failure*** to establish written procedures that include all steps to be followed in the collection, processing, compatibility testing, storage, and distribution of blood [21 CFR 606.100(b)]. For example, no written procedures exist that define the [redacted] software are Donor Module that has been in use at your facility since November 1998."

The party with regulatory responsibility for compliance of computer systems to the above may be the blood establishment itself or the system supplier (as the manufacturer/distributor of the medical device, in this case, the computer software).

VALIDATION STRATEGY

Recent regulatory developments in the U.S. have highlighted the need for a "back to basics" approach focusing on the predicate rules identified above and a risk-based rationale to meeting these requirements. Even the most superficial of Risk Analyses or GxP assessments will swiftly confirm that blood establishment database systems handling records such as the following should be classed as systems handling high risk critical data and hence requiring validation:

- Donor registry, donor deferral (donor personal details)
- Blood unit laboratory analysis results (ABO/Rh/infectious disease, e.g., hepatitis, HIV)
- Compatibility testing data (donor blood/blood components to potential recipient)
- Quarantine/release/rejection data
- Shelf life of blood products

These computer systems should be *prospectively* validated because only by so doing can one be assured that they will reliably and consistently meet their intended function, all relevant regulatory requirements, and most important of all, the safety of the blood recipient.

Prospective validation is executed and completed before release of the computer system for use and operation in the live environment and will encompass the complete system, including all hardware and software components and interfaces, and its associated operational and maintenance environment.

Retrospective validation, i.e., the validation of any blood establishment computer system which is already being used to manage critical records, should only be viewed as a corrective interim measure in response to deficiencies noted regarding existing system capability (e.g., inadequate access control vs. the requirements of 21 CFR Part 11[11]) and validation efforts. Such retrospective validation should be part of a remediation program that may include planned retirement or replacement of the system to meet current regulatory expectations. In the case of database systems, retrospective validation (research of existing documentation, supplementary testing, and additional SOPs) may be used to confirm the integrity of existing records.

A STRUCTURED LIFE-CYCLE APPROACH

A life-cycle approach, for example, as outlined in GAMP 4,[12] to the specification, design, implementation, and testing of the computer system components will provide a sound basis for the validation of blood establishment database systems.

One of the first decisions to be made by the blood establishment is whether the database system needs:

- Supplier development of an existing product
- In-house development of an existing product
- In-house development of an entirely new system

These days most data management systems are based on standard database packages such as Oracle, Sequel Server, etc. Although some data entry screen and reports may be customized, much of the user interface will be via configured routines and standard database queries. Completely customized in-house developed systems are, therefore, becoming more rare. However, a larger blood establishment may well choose to develop its own system, in which case the blood establishment also assumes the responsibilities associated with these life-cycle steps and so should follow one of the industry standards for software development, e.g., ISO 90001:2000 TickIT and BS7799 for Information Security. The U.S. CDRH's General Principles of Software Validation[13] also provides guidance concerning software product development and the regulatory authorities' current approach to evaluating validation of a computer system.

Computer System Definition

The first step in a structured approach to the validation of a database system is the Computer System Definition. This is one of the most important steps for any pharmaceutical project, particularly so for a blood establishment database system. Critical considerations include:

- Scope of system — donor management, blood inventory management
- Shared resources — physical and logical data separation
- GxP and non-GxP data in separate databases
- GxP and non-GxP data on separate hardware (e.g., different servers)
- Separate SOPs for GxP and non-GxP operations/maintenance
- Type, structure, content, format, and quantity of data records to be stored
- Data processing, manipulation, search criteria
- Data entry screen layout
- Reports: number, format, content, scheduling
- Interfaces with other systems, e.g., with local LIMS, remote blood collection centers, etc.
- Number of users and potential for simultaneous usage of system
- Required physical and logical system access
- Management and protection of confidential data, access privileges
- Controls and checks on data input
- Regulatory reporting requirements
- Compliance with local, state, and national/federal regulations

This documentation of the user's needs and requirements, which is often referred to as the User Requirements Specification (URS), is crucial in minimizing the risk of misunderstanding between the user (blood establishment) and the system developer/supplier.

System Validation Protocols

A validation protocol is a documented, pre-agreed sequence of activities that are to be executed by a nominated team of people who are to review and test the computer system or part thereof. The objective of the protocol is to provide documented evidence that the system is installed correctly and will operate reliably and consistently to meet all user and regulatory requirements.

Computer system validation activities will occur both during as well as at the end of the database development life cycle. Successful validation is highly dependent upon a comprehensive approach to specifications, reviews, installation, inspections, analyses, and testing of both the system hardware and software.

Database validation protocols should cover, as a minimum:

- Scope of system, boundaries, interfaces with other systems
- Sequence of validation activities to be performed
- Roles and responsibilities
- Quality standards to be followed
- Procedures and equipment to be used
- Test execution steps
- Acceptance criteria

These tests, along with all other validation activities, may be summarized in an overall computer system Validation Plan (VP) which is prepared early in the system development life cycle (often concurrent with the system-detailed user requirements definition).

The system validation protocols may be produced either by the blood establishment or on its behalf. Responsibility for approval of documents as fit for use remains with the blood establishment in all cases.

Supplier Selection

As with any computer system handling critical data, the choice of supplier is crucial to on-time and in-full delivery of a system meeting both client and regulatory expectations. Key considerations should be the robustness of the supplier's Quality Management System, the caliber of the project team put forward by the supplier, and the supplier's previous experience with similar blood establishment clients/projects.

Unless the supplier in question has already been accredited (better yet, proven) as a preferred supplier, a formal Supplier Audit is an essential prerequisite. Smaller blood establishments may choose to delegate a trusted third party to carry out an audit on their behalf. In all cases, a comprehensive checklist of the type outlined in Appendix M2 of GAMP 4[12] should be used.

Database Development

It is a *sine qua non* of successful project completion that the supplier fully understands its responsibilities throughout the system development phase. These encompass not only those associated with any critical computer system but also the specific requirements for medical devices as defined in 21 CFR 800,[7] 820,[8] etc. In particular, formal, independent Design Reviews as outlined in Reference 13 may not be part of the supplier's Quality Management System but are considered a key tool in managing and controlling system development in particular system changes. A key Design Review activity will be the evaluation of how comprehensively user and functional requirements have been captured in the system design. This may be done via a cross-reference mechanism like Requirements Traceability Matrix (RTM). Current regulatory expectations for Design Review would include Source Code Reviews to address any bespoke elements, e.g., for reporting or data extraction and Configuration Reviews for standard elements such as reports.

Change Management

Throughout the computer system project, a rigorous approach to Change Management should be adopted. Specifically, formal change control should be applied to all changes to approved items, e.g., validation documentation and database design. As mentioned above, a significant change impacting database functionality (e.g., the way data is manipulated or interpreted) or impacting its equivalence with a previously approved database system may require a change to the submission for blood establishment computer software for CBER or other regulatory authorities.

System Testing

The structured life-cycle approach continues with testing of data entry screens and database reports as and when they become available. On a large database project testing may be a phased activity. System testing through the installation phase has the objective of demonstrating and documenting that the system is installed and performs according to specifications and requirements and may include:

- ***Normal testing***, e.g., input of donor registry data which falls within the expected ranges (the validation of input data will have been specified in the design documentation)
- ***Boundary testing***, e.g., testing of unusual data input format — e.g., lack of house number in a donor address
- ***Invalid case testing***, e.g., invalid blood group entered
- ***Stress testing***, e.g., testing for maximum number of simultaneous system users
- ***Special case testing***, e.g., incomplete entry of blood analysis data

Good practice for database systems development would include the provision of separate system environments for development and testing/validation as well as the live environment. While the first two may be installed on the same physical hardware (server) as the live environment, rigorous quality management systems including logical access control would need to be used to minimize the potential for impact on blood establishment operations in the live environment. It is also an essential requirement to demonstrate equivalence between the testing/validation and live environments to provide assurance that following transport of the database to the live environment further validation is not required. A separate server for development and testing is normally a more practical solution.

Parallel System Operation/System Cut-Over

In many cases, the blood establishment database system may not, strictly speaking, be a “new” system but rather an expansion or replacement of an existing system (which in turn may be computerized or manual). In the U.K. it is a regulatory expectation that parallel use of paper and computer systems is performed when a database is replacing a paper system for a period sufficient to confirm that the computer system is functioning correctly and reliably.

Such a situation offers both benefits and challenges. Parallel testing, i.e., the running of two systems in parallel and comparing the outputs, offers the opportunity of increasing the level of confidence that the system meets all expectations and user requirements. However, at some point, cut-over to sole operation of the new or modified database will have to be achieved. A crucial consideration here is the assurance that all records from the previous system (be they manual, from a different database, or from a “smaller” version of the new database) are available for further processing, inspection, and manipulation, and most importantly, have had their data integrity preserved. This is usually achieved by validation of the process used to transfer the data into the new system.

Computer System Environments

Computer System Environments, illustrated in Figure 41.2, denote the areas of responsibility for activities and system documentation supporting the development, validation, and ongoing management/control of the database.

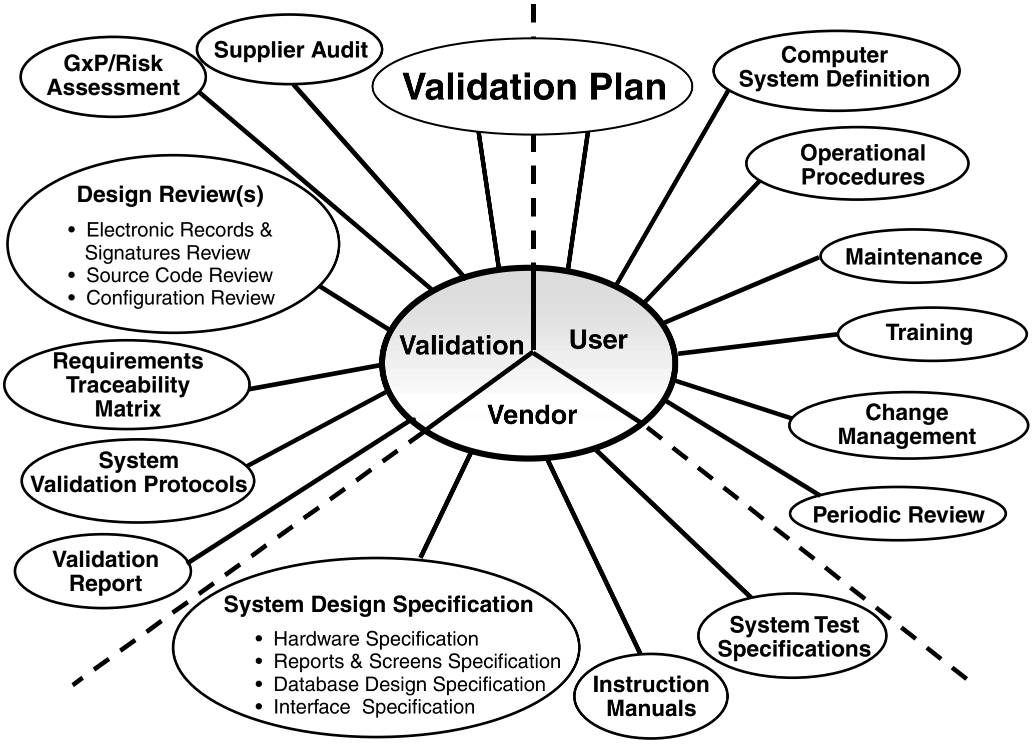

Validation Plan
GxP/Risk Assessment
Supplier Audit
Computer System Definition
Operational Procedures
Maintenance
Training
Change Management
Periodic Review
System Test Specifications
Instruction Manuals
Design Review(s)
• Electronic Records & Signatures Review
• Source Code Review
• Configuration Review
Requirements Traceability Matrix
System Validation Protocols
Validation Report
System Design Specification
• Hardware Specification
• Reports & Screens Specification
• Database Design Specification
• Interface Specification
Validation
User
Vendor

FIGURE 41.2 Computer System Environments.

System Documentation

System documentation consists of the following:

Supplier Documentation

- System specifications
- Instruction manuals (for system operation and maintenance)
- Design documentation including database structure
- Flow charts/flow diagrams used for business process modeling
- Test data sets

Validation Documentation

The validation documentation set will include all system validation protocols, duly approved and with records appended, which demonstrate that all specified acceptance criteria for the specifications, reviews, and testing have been met. These include:

- Validation Plan
- GxP Assessment/Risk Analysis
- Design Review documentation (e.g., RTM, Source Code Review reports, Configuration Review reports, etc.)
- System validation protocols including installation and operational performance
- Validation Report

The final Validation Report will summarize the completed validation activities and confirm that the requirements documented in the Validation Plan have been met. The Validation Report will be the vehicle for allowing the database system to be released for use in the live environment.

User Documentation

- Computer System Definition/URS (while this document may be prepared by others on the user's behalf, it remains the responsibility of the blood establishment)
- Standard Operating Procedures (SOPs)

While both supplier and validation documentation may provide the basis for user documentation, a comprehensive set of SOPs should be generated covering all features of database operation. These procedures would cover all aspects of system use and maintenance, from normal operation through to business continuity in the event of major system failure.

For any blood establishment system, key considerations will be confidentiality and data integrity of records. Key topics for SOPs, therefore, include:

- **Access security: physical and logical**
  This SOP should detail any physical measures which may enhance system security such as building access controls and also any measures, be they procedural or technical, required to enhance security at remote input terminals. With 21st century database systems, the provision of logical access controls to meet the requirements of, among others, 21 CFR Part 11[11] should not be an issue. Consideration should be given to the use of user profiles or other such additional measures to enhance security on sensitive files such as donor confidential/personal information.
- **Data backup, archiving, and restoration**
  The importance of effective procedures which are followed according to the specified schedules cannot be overemphasized. These SOPs need to detail not only instructions for backing up records to the network or on to portable media devices but also physical considerations such as labeling of media, storage locations, and a clear split of responsibilities between the system owner, system administrator, and other organizational groups such as QA, IT, and Document Control. In order to prove the process, a regular schedule should be followed for demonstration so that the restoration of archived records back to the system can be achieved and the integrity of the record data can be confirmed. Consideration should also be given to transfer/further copying of data for long-term record archiving.
- **Business Continuity Plan(s)**
  Also known as Disaster Recovery procedures. These detail responsibilities and procedures in the event of major system maloperation/failure, e.g., database corruption, and need to be periodically reviewed/rehearsed and proven effective.

Training

Training record deficiencies have been noted during regulatory inspections.

As with any GxP system project, the SOPs required to assure maintenance of the validated state should be identified at an early part of the life cycle and a program established to train all system users before "go-live" of the database and also to capture any new system users.

During this training, the importance of the security and access procedures as a safeguard on the integrity and confidentiality of donor and blood data should be stressed. As per 21 CFR Part 11.10, 11.200, and 11.300 all system users should clearly understand their responsibility for actions carried out under their names.

While all new system users should be fully trained in their computer system duties and responsibilities as part of their induction or transfer to the role, regular refresher training should also be carried out, particularly after database modifications or functionality enhancements.

MAINTAINING THE VALIDATED STATE

The release of a database system for GxP operations is a major milestone, which will normally be documented in a Validation Report referring back to the system Validation Plan. The ongoing challenge after the approval of the Validation Report is the maintenance of the validated state in what is a potentially continuously changing environment.

The maintenance of the validated state entails an effective compliance management system that includes (but is not limited to):

- Document (supplier documentation, SOPs, etc.) management and maintenance.
- System maintenance — this is to include both preventative and unplanned activities on the database and associated hardware, e.g., server.
- Record reviews — routine reviews of master database transaction logs, audit trail, and other system logs. These will be documented and any appropriate corrective measures actioned and closed out under incident management.
- Change management system — this critical GxP system should be under a formal change control system for any and all hardware and software changes. All changes, even to the addition of what appears to be an innocuous SQL query, shall be formally specified, risk assessed (at a minimum for GxP impact), and an implementation plan agreed upon (together with a plan for system/function testing and (re)validation as appropriate).
- QA consultation and sign-off on all GxP impacting changes assure both independence and capture of the necessary validation input.
- QA audits — to be performed by specified (independent) personnel as per the agreed schedule. These audits will evaluate the use and performance of the system in its operational and maintenance environment and will also address system users: their competency assessment, training, accreditation, proficiency, and supervision.
- Periodic review — these should be conducted at least annually and are intended to evaluate the current compliance status of the system and identify any need for any system revalidation due to an accumulation of changes to the system or its functionality.

USER RESPONSIBILITIES

Two key user roles/responsibilities are:

System Administrator

Much of the responsibility for the operation and management of access security controls will devolve on the assigned System Administrator. The System Administrator is the person who will effect the measures required to maintain and operate the database within a compliant framework. These are:

- Assignation of user privileges
- Addition/removal of users
- Password management
- Data backup, archiving, and restoration
- Periodic running of utilities or diagnostic software (to monitor system use and performance and/or check for unwanted duplicate or discrepant data)

This person should be clearly identified within the blood establishment and be provided with the appropriate (documented) training and resources to carry out these tasks.

System/Database Owner

This is another key individual who will be responsible for maintaining the validated state of the computer system following go-live. "Buy-in" to the delivered solution and the measures required to maintain its compliance can be greatly facilitated by involvement of this person right at the start when user needs and requirements are defined.

CONCLUSION

The general trend toward e-commerce and e-operations of a range of facilities across the pharmaceutical sector continues to stimulate increased use of database systems within blood establishments. This reflects greater experience within the supplier community of larger and more distributed applications and thus a larger body of evidence as to the robustness of these installations and the security and integrity of the records which they handle.

In parallel with these developments, the regulatory environment is increasingly supportive of paperless operations and record systems, provided they can be demonstrated to be equally compliant with predicate rule requirements.

Any blood establishment proposing to install, or significantly modify, a database system for assisting in decision making and the management of data associated with blood products and their donors must comply with a number of different regulations, which may appear daunting in the responsibilities they place on them and their suppliers. Failure to validate the current regulatory expectations can have significant financial and patient safety implications.[14] However, regulatory expectations can be satisfied by adopting a structured life-cycle approach to the validation of the system. As outlined above, much of what is required is familiar as Good Software Development Practice, supplemented by those additional activities and documentation appropriate to the validation of computer systems in line with current regulatory expectations.

The trend toward database applications for more sophisticated data manipulation, analysis, and reporting is therefore expected to continue and to offer the potential for flexibility and operating efficiency benefits to blood establishments without any increased risk to public health.

REFERENCES

1. U.S. Public Health Service (PHS) Act.
2. U.S. Code of Federal Regulations Title 21: Part 600, Biological Products: General, Rockville, MD.
3. U.S. Code of Federal Regulations Title 21: Part 606, Current Good Manufacturing Practice for Blood and Blood Components, Rockville, MD.
4. U.S. Code of Federal Regulations Title 21: Part 610, General Biological Products Standards, Rockville, MD.
5. U.S. Code of Federal Regulations Title 21: Part 210, Current Good Manufacturing Practice in Manufacturing, Processing, Packing, or Holding of Drugs: General, Rockville, MD.
6. U.S. Code of Federal Regulations Title 21: Part 211, Current Good Manufacturing Practice for Finished Pharmaceuticals, Rockville, MD.
7. U.S. Code of Federal Regulations Title 21: Part 800, Medical Devices: General, Rockville, MD.
8. U.S. Code of Federal Regulations Title 21: Part 820, Good Manufacturing Practice for Medical Devices: General, Rockville, MD.
9. U.S. Code of Federal Regulations Title 21: Part 640, Additional Standards for Human Blood and Blood Products, Rockville, MD.

10. *Rules Governing Medicinal Products in the European Community*, Volume IV, Good Manufacturing Practice for Medicinal Products, Office for Official Publications of the EC, Luxembourg, 1992.
11. U.S. Code of Federal Regulations Title 21: Part 11, *Electronic Records; Electronic Signatures*, Food and Drug Administration, Rockville, MD.
12. ISPE, Good Automated Manufacturing Practice (GAMP4) Guide for Validation of Automated Systems, December 2001.
13. FDA (2002), *General Principles of Software Validation; Final Guidance for Industry and FDA Staff*, Center for Devices and Radiological Health, Center for Biologics Evaluation and Research, January.
14. FDA (2002), Inspectional Observations Issued to Biomedical Services, American Red Cross, Form 483, December.

42 Case Study 24: Process Analytical Technology

Guy Wingate, GlaxoSmithKline

CONTENTS

Process Analytical Technologies (PAT) provide "systems for analysis and control of manufacturing processes based on timely measurements during processing of critical quality parameters and performance attributes of raw and in-process materials and processes to assure acceptable end product quality at the completion of the process."*[1] As such, PAT offers higher manufacturing efficiency but requires a change in the established quality paradigm from analytical QC on finished goods to a practice of process QA philosophy. This brief case study considers the implications for supporting computer systems.

FOUNDING PRINCIPLES

Although not formally defined as such, PAT is founded on three basic principles concerning:

- Process Analysis — registered critical control points
- Process Control — control of product variation
- Process Understanding — process validation and risk management

Process analysis should be based on moving away from testing to document quality toward "quality by design." Current process characterization does not facilitate a smooth transition through drug development, registration, manufacturing scale-up, and subsequent process improvement. It

* The MHRA has suggested that the acronym PAT might alternatively be used for Process Assurance Technologies.

is vital that critical control points and associated critical parameters registered with regulatory authorities support demonstrable process control but not impede ongoing process capability.

Process control should be based on continuous quality assurance; all key stages should be quality assured and monitored for acceptability. Contemporaneous quality decisions are facilitated in anticipation of potential product failures if no corrective action was taken. The whole manufacturing process becomes more tightly coupled with product quality.

Process understanding is based on a significantly better understanding of the changing product characteristics during the manufacturing process. The process capability is registered including valid process variation. Critical control points and associated critical parameters are defined. No extraneous measurements and data are registered with regulatory authorities. Potential risks associated with change are thereby mitigated.

PAT removes redundant waiting time for finished goods testing, replaces it with a continuum of in-process sampling, and facilitates enhanced closed-loop control of processes.

TOPOLOGY

The topology of PAT systems varies depending on the precise needs of the process it supports. An example topology is presented in Figure 42.1 for the purpose of discussion. It consists of:

- Data Acquisition (process measurement)
- Chemometrics (multivariate data manipulation)
- On-line Prediction (product models)
- User Reporting (contemporaneous reporting)
- Data Historian (archiving)
- Interfacing to other systems

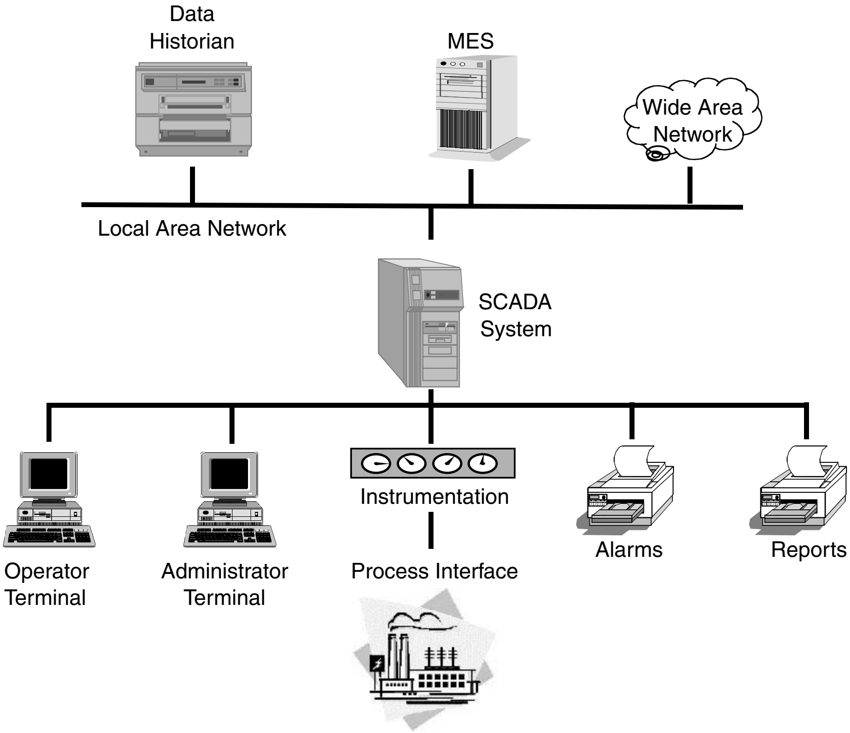


FIGURE 42.1 Example PAT Topology.

Data acquisition is provided by instrumentation. Chemometrics, on-line prediction, and reporting are typically supported by some kind of Supervisory Control And Data Acquisition (SCADA) system and/or Manufacturing Execution System (MES). This may be a COTS product or a custom development. Data Historian is likely to be a separate interfaced application. To support this and exchange of information with other systems an "open systems" approach should be adopted.

Data Acquisition

The development of sensor technology to allow a range of nonintrusive measurements is a significant enabler of PAT. Examples of nonintrusive instruments are:[2]

- Near Infra-Red (NIR) spectroscopy
- FT-IR
- Raman spectroscopy
- UV/visible spectroscopy
- Acoustic Emission spectroscopy
- Particle size characterization
- X-ray tomography
- NMR
- Mass spectrometry

The selection of instrumentation will be dependent on the chosen product characteristics being measured for the product form. Product characteristics can be categorized into physical structure, chemical identity, and homogeneity. The focus should be on critical process parameters affecting end product quality, although there may also be some process performance measurements.

Chemometrics

Chemometrics provides a means of contemporaneously analyzing sample data to optimize processes. Data sources may be spectral, wet chemistry or a combination.

Chemometrics uses multivariate, multidimensional data to generate product specific models. These models are the basis against which future data can be compared to allow both qualitative and quantitative predications to be made.

Product specific models require management from initial creation through approval, use, refinement, and eventual withdrawal and archiving. Data from different sensors must be correctly collated and built into the models for the products they support. As the body of knowledge concerning a process increases, so the product specific model can be refined and made ever more robust.

On-Line Prediction

Potential product rejects should be anticipated so that intervention can be prompted and corrective action taken in a timely manner to avoid final product rejection. Rejects may be due to product being Out Of Specification (OOS) or the model being too sensitive. Data associated with the model being too sensitive should be analyzed and used to refine the model. Once the model has sufficient status it can be used as part of the registration of a product with a regulatory authority.

User Reporting

PAT operators do not solely react to alerts and warnings but rather contemporaneously interact to what process is doing. Real-time reporting on product analysis is therefore required to enable immediate quality critical decisions to take place. It should be possible to configure standard reports for both on-screen display and printing. Trends will allow proactive management and optimization of in-process manufacturing.

Data Historian

Potentially huge volumes of data might need to be archived to satisfy regulatory electronic record requirements. By understanding what constitutes critical parameters as to the amount of data requiring long-term storage, the volume of data archiving can be significantly reduced.

VALIDATION

Computer Validation

The basic requirements for computer validation are unchanged from the principles outlined earlier in this book. A life-cycle approach should be adopted as discussed elsewhere. The main computer elements in Figure 42.1 consist of:

- Instruments
- SCADA/MES Systems
- Data Historian

Instruments should be validated. Data transfer after acquisition should have integrity checking. This is normally facilitated through industry standard protocols. Supplier auditing might be appropriate for new instrument developments.

SCADA/MES should be validated. There is likely to be a mix of COTS software and bespoke code. The correct operation of statistical analysis software and predictive control software must be assured. User interfaces should receive particular attention. User reports should be defined and tested. Custom programming such as macros should conform to good programming practices and be subject to source code reviews. Interfaces between systems should also be specified and tested.

Data Historians should be validated. Data must be protected from unauthorized and unintentional modification. This might be achieved through locking down data (i.e., no subsequent write permissions given after data created). The requirement for any audit trails needs to be specified. Data retention requirements need to be defined in accordance with company policy and regulatory requirements. Software used to retrieve data should be validated. Any dependencies on storage media need to be understood and managed. Backup and restore are key processes. System interfaces should also be specified and tested.

All computer system elements should be maintained under change control and configuration management. Inspection readiness will depend largely on being able to demonstrate an understanding of critical control points and critical parameters affecting product quality and safety exist, how they are supported by computer systems, and how any change is managed. Hazard Analysis and Critical Control Point (HACCP) is a useful method for understanding process controls. More detail on validating different computer systems can be found in the other case studies in this book.

Process Validation

Process validation will change dramatically compared to traditional concepts. Conventional validation is actually based in practice on QC, e.g., three batch runs. This approach works if conditions do not change, but in reality they often do as part of process improvement. Validating all possible process ranges can be cumbersome if not impossible. In these circumstances process validation does not help development of robust processes, rather it often confirms a lack of a robust process. Such late identification of issues tends to lead to rework or rejection of product batches. PAT should bring about a fundamentally better understanding of processes and hence control. The challenge is to determine if advanced monitoring and control has supplanted the value of process validation in assuring product quality.[3] Validation in essence would be achieved through inherent process capability that is continually proven by successive successful product batches.

ELECTRONIC RECORDS AND ELECTRONIC SIGNATURES

Electronic records requiring particular regulatory control should be identified based on critical process control points and associated critical parameters that directly impact product quality or product safety. A defined process should be used to conduct this analysis, and it should be one that builds on or is complementary to any assessment conducted as part of product registration. Consistency is key. There may be additional records identified by predicate rules but care must be taken not to extend beyond these records.[4] A risk assessment should be conducted to determine appropriate electronic record management controls such as audit trail and archiving.[5] Electronic records will need to be archived for retention periods specified in predicate rules. Other data related to process performance rather than product quality or product safety requires only basic data maintenance and may be retained for much shorter periods before being purged.

The electronic record strategy is likely to be focused on the role of the Data Historian. General audit trail requirements may be satisfied by the implementation of a transaction log. Particularly critical records may warrant electronic audit trails implemented for individual records. Consideration should also be given to how copies of regulated records, including audit trails, may be provided during inspections and submissions to regulatory authorities.

Finally, a decision will need to be taken on how best to handle the application of signatures where this is required. The FDA and other authorities will accept hybrid signatures applied to printed copies so long as it can be demonstrated that the equivalence of electronic and printed copies is maintained. Other systems might apply electronic signatures in which case there is no requirement to take a paper copy so long as electronic records with their signatures are secure and archived. The requirements for electronic records and electronic signatures are discussed more fully elsewhere in this book.

CONCLUSION

The principles extolled by PAT are not new and there are many examples of successful implementation from GlaxoSmithKline, Pfizer, and AstraZeneca over the past 10 to 20 years.[2] The difference now is the recognition of the PAT philosophy as an approach, and the widespread available technology that can make this reality. The way product quality is addressed could be revolutionized. Only time will tell.

REFERENCES

1. FDA (2002), Process Analytical Technologies Initiative, U.S. Food and Drug Administration.
2. Royal Pharmaceutical Society and American Association of Pharmaceutical Scientists (2003), The Key for Achieving New Standards of Manufacturing Excellence and Regulatory Compliance: Process Analytical Technology, Eighth Arden House European Conference, March 24–26, London.
3. PhRMA (2003), A Risk-Based Approach to cGMPs, White Paper, Pharmaceutical Research and Manufacturing of America.
4. U.S. Code of Federal Regulation (1997), *Electronic Signatures; Electronic Records*, Title 21: Part 11, Food and Drug Administration, Rockville, MD.
5. FDA (2003), Part 11, Electronic Records; Electronic Signatures — Scope and Application, Guidance for Industry, Draft for Comment, February.

Glossary

Acceptance Criteria ANSI/IEEE (1983): The criteria that a software product must meet to successfully complete a test phase or to achieve delivery requirements.

Actuator FDA (1995): A peripheral output device that translates electrical signals into mechanical actions, e.g., a stepper motor that acts on an electrical signal received from a computer system to turn its shaft a certain number of degrees or a certain number of rotations

Alpha Testing FDA (1995): Acceptance testing performed by the customer in a controlled environment at the developer's site. The software is used by the customer in a setting approximating the target environment with the developer observing and recording errors and usage problems.

As-Built FDA (1995): Pertaining to an actual configuration of software code resulting from a software development project.

Assembly Language FDA (1995): A low-level programming language that corresponds closely to the instruction set of a given computer, allows symbolic naming of operations and addresses, and usually results in a one-to-one translation of program instructions (mnemonics) into machine instructions.

Audit GMA-NAMUR (1996): An activity to determine through investigation the adequacy of, and adherence to, established procedures, instructions, specifications, codes, and standards or other applicable contractual and licensing requirements, and the effectiveness of implementation.

Garston-Smith (1997): An independent review for assessing compliance with software requirements, specifications, baselines, standards, or procedures.

Automation System A system based on a computer technology with input devices (e.g., sensors), output devices (e.g., actuators), and communication links (e.g., telemetry and cable networks) that are collectively designed to perform a specific function or group of functions (e.g., control, protection or monitoring). Automation systems may be linked into larger integrated systems. [Defined for this book.]

Baseline FDA (1995): A specification or product that has been formally reviewed and agreed upon, that serves as the basis for further development, and that can be changed only through formal change control procedures.

Batch Record IQA (1994): Documents (including those stored in photographic and electronic form) that record stages in the manufacture of a batch, details of ingredients and process equipment used, methods followed, in-process controls carried out, test results obtained, dates of manufacture, and testing and history of the storage of the pharmaceutical raw material.

Beta Testing FDA (1995): Acceptance testing performed by the customer in a live application of the software at one or more end user sites in an environment not controlled by the developer.

Bespoke Software GAMP (1996): A system produced for a customer, specifically to order, to meet a defined set of user requirements. [Note: Bespoke code includes so-called standard software where the version of the software to be used has not been market-tested over a period of time by other customers.]

Black Box Testing See Functional Testing.

Business Continuity Planning A documented process by which the recovery and continuation of critical business functions in the presence of events which significantly disrupt business operations. [Defined for this book.]

Bomb FDA (1995): A Trojan horse that attacks a computer system upon the occurrence of a specific logical event ("logic bomb"), the occurrence of a specific time-related logical event ("time bomb"), or something that is hidden in electronic mail or data and triggered when read in a certain way ("letter bomb").

Bootstrap FDA (1995): A short computer program that is permanently resident or easily loaded into a computer and whose execution brings a larger program, such as an operating system or its loader, into memory.

Calibration FDA (1995): Ensuring continuous adequate performance of sensing, measurement, and actuating equipment with regard to specified accuracy and precision requirements.

Certification FDA (1995): In computer systems, a technical evaluation, made as part of and in support of the accreditation process that establishes the extent to which a particular computer system or network design and implementation meet a prescribed set of requirements.

Change Control FDA (1995): The processes, authorities for, and procedures to be used for all changes that are made to the computerized system and/or the system's data. Change control is a vital subset of the Quality Assurance program within an establishment and should be clearly described in the establishment's Standard Operating Procedures.

GAMP (1996): A formal system by which qualified representatives of appropriate disciplines review proposed or actual changes that might affect a validated status. The intent is to determine the need for action that would ensure and document that the system is maintained in a validated state.

OECD (1995): Ongoing evaluation and documentation of system operations and changes to determine whether a validation process is necessary following any changes to the computerized system.

CHAZOP Computer HAZard and OPerability study to assess the threats and their control between Automation Systems, their users and operational environments, and the manufacturing process. CHAZOP studies for IT systems concentrate on the threats and their controls affecting data integrity. [Defined for this book.]

Client-Server FDA (1995): A term used in a broad sense to describe the relationship between the receiver and provider of a service … a networked system where front-end applications, as the client, makes service requests upon another networked system.

Code See Software.

Comment FDA (1995): In programming languages, a language construct that allows explanatory text to be inserted into a program and that does not have any effect on the execution of the program.

Commercial Off-The-Shelf (COTS) Products Versions of products that have been commercially available for at least 6 months and are widely used. Beta releases of products that are still under supplier evaluation and COTS products specifically customized (rather than configured) for application are excluded from this definition. [Defined for this book.]

Computer Aided Software Environments (CASE) Tools designed to support the analysis and design phases of the software development life cycle. The tools are usually oriented toward the support of graphical notations. [Defined for this book.]

Computer System See Automation System.

Computer Virus A program that alters other programs to include a copy of itself and executes when the host program is executed. The execution of a virus program compromises a computer system by performing unwanted or unintended functions that may be destructive. [Defined for this book.]

Computerized System A computer system plus the controlled process it operates. [Defined for this book.]

Configuration FDA (1995): The arrangement of a computer system or component as defined by the number, nature, and interconnection of its constituent parts.

Configuration Parameters Parameters that provide control values for computerized equipment. Configuration includes operating parameters (e.g., drug product manufacturing recipes, set-points) and system environment parameters (e.g., file names, directory structures). Configuration provides a method to accomplish specific functionality without using a programming language. [Defined for this book.]

Crash FDA (1995): The sudden and complete failure of a computer system or component.

Critical Impact Computer systems have a critical impact if failure or latent design flaws can result in injury or illness to the consumer of the drug that:

- Is life threatening
- Results in permanent impairment of body function
- Results in permanent damage to body structure
- Necessitates medical or surgical intervention to preclude the above

[Defined for this book.]

Critical Parameter A process parameter that may cause significant variation in the quality of a finished product. [Defined for this book.]

Critical Process HPB (1998): A process that may cause significant variation in the quality of a finished product.

Critical Step A step in a process that may cause significant variation in the quality of a finished product. [Defined for this book.]

Data Integrity FDA (1995): The degree to which a collection of data is complete, consistent, and accurate.

Data Validation FDA (1995): A process used to determine if data are inaccurate, incomplete, or unreasonable. The process may include format checks, completeness checks, check key tests, reasonableness checks, and limit checks.

Dead Code FDA (1995): Program code statements that can never execute during program execution. Such code can result from poor coding style, or can be an artifact of previous versions or debugging efforts. Dead code can be confusing, and is a potential source of erroneous software changes. Dead code is program logic that cannot execute because the program path does not permit the logic to be reached. Newly developed programs should be reviewed for the presence of dead code. Dead code must be removed prior to compilation and submission for production implementation. In instances where program logic becomes dead code as a result of program modifications, the associated dead code should be removed from the program before recompilation and submission to the production implementation. Commented source code is not dead code because it is ignored by the compiler and does not become program logic. Code rendered inaccessible by configuration (e.g., switches, parameters, calls, etc.) is not dead code because this code is intended to be available for use depending on the need of a particular implementation. Similarly, code residing within a standard library, which is not accessed by the calling program, is not considered dead code because this code is intended to be available for use depending on the need of a particular implementation. Code that has been included for the purposes of testing or for later diagnosis during support work, and which can be configured "on" or "off" is not regarded as dead code. If the code is configurable for use in many different projects, each with a different configuration of options, the unused options should not be removed; however, the source code and configuration review and testing processes must demonstrate that the correct options have been correctly deselected and do not function.

Debugging FDA (1995): Determining the exact nature and location of a program error, and fixing the error.

Design FDA (1995): The process of defining the architecture, components, interfaces, and other characteristics of a [automation] system or component.

Design Qualification (DQ) GAMP (1996): Formal and systematic verification that the requirements defined during specification are completely covered by the succeeding [design] specification or implementation.

GMA-Namur (1996): Formal and systematic verification that the requirements determined at the functional specification phase were completely met in the subsequent specification or implementation phase and that the higher authority of guidelines or laws have been taken into account.

Design Review Phrase synonymous with DQ used in relation to computer systems.

Desktop GAMP (2001): Represents the end user workstation and local software environment. Normally provides a Graphical User Interface (GUI) front-end menu providing users with access to required applications. Many desktop environments can be reconfigured by the end user.

Desktop Build Set of software on end user workstations making up desktop environment. Also referred to as desktop configuration. [Defined for this book.]

Disaster A sudden, unplanned calamitous event that creates an inability on an organization's part to provide critical business functions for some period of time, which results in great damage or loss. [Defined for this book.]

Electronic Signature OECD (1995): The entry in the form of magnetic impulses or computer data compilation of any symbol or series of symbols, executed, adapted, or authorized by a person to be equivalent to the person's handwritten signature.

Embedded System GAMP Forum (1996): A system, usually microprocessor or PLC based, whose sole purpose is to control a particular piece of automated equipment. This is contrasted with a stand-alone computer system.

Emulation FDA (1995): A model that accepts the same inputs and produces the same outputs as a given system. To imitate one system with another.

Escrow ACDM/PSI (1998): A legal term in Anglo-American law. A written agreement, constituting evidence between two or more parties (in this case the supplier and purchaser), that is given to a third party with instructions (in this case, to deliver source code and associated documentation) to be executed only upon a future condition (in this case, the supplier going into receivership).

Failure Mode Effects Analysis (FMEA) A technique used to define, identify, and reduce known or potential failures to an acceptable level. [Defined for this book.]

Firmware FDA (1995): The combination of a hardware device, e.g., an Integrated Circuit, and computer instructions and data that reside as read-only software on that device. Such software cannot be modified by the computer during processing.

Functional Specification A written definition of the function that a system or system component can perform. [Defined for this book.]

Functional Testing GAMP Forum (1996): Also known as "Black Box" testing, since source code is not needed. This involves inputting normal information and abnormal test cases and then, evaluating outputs against those expected. Can apply to computer system or to a total system. [Adapted.]

Good Clinical Practice (GCP) The standard by which clinical trials are designed, implemented and reported so that there is public assurance that the data are credible, and that the rights, integrity, and confidentiality of subjects are protected. [Defined for this book.]

Good Distribution Practice (GDP) MHRA: GDP is that part of quality assurance that ensures that products are consistently stored, transported, and handled under suitable conditions.

Good Laboratory Practice (GLP) (U.K. DoH, 1995): GLP is concerned with the organizational processes and conditions under which studies are planned, performed, monitored, recorded, and reported in order to promote and maintain the quality and reliability of the test data generated.

Good Manufacturing Practice (GMP) EU (1991): That part of quality assurance which ensures that products are consistently produced and controlled to the quality standards appropriate to their intended use.

IQA (1994): [supplement EU definition with …] It concerns production, quality control, and warehousing and distribution procedures.

GMP Critical An aspect of the manufacturing process that if not properly managed can impact product quality. [Defined for this book.]

Handshake FDA (1995): An interlocked sequence of signals between connected components in which each component waits for the acknowledgment of its previous signal before proceeding with its action, such as data transfer.

Hardware FDA (1995): The physical equipment [making up a computer system], as opposed to programs, procedures, rules, and associated documentation. [Adapted.]

Hardware Design See Design.

Hardware Platform GAMP (2001): All computer hardware deployed to run software application programs. The definition covers servers, CPUs, memory devices, and peripheral controllers. [Adapted.]

Hazard Analysis See CHAZOP.

Industry Standard FDA (1995): Procedures or criteria recognized as acceptable practices by peer, professional, credentialing, or accrediting organizations.

Infrastructure GAMP (2001): All of the computer systems with their associated hardware, operating software (other than software applications), and networks used to run the business.

In-Process Control EU (1991): Checks performed during production in order to monitor and, if necessary, to adjust the process to ensure that the product conforms to its specification. The control of the environment or equipment may also be regarded as part of in-process control.

Installation Qualification (IQ) FDA (1995): Establishing confidence that process equipment and ancillary systems [including computer systems] are compliant with appropriate codes and approved design intentions, and that manufacturer's recommendations are suitably considered.

PMA (1990): Documented verification that all key aspects of hardware installation adhere to appropriate codes and approved design intentions and that the recommendations of the manufacturer have been suitably considered.

Integrated Project Support Environment (IPSE) Tools supporting the configuration management of documentation and programming during the software development life cycle. [Defined for this book.]

Major Level of Concern Computer system failure or latent design flaws potentially have a critical impact on the consumer of the drug product, operator, or bystander. [Defined for this book.]

Metadata (DOD 5015.2-STD) Data describing stored data, that is, data describing the structure, data elements, interrelationships, and other characteristics of electronic records.

Minor Level of Concern Computer system failure or latent design flaws are not expected to result in any injury or illness to the consumer of the drug product, operator, or bystander. [Defined for this book.]

Moderate Level of Concern Computer system failure or latent design flaws could result in injury or illness (but without a critical impact) on the consumer of the drug product, operator, or bystander. [Defined for this book.]

Network GAMP (2001): A network is a data communications system that links two or more computers and peripheral devices. It consists of cabling, the network hardware, and communications software.

Object Code A computer program that is the output of translated [assembler or compiler] source code. [Defined for this book.]

Operational Qualification (OQ) FDA (1995): Establishing confidence that process equipment and subsystems [including computer systems] are capable of consistently operating within established limits and tolerances.

PMA (1990): Documented verification that the equipment-related system or subsystem performs as intended throughout all anticipated operating ranges.

Patch A change made directly to object code without retranslating [assembler or compiler] the source code. [Defined for this book.]

Performance Qualification (PQ) EPA (1995): Documented verification that the process-related system performs as intended throughout representative or anticipated operating ranges.

FDA (1995): Establishing confidence that the [manufacturing] process is effective and reproducible.

Peripherals GAMP (2001): Hardware deployed to extend the capability of the hardware platform. It includes printers, modems, keyboards, tape drives, screens, and scanners.

Platform FDA (1995): The hardware and software that must be present and functioning for an application program to perform as intended. A platform includes, but is not limited to, the operating system or executive software, communication software, microprocessor, network, input/output hardware, any generic software libraries, database management, user interface software, etc.

Procedures EU (1991): Description of the operations to be carried out, the precautions to be taken and measures to be applied directly or indirectly related to the manufacture of a medicinal product.

Process Qualification FDA (1995): Establishing confidence that a process is effective and reproducible.

Product Qualification FDA (1995); Establishing confidence through appropriate testing that the finished product produced by a specified process meets all release requirements for functionality and safety.

Production EU (1991): All operations involved in the preparation of a medicinal product, from receipt of material, through processing and packaging, to its completion as a finished product.

Program See Software.

Project Plan Similar to Quality Plan but including a detailed schedule of project activities and deliverables. [Defined for this book.]

Prospective Validation FDA (1995): Validation conducted prior to the distribution of either a new [drug] product, or [drug] product made under a revised manufacturing process, where the revisions may affect the [drug] product's characteristics.

GMA-Namur (1996): Documented evidence that a system does what it was planned to do before it is used in production. Also referred to as validating new systems.

Prototyping FDA (1995): An approach to accelerate the software development process by facilitating the identification of required functionality during analysis and design phases. A limitation of this technique is the identification of system and software problems and hazards. [Adapted.]

Quality Garston-Smith (1997): The totality of features and characteristics of a product or service that bears on its ability to satisfy given needs.

Quality Assurance (QA) ANSI/IEEE (1983): A planned and systematic pattern of all actions necessary to provide adequate confidence that the item conforms to established technical requirements.

Quality Control (QC) GAMP (1996): The regulatory process through which industry measures actual quality performance, compares it with standards, and acts on the differences.

Quality Plan GAMP Forum (1996): A plan created by the supplier to define actions, deliverables, responsibilities, and procedures to satisfy the customer's quality and validation requirements.

Raw Data The first records of an action or observation saved onto a durable storage medium that are capable of being used immediately or as part of a further GMP decision or review process. Raw data may include photographs, microfilm or microfiche copies, computer printouts, magnetic media, including dictated observations, and recorded data from automated systems. Raw data excludes transient electronic data. [Defined for this book.]

GAMP Forum (1996): Any worksheets, records, memoranda, notes, or exact copies thereof that are the result of original observations and activities of a study and are necessary for the reconstruction and evaluation of a work project, process or study report, etc. Raw data may be hard/paper copy or electronic but must be known and defined in the system procedures.

FDA (21 CFR 58): Raw data means any laboratory worksheets, records, memoranda, notes, or exact copies thereof that are the result of original observations and activities of a nonclinical laboratory study and are necessary for the reconstruction and evaluation of the report of that study. In the event that exact transcripts of raw data have been prepared (e.g., tapes that have been transcribed verbatim, dated, and verified accurate by signature), the exact copy or exact transcript may be substituted for the original source as raw data. Raw data may include photographs, microfilm or microfiche copies, computer printouts, magnetic media, including dictated observations, and recorded data from automated instruments.

Real-Time FDA (1995): Pertaining to a system or mode of operation in which computation is performed during the actual time that an external process occurs, in order that the computation results can be used to control, monitor, or respond in a timely manner to the external process.

Redundant Code See Dead Code.

Retrospective Validation FDA (1995): Validation of a process for a [drug] product already in distribution based upon accumulated production, testing and control data.

GMA-NAMUR (1996): Documented evidence that a system does what it purports to do based on an analysis of historical information. Also referred to as validating existing systems.

Revalidation FDA (1995): Relative to software changes; revalidation means validating the change itself, assessing the nature of the change to determine potential ripple effects, and performing the necessary regression testing.

GMA-NAMUR (1996): Repetition of the validation process or a specific portion of it [in response to a change].

Risk Analysis See Risk Assessment.

Risk Assessment A systematic approach to identifying the potential failures in processes and quantifying the risk they present. [Defined for this book.]

Security OECD (1995): The protection of computer hardware and software from accidental or malicious access, use, modification, destruction or disclosure. Security also pertains to personnel data, communications, and physical protection of computer installations.

Sensor FDA (1995): A peripheral input device that senses some variable in the system environment, such as temperature, and converts it to an electrical signal that can be further converted to a digital signal for processing by the computer.

Service Level Agreement A formal agreement, possibly contract, defining the services to be provided by a supplier to a customer. [Defined for this book.]

Simulation FDA (1995): A model that behaves or operates like a given system when provided with a set of controlled inputs.

Software GAMP (1996): A collection of programs, routines, and subroutines that controls the operation of a computer system. [Adapted.]

Software Design See Design.

Software Inspection FDA (1995): A manual testing technique in which program documents (including source code) are examined in a very formal and disciplined manner to discover errors, violations of standards, and other problems.

Source Code GAMP (1996): An original computer program expressed in human readable form (programming language) which must be translated into machine readable form before it can be executed by the computer.

Source code is the human readable form of program code, written in its original (source) programming language. Source code must be compiled, assembled or otherwise interpreted before it can be executed by a computer. The executable code is referred to as object code because it is not readily understandable because it exists as machine hexadecimal code. [Defined for this book.]

Source Code Review See Software Inspection.

Standard See Industry Standard.

Standard Operating Procedures FDA (1995): Written procedures prescribing and describing the steps to be taken in normal and defined conditions which are necessary to assure control of production and processes. See also Procedures.

GMA-NAMUR (1996): Instruction which describes how something is to be accomplished. SOPs regulate the operation and maintenance of a computerized [automated] system in order to use it in a correct way and also to fulfill its real purpose permanently.

Structured, detailed instructions on how to do a task to ensure consistency and compliance. [Defined for this book.]

Stepwise Refinement FDA (1995): A structured software design technique; data and processing steps are defined broadly at first, and then further defined with increasing detail.

Structural Testing GAMP Forum (1996): Also known as "White Box" testing, it involves examining the internal structure of the source code. Includes low-level and high-level code review, path analysis, auditing of programming procedures, and standards actually used, inspection for extraneous "dead code," and boundary analysis and other techniques. Requires specific computer science and programming expertise. [Adapted.]

Superfluous Code Software code that unnecessarily recalculates, rechecks, or reperforms calculations or actions that are unnecessary or that have already been done. [Defined for this book.]

Supplier GMA-NAMUR (1996): The company or group responsible for developing, constructing, and delivering a system or part of a system. A supplier can be an [equipment] vendor, a contractor [including a system's application integrator], or a consultant.

GAMP (1998): Any organization of individuals contracted directly by the customer to supply a product.

TAG Unique label identifier given to instrumentation and/or equipment. [Defined for this book.]

Testing FDA (1995): The process of operating a system or component under specified conditions, observing or recording the results, and making an evaluation of some aspect of the system or component.

Transient Data Data that have a temporary existence and is not retained. [Defined for this book.]

Trojan Horse FDA (1995): A method of attacking a computer system, typically by providing a useful program that contains code intended to compromise a computer system by secretly providing for unauthorized access, the unauthorized collection of privileged system and user data, the unauthorized reading or altering of files, the performance of unintended and unexpected functions, or the malicious destruction of software and hardware.

User GMA-NAMUR (1996): The company or group responsible for the operation of a system.

User Requirements The customer's written functional needs with regard to a computer system. [Defined for this book.]

Validation EU (1991): Action of proving, in accordance with the principles of Good Manufacturing Practice, that any procedure, process, equipment, material, activity, or system actually leads to the expected results.

FDA (1995): Establishing documented evidence that provides a high degree of assurance that a specific process will consistently produce a product meeting its predetermined specifications and quality attributes.

GMA-NAMUR (1996): Documented evidence that a specific process will consistently produce a product meeting its predetermined specification and quality attributes.

IQA (1994): The process of establishing documentary evidence which provides a high degree of assurance that any product, process, activity, procedure, system, equipment, or software used in the control or manufacture consistently meets its predetermined specification.

OECD (1995): The demonstration that a computerized system is suitable for its intended purpose.

Validation Master Plan A high-level plan coordinating a number of validation plans. [Defined for this book.]

Validation Plan GMA-NAMUR (1996): A prospective plan of action whose implementation should produce formal and documented proof that the system is validated.

Vendor PICS (1999): A company or group responsible for developing, constructing, and delivering a system or part of a system. See also Supplier.

Virus FDA (1995): A program that secretly alters other programs to include a copy of itself, and executes when the host program is executed. The execution of a virus program comprises a computer system by performing unwanted or unintended functions that may be destructive.

An independent program that can travel from computer to computer across network connections replicating itself in each computer. They do not change other programs, but compromise a computer system through their impact on system performance. [Defined for this book.]

White Box Testing See Structural Testing.

White Space Blank lines of code that have been purposely inserted into the software listing to make it easier to read. [Defined for this book.]

Wireless Device Devices, usually handheld, used for wireless data acquisition and communications. Examples include mobile phones and pagers. These devices can connect to intranet and internet services as well as facilitating dedicated communication links to host computer systems. [Defined for this book.]

REFERENCE WEBSITES

www.21part11.com	Industry View of U.S. CFR Part 11
www.abpi.org.uk	Association of the British Pharmaceutical Industry
www.acdm.org.uk	Association of Clinical Data Management
www.barqa.com	British Association of Research Quality Assurance
www.bira.org.uk	The British Institute of Regulatory Affairs
www.bsi.com	British Standard Institute
www.computervalidation.com	Independent Site
www.dashnet.com/acrpi	Association for Clinical Research in the Pharmaceutical Industry
www.diahome.org	Drug Information Association
www.eudra.org/emea.html	European Agency for the Evaluation of Medicinal Products
www.eudra.org	EMEA Home Page
www.fda.gov	FDA Home Page
www.ifpma.org/ich1.html	Internal Conference on Harmonization
www.ispe.org	ISPE Home Page (GAMP Forum)
www.ivthome.com	IVT Home Page
www.jettconsortium.org	JETT Consortium Home Page
www.labcompliance.com	Ludwig Huber's Compliance Home Page
www.mhra.gov.uk	MHRA Home Page
www.pda.org	PDA Home Page
www.phrmafoundation.org	PhRMA Home Page
www.picscheme.org	PIC/S Home Page
www.psiweb.org	Statisticians in the Pharmaceutical Industry