

Computer System Validation Guide

Introduction to the Computer System Validation

This guide was developed as a resource document to assist regulated industry Validation, Quality Assurance, Technical Services, and regulated industry professionals to identify and adapt "best practices" in their management of validation and qualification of computer systems, software, hardware and developmental practices and activities.

This guide was developed to be a concise, step-by-step set of management aids, which are consistent with industry standards. They are designed to guide implementation to the minimum recommended level of practices and standards. Local management, at its discretion, may decide that these recommended levels are insufficient for local conditions and needs and therefore require more stringent practices and controls.

The practices within the guides, when fully implemented will serve to ensure secure and cost effective operation and evolution of protocol implementation.

Suggestions for improvement to this guide are always welcome. This document is intended to be living document and will be upgraded and adapted as 'better practices' emerges. [Email All Comments](#)

John F Cuspilich, Senior Editor, GMP Publications, Inc.
Senior Consultant, FDA.COM

Introduction This introduction provides an overview of the Computer System Validation Corporate Computer System Validation (CSV) Guide.

Process Validation In 1987 the Food and Drug Administration published a document entitled '*FDA Guidelines on General Principles of Process Validation*'.

It states the following:

Process validation is 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.

Note: This definition indicates that validation can apply to any process including process managed/controlled by computer systems.

Validation

Validation is applied to many aspects of the healthcare and other regulated industries and businesses.

Examples include:

services equipment computer systems processes cleaning

In each case, the objective of validation is to produce documented evidence, which provides a high degree of assurance that all parts of the facility will consistently work correctly when brought into use.

Note: Validation requires documented evidence, if the validation process is not documented then it cannot be proved to have occurred.

Scope

This guide looks at computer systems validation only. Computer systems validation includes validation of both new and existing computer systems.

Definition of computer system

For the purposes of this guide, a computer system is defined as:

any programmable device including its software, hardware, peripherals, procedures, users, interconnections and inputs for the electronic processing and output of information used for reporting or control.

Examples of computer systems

Examples of computer systems include:

- automated manufacturing equipment
- control systems
- automated laboratory equipment
- laboratory data capture system
- manufacturing execution systems
- computers running laboratory, clinical or manufacturing database systems

Purpose

The purpose of this guide is to help you:

- identify computer systems that require validation
 - determine how to validate, and the extent of validation required, for the computer systems that have been identified
- comply with the validation requirements documented in the (CSV) in accordance with your Company's Software Systems Development Master Plan (SDMP).

Audience

This guide is intended for:

- Information Resources, Services and Technology members
- all concerned managers, auditors and regulatory personnel
- all developers of computer systems applications
- all personnel involved in computer systems procurement
- all users of computer systems involved in validation activities

Structure

This guide is divided into:

- 11 chapters
- Reference material

Chapter 1 looks at who is responsible for validation activities

Chapter 2 looks at the things you need to do before you perform a system validation

Chapters 3 outlines the validation process

Chapters 4 to 11 take a more detailed look at the validation process

The Reference material consists of a glossary, references and standards.

Note: The Glossary consists of a lexicon of validation terms.

Use of Standard terms

The terms and meanings in the Glossary have been adopted as a standard for use within this guide.

Consistent use of these terms will facilitate communication about

computer system validation throughout the Company.

Chapter 1

Validation Responsibilities

Overview

Introduction Computer system validation must be supported at different levels within the organization through policies, standards and documentation.

This chapter describes the responsibilities for validation at the following levels:

- Validation, Qualification, and Protocol Development and Executing (D&E) Personnel
- Business Sector / Unit
- Site or departmental personnel (End Users)
- Quality Assurance and Regulatory Units
- Corporate Management
- Information Management, IT or IS departments

Corporate Responsibilities

Introduction This topic looks at Development and Execution (D&E) Personal and Business Unit validation responsibilities.

D&E Personnel responsibilities

- Computer System Validation and Computer System Validation Policy

The (CSV) Computer System Validation and Computer System Validation Guide (this document), provides guidance on how to achieve conformance to the above policy.

Business Unit responsibilities Each business unit is responsible for establishing a policy on computer systems validation requirements.

Business Unit policy The Business Unit policy may apply to an entire Business Unit, or to some other logical business grouping, for example, site or

department.

The Business Unit policy should be based on and comply with the (CSV) Corporate policy as a minimum requirement.

The policy should also take into account additional Business Unit specific validation requirements. These additional requirements may result from other policies within a Business Unit or a regulatory requirement.

Note: If a Business Unit policy requires higher standards than the (CSV) Corporate policy, then the Business Unit policy should be followed.

Site or Departmental Responsibilities

Introduction This topic looks at site or departmental validation responsibilities.

Responsibilities Sites or departments are responsible for:

- Computer system validation Standard Operating Procedures (SOPs)
- System inventory and assessment
- System specific validation protocols
- System specific validation documentation

Standard Operating Procedures (SOPs) SOPs provide specific procedures for validating computer systems at a site or within a department.

SOPs must:

- Comply with the Computer Systems Validation Policy and any Business Unit policies that may apply
- be approved by the appropriate management for that site or department

System inventory and assessment Site or departmental management is responsible for compiling and maintaining details about their computer systems. This information includes identifying the systems that are being used and for what purposes those systems are being used.

The system inventory and assessment information is used to determine which systems need to be validated.

See Chapter 2 - Before You Perform System Validation for details on how to perform the system inventory and assessment.

Note: The systems that require validation will be incorporated into the Site or Departmental Validation Master Plan.

After the initial site/departmental inventory has been compiled a mechanism must be established to capture new computer systems (preferably prior to purchase / development) and to add them with the inventory and Validation Master Plan.

System specific validation protocols Validation protocols are documents associated with each system identified as requiring validation. The protocol describes the scope, procedure to be followed, responsibilities and acceptance criteria for the validation.

System specific validation documentation Validation protocols should comply with the SOPs. Documentation that verifies each validation activity must be generated and stored with the validation protocol in the appropriate archive.

Validation documentation may include:

- test data
 - summary reports
 - procedures
 - certification forms produced during the validation process
-

Chapter 2

Before You Perform System Validation

Overview

Introduction This chapter looks at the things you need to do before you perform system validation on a specific computer system.

You need to:

- Set up the Validation Committee
- Under 21 CFR Part 11 Requirements
- identify systems that require validation
- Conduct appropriate training

Note The following topics are the four steps involved in identifying systems that require validation:

- Creating an Inventory
- Identifying the Systems
- Assessing Each System for Validation
- Determining Validation Priority

Setting Up the Validation Committee

Introduction This topic looks at setting up the Validation Committee.

This is one of the first activities that you must do before you can validate a system.

Role of Validation Committee

The Validation Committee oversees *all* of the computer system validation projects at a site, or within a department.

It should be noted that in some instances this committee would be responsible for all validation projects at a site, not just computer

systems validation projects.

These validation projects will be defined in the Validation Master Plan.

Team members

The Validation Committee should be made up of representatives from the following functional areas as appropriate to the site or department:

- Users
- Quality Assurance
- Engineering
- Validation
- Information Resources
- Other relevant parties

The Validation Committee should be sponsored by one of the most senior managers within the site or department.

Responsibilities

The responsibilities of the Validation Committee include, but are not necessarily limited to:

- identifying components (including computer systems/applications) requiring validation
- prioritizing and justifying the validations to be performed
- developing the initial Validation Master Plan and producing subsequent revisions as required
- establishing a mechanism to include new computer systems in the Validation Master Plan
- establishing site specific procedures for computer systems validation
- coordinating validation projects
- resourcing validation projects, including approving the use of consultants

- reviewing the progress of individual validation projects to ensure timely execution of the Validation Master Plan
- resolving issues arising due to conflicting priorities, schedules, or resources

Validation Master Plan

Each site should establish a Validation Master Plan, which describes all the required validation activities at the site together with assigned responsibilities, priorities and timings for actions. The Validation Committee should approve this site plan.

All computer systems validation projects should either be included in this Validation Master Plan or a separate Computer Systems Validation Master Plan established.

The Validation Master Plan should be a dynamic document which at any point in time will represent:

- which systems exist on site
- which systems require validation
- who is responsible for each validation project
- the status of the individual validation projects
- the date for completion of each validation project

All upgrades and periodic reviews should also be added to this plan.

Identifying Systems that Require Validation

Introduction

Before you can validate a system, you need to identify the systems that require validation. This topic looks at all of the steps involved in identifying systems that require validation.

It also looks at the:

- scope of the procedure
- definition of hardware

- hardware owner
- system / application owners

Procedure Identifying systems that require validation consists of the following steps:

Step	Action
1	Create an Inventory - Note: The inventory must consist of all hardware and software in use within a given site or department
2	Identify the Systems
3	Assess Each System for Validation
4	Determine Validation Priority

Scope This procedure should be applied to all systems within a site or department.

Definition of hardware Hardware is defined as any programmable device including:

- mainframe
- midrange
- mini
- workstations e.g., SUN
- personal computers
- any programmable equipment used in a quality process.

Hardware owner Each computer or computerized piece of hardware should have a designated owner.

The hardware owner is responsible for:

- identifying all software residing on the hardware - both system software and application software

- maintaining the inventory whenever changes are made to the hardware
- managing the change control process for the system software and hardware and working with the system/application owners to determine the impact of the change to the system/application.

The hardware owner will be involved in step 1.

System / application owner

Each system/application should have an owner.

This owner is responsible for:

- defining the system (hardware and application)
- completing the system assessment
- ensuring that the information pertaining to their specific system is correct and complete
- updating the system assessment whenever changes are made to the system.

The system owner will be involved in steps 1, 2 and 3.

Creating an Inventory

Introduction This topic looks at creating an inventory of all hardware and software in use within a given site or department. Creating an inventory is the first step in identifying systems that require validation.

Responsibility The hardware owner and the system/application owner are responsible for step 1.

See the previous topic to have a look at their responsibilities.

List all details For each computer or computerized piece of hardware (or, if appropriate, each group or class of hardware), the operating system and all applications residing on the hardware should be listed.

Other key information may also be recorded.

Best practice It is best practice to have a single repository within the site or department for all data resulting from the inventory.

Inventory format The exact information and format of the inventory will vary according to the type of hardware and the needs of the business unit.

The following three tables provide examples of information that should be included in the inventory.

Hardware information

The following hardware information should be included in the inventory:

Information	Description
Reference number	A unique identifier for each piece of hardware.
Name	The model name of the main Central Processing Unit (CPU).
Manufacturer	Manufacturer of the main CPU.
Supplier	Supplier name of the main CPU, if different from manufacturer.
Owner	Name of person responsible for the hardware.
Departments served	The departments or Business Unit relying on the hardware.
Location	Physical location of the CPU.
Qualification status	Whether Installation Qualification (IQ) has been performed on the hardware. Mark "Qualified" if IQ has been done. Mark "Not Qualified" if IQ has not been done.

System software information The following system software information should be included in the inventory:

Operating system	Name of the operating system residing on the CPU.
Operating system version	Version of the operating system residing on the CPU.

Application software information

The following application software information should be included in the inventory:

Reference number	A unique identifier for each piece of application software.
Application name	For purchased systems/applications, use the product name. For in-house systems/applications, use the name by which generally known. For computer controlled instruments, call the application "Resident Software".
Version number	Version number of the application.
Owner	Name of person responsible for the application software.
Origin	Statement about origin. Examples: Vendor Supplied - no modifications Vendor Supplied - with modifications Bespoke/Custom
Supplier	Name and location of supplier.
Use	General category of use.

	<p>Examples: Raw Data Storage</p> <p>Raw Data Collection</p> <p>Data Processing</p> <p>Equipment Control</p> <p>System Software</p> <p>Utility</p> <p>Note: A computer system may fall into multiple categories e.g., a chromatography system provides raw data collection and storage, data processing and, in some cases, equipment control.</p>
System name	The name of the system of which the application software is a component. This may be the same as the application software name.
System component name	The name of the system component by which the application software is known.

Identifying the Systems

Introduction This topic looks at identifying the systems, the second step in identifying systems that require validation.

System definition For inventory purposes, a system is defined as:

any programmable device including its software, hardware, peripherals, procedures, users, interconnections and inputs for the electronic processing and output of information to be used for reporting or control.

Responsibility The system/application owner is responsible for identifying the systems from the inventory.

Procedure **Identifying the systems consists of the following steps:**

Step	Action
1	Identify all hardware, software and interfaced equipment that describes a system
2	Identify the systems involved in step 1
3	Evaluate each software application identified on the inventory to determine if it is part of a system or not
4	Identify and record the primary functions of the system
5	Record any additional information

Format The exact information and format for recording the functions of a system will vary according to the type of system and the needs of the business unit.

Examples of information The following table provides examples of information that should be recorded:

Information	Description
System name	The system or component name.
System owner	The name of the person responsible for the overall system, including hardware, software and interfaced equipment. This is often the manager of the department in which the system is used.
Departments served by system	Those departments or Business Unit that rely on the system.
Associated hardware	The name and reference number of all hardware associated with the system.
Associated applications	The name and reference number of all application software associated with the system.
Equipment	Any equipment associated with the system for the purpose of control or data acquisition.
Major functions	All major functions of the system. Unused major functions should be listed separately, since these may not be evaluated against quality critical criteria. A risk assessment must be made on the criticality of the function and whether the function can be turned on and off by a system administrator. NOTE: The application or system function, depending on the

application type and the use strategy and process, may be audited by regulatory agencies, and based on their recommendations or audit findings, determine that all existing functions must be qualified or validated. If a function is determined not be of a critical or usable facet of the application, and if validating or qualifying is not performed, then a statement could be drafted summarizing the reasoning for not testing.

Assessing Each System for Validation

Introduction This topic looks at assessing each system for validation, the third step in identifying systems that require validation.

Each system that has been identified must be assessed to see whether it performs quality or business critical functions, whether there are sufficient controls to ensure its performance, and whether it should be validated or not.

Responsibility The system/application owner is responsible for completing the assessment of each system and updating that assessment whenever there are changes to the system.

Procedure Assessing each system for validation consists of the following steps:

Step	Action
1	Develop a list of: <ul style="list-style-type: none"> • quality critical criteria • business critical criteria.
2	Evaluate each function against the quality critical criteria and record the outcome. Note: If a function meets the quality critical criteria, it is classed as quality critical.
3	Evaluate each function against the business critical criteria and record the outcome. Note: If a function meets the business critical criteria, it is classed as business critical.

4	<p>Identify the external controls, if any, for each quality critical and business critical function.</p> <p>Examples: External controls include:</p> <ul style="list-style-type: none"> • Parallel manual procedures • 100% data reconciliation.
5	<p>Determine whether the system requires validation according to the following criteria:</p> <p>If the system has quality critical functions, validation is mandatory.</p> <p>If the system has business critical functions, validation is recommended.</p>
6	Prepare the Assessment Report

Mandatory quality critical criteria Company processes are subject to various regulations listed in the table below. Any system or function used in these regulated areas is deemed quality critical.

Regulations	Example Authorities
<p>Good Laboratory Practice (GLP)</p> <p>Good Clinical Practice (GCP)</p> <p>Good Manufacturing Practice (GMP)</p>	<ul style="list-style-type: none"> • US Food and Drug Administration • European Union • Therapeutic Goods Administration (TGA) • Organization for Economic Co-operation & Development (OECD) <p>* There are many local country regulations. However, the authorities listed above represent the major regulatory authorities with specific regulations associated with computer systems.</p>

Examples of quality critical criteria

Examples of systems or functions that fall under these regulations are:

- Pre-clinical safety assessment/toxicology studies
- Clinical safety assessment/efficacy studies
- System functions which control equipment, and collect, store, and/or process the following types of data:

Data type	Description
Manufacturing	Manufacturing instructions Lot status Lot traceability - composition, disposition Inventory control Expiry date Label control
Process	Process control
Quality	Test methods and test specifications

	<p>Test results/QC records</p> <p>Verdicting i.e. the association of a disposition with a lot/batch/sample</p>
Study	<p>Stability trial - scheduling, data processing, sample storage and inventory</p> <p>Clinical trial - scheduling, data processing, sample storage and inventory</p> <p>Process studies</p> <p>Patient and animal records and results</p>
Equipment and facilities	<p>Environmental monitoring</p> <p>Calibration and maintenance records</p>
Data used to support regulatory submissions	<p>Stability data</p> <p>Development Summary Reports</p>
Regulatory documents	<p>Management of SOPs and protocols</p> <p>Regulatory Document Management System</p> <p>Electronic data archiving</p> <p>Indexes of archived documents</p>
Data associated with Clinical Laboratory System	<p>Patient Orders</p> <p>Samples</p> <p>Analytes</p> <p>Results</p>

Definition of manufacturing quality critical functions

The quality critical functions for manufacturing may be defined with reference to current Good Manufacturing Practices (CFR 21, Parts 210 and 211) as those functions that relate to:

"methods to be used in, and the facilities or controls to be used for, the manufacture, processing, packing, or holding of a drug to assure that the drug meets the requirements of the Act as to safety, and has the identity and strength and meets the quality and purity characteristics that it purports or is represented to possess."

Note: This definition is used as an example and synonymous definitions may also be found in the European Guide to GMP and the Australian Guide to GMP.

Business critical functions

Business critical functions relate to areas critical to the operation of the business, but are not related to functions, which are directly covered by regulatory requirements.

Any system or function which collects, stores or processes information in the following areas is designated business critical:

- product costs
- customer information
- supplier information
- payroll activities
- accounting data
- personnel information
- competitor information
- office automation

Assessment report

The assessment report identifies the quality critical and business critical functions and documents the validation status of the system.

The assessment report should include:

- whether validation is recommended and, if so, the scope of the validation
- quality critical functions
- external controls for quality critical functions
- business critical functions
- external controls for business critical functions
- current validation status of the system

- details about the validation documentation, if any

Note: The validation documentation details should include, the location and unique reference number, the identity of the person or persons who approved the validation documentation and the date of the approval.

Determining Validation Priority

Introduction This topic looks at determining validation priority, the last step in identifying systems that require validation.

When you have identified those systems that require validation, you need to determine the priority of each system validation (or validation project) within a site or department. To determine the priority of each validation, you need to perform a risk assessment of all the validation projects.

The findings should be documented in a report and include a justification for the validation priority for all of the systems.

Suggested method The table below outlines the suggested method of conducting a risk assessment.

Step	Action
1	Determine the risk assessment criteria (refer to the list below).
2	Weight or score each of the criteria.
3	Match the system to each criterion.
4	Compare the total system score with other systems and prioritise.
Note: Use the scoring system as a guide only. Professional judgement will still be required.	

Suggested criteria Consider the following when determining the risk assessment criteria:

- Company/site history and experience with the system
- complexity of the system

- industry experience with the system
 - have the regulatory authorities shown interest in this type of system either within or external to Company?
 - stage in system life cycle. e.g. is the system new or has it been installed for some time, or, is the system almost at the end of its use and will shortly be replaced?
 - criticality of system or data contained within the system in terms of patient risk and product quality
 - what products are associated with the system; are those products strategic?
 - Validated status of system
 - which department is the system used by e.g. QA, Production, Packaging, Development
 - the impact on the business if the system was not operational for a period of time
-

Chapter 3

The Validation Process

Overview

Introduction This chapter looks at the validation process.

Note: This validation process is used to validate a specific computer system. It may be done on an existing computer system or on a new computer system.

Purpose The purpose of the validation process is to provide a high degree of assurance that a specific process (or in this case computer system) will consistently produce a product (control information or data) which meets predetermined specifications and quality attributes.

The Validation Facets The validation effort consists of 5 specific facets or processes, each alone, would not constitute a validation. However, depending on the specifics of the application, system or process, would depend on which facets would be required. There following facets are:

- The Validation Master Plan (VMP)
- The Project Plan
- Installation Qualification (IQ)
- Operational Qualification (OQ)
- Performance or Process Qualification (PQ)

Types of validation The two types of validation are:

- Prospective validation: the validation of a new system as it is developed
- Retrospective validation: the validation of an existing system

Validation process	The validation process and document references are shown below:
Step	Action
1	Establish Team(s)
2	Determine Validation Activities
3	Write the Validation Protocol
4	Specify the System Development Details
5	Perform Qualification Activities
6	Develop/Review Controls and Procedures
7	Certify the System
8	Review Periodically

Steps 1 to 8

Introduction This topic provides an overview of the validation process.

Step 1: Establish team(s) The first step in the validation process is to establish the System Validation Team and if required the System Validation Steering Team.

These are the teams that will be responsible for the validation process.

Step 2: Determine validation activities The second step in the validation process is to determine and record all of the validation activities that will be undertaken in order to validate the computer system.

The validation activities are the exact details or activities that will be required for each of the steps in the validation process. The output from this activity will be the Validation Plan.

Example: At step six of the validation process (*Develop/Review Controls and Procedures*) the exact controls and procedures that will be required to keep the computer system validated will be

determined and recorded.

Note: The type and number of validation activities will depend on the nature of the computer system that is being validated.

**Step 3:
Write the
Validation
Protocol**

The third step in the validation process is to write the Validation Protocol.

The Validation Protocol describes the procedure and the steps within the procedure that will be followed in order to validate the system.

The Validation Protocol must also provide a high level description of the overall philosophy, intention and approach.

**Step 4:
Specify the
system
development
details**

The fourth step in the validation process is to specify the system development details.

You should specify to the supplier or developer of a system that they must have:

- a good methodology in order to develop a system
- a formal quality management system for the development, supply and maintenance of the system

You may need to specify to the supplier or developer the types of items you want to see - this could be done in the form of a Quality Plan. These items will help you ensure that the supplier or developer has a good methodology and formal quality management system in place.

Examples:

Items that will help you ensure a good methodology and formal quality management system include:

- quality management procedures
- life cycle definition
- specifications, for example user requirements specification and functional specification
- documentation controls and various items of documentation, for example user manuals and administrator documentation

- testing procedures

If the computer system is a new one, then the system development requirements will be identified prior to system selection/development.

If the computer system is an existing one, then the system development requirements will still be identified and used as a basis against which to evaluate the system.

The fifth step of the validation process is to perform the qualification activities, which are comprised within the validation process.

Some examples of these qualification activities include:

**Step 5:
Perform
qualification
activities**

- Supplier audit
- Specification qualification
- Design qualification
- Installation qualification
- Operational qualification
- Performance qualification

**Step 6:
Develop /
review
controls and
procedures**

The sixth step of the validation process is to develop/review controls and procedures.

If the computer system is a new one, then you will need to develop the controls and procedures, or check the suitability of existing generic procedures applicable to the site or department.

If the computer system is an existing one, then you will need to review the controls and procedures and update them if required.

**Step 7:
Certify the
system**

The seventh step of the validation process is to certify the system.

This step is where you certify that the validation deliverables have

met the acceptance criteria that were described in the Validation Protocol.

When you certify the system you should prepare a validation report. The validation report should outline the details of the validation process.

Examples of details that should be outlined include:

- what was done and the results that were obtained
- any special considerations
- whether the validation procedure (as described in the Validation Protocol) was followed
- a summary of all documentation that was generated
- the location of the validation documentation
- the retention period for the documentation

**Step 8:
Review
periodically**

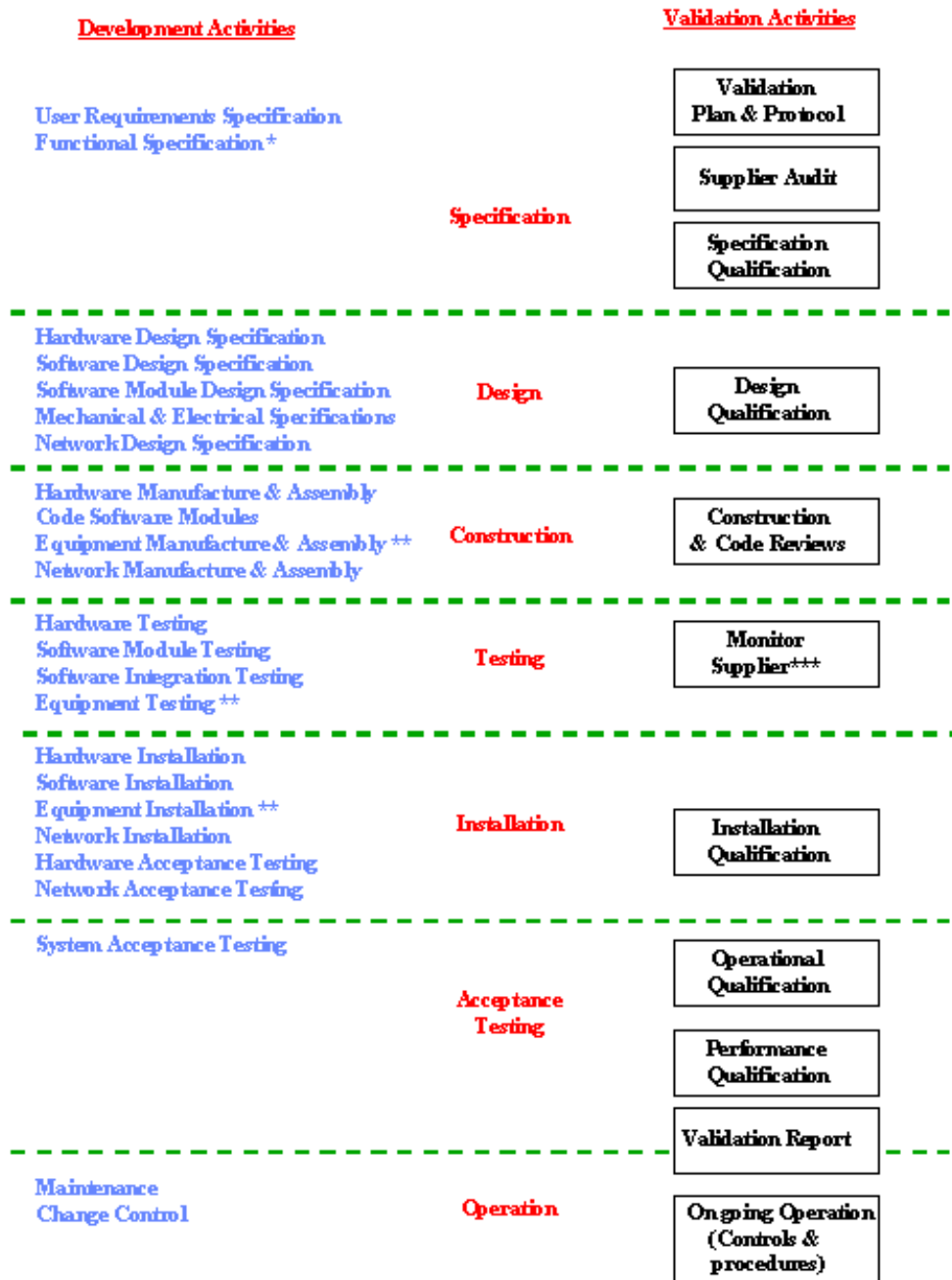
The eighth and final step of the validation process is to review the system validation periodically.

The system should be reviewed periodically to provide additional assurance of validation.

There should be documentation outlining the details of how the review is to be done and what the review should cover.

The end result of a review should be a summary of the review and a recommendation as to what to do next.

Timing and Documentation



Introduction Timing

This topic looks at the timing of the validation process and documentation. Ideally, the validation process begins at the inception of the system selection or design process. It then proceeds alongside the system development and is completed prior to implementation of the system.

Many aspects of computer systems validation are just "Good Informational Resources (IR) Practice" and as such should occur anyway during the implementation of a system.

For many reasons, a system may not have been validated until after it has been in use for some time. The basic validation process is the same as for a new system. The timing of some of the validation activities may, however, differ.

Note: Retrospective validation is becoming increasingly unacceptable to regulatory inspectors. New systems should be validated before use.

Timing for a new system

The steps in the validation process, and their associated validation activities are performed in parallel with the system development life cycle and reference the development documentation as it is produced.

Timing for an existing system

For existing systems, the validation activities will still follow the development life cycle but will reference the development documentation retrospectively.

Example

An example of the parallel between system development and validation activities is shown below.

* Functional Specification can comprise mechanical, electrical and software functional specification for systems embedded in equipment

** Systems embedded in equipment with significant control and monitoring instrumentation

*** Testing carried out by supplier can form part of subsequent qualification activities if adequately controlled. This can help reduce the amount of testing needed later, particularly at operational qualification.

Documentation

Every step in the validation process, and the activities within the steps, requires documented evidence that the steps or activities have been completed.

The table below shows the documents that must be generated at each step.

Note: In some cases some of these documents may not be required.

Step	Action	Documents Generated
-------------	---------------	----------------------------

1	Establish Validation Team(s)	Team Charter Terms of Reference Role Definition Team Organization Chart
2	Determine Validation Activities	Validation Plan
3	Write the Validation Protocol	Validation Protocol
4	Specify the System Development Details	Systems Development Life Cycle documentation
5	Perform qualification activities	Supplier Audit Report In-house Audit Report Source Code Review Report Specification Qualification Report Design Qualification Report Installation Qualification (IQ) Protocol IQ Results IQ Summary Report Operational Qualification (OQ) Protocol OQ Results OQ Summary Report Performance Qualification (PQ) Protocol PQ Results PQ Summary Report
6	Develop/Review Controls and Procedures	SOPs (Standard Operating Procedures) Training procedures Training records
7	Certify the System	Validation Report Validation Certification
8	Review the System Validation Periodically	Periodic Review Procedure Periodic Review Audit Report

Chapter 4

Establish Team(s)

Overview

Introduction This chapter looks at establishing the System Validation Steering Team and the System Validation Team. This is the first step in the validation process.

Note: The names of these teams may vary according to the preference of the site. For the purposes of standardization, this document will always use the above names.

Validation Protocol The way in which the members of these teams fit into the organization and their responsibilities should be documented in the Validation Protocol.

Relationship to Validation Committee Chapter 2 described the formation of a Validation Committee. The System Validation Steering Teams and the System Validation Teams within a site or department can be positioned as follows:

Existing system The organizational structure for the validation of an existing system is the same as for a new system. If the system is already installed and validation is required, then the System Validation Steering Team and the System Validation Team may be formed at that time.

System Validation Steering Team

Introduction This topic looks at the System Validation Steering Team. The System Validation Steering Team oversees *a specific* computer system validation.

Team members The System Validation Steering Team should be made up of representatives from:

- Quality Assurance
- Information Resources

- Department in which the system is used
- Group managing the project (if different from the user department)

Note: For small systems, the Validation Steering Team may be the same as the Validation Committee (See, *Chapter 2 - Before You Start the Validation Process*)

Responsibilities The responsibilities of the System Validation Steering Team include, but are not necessarily limited to:

- overseeing activities for a specific computer system validation
- ensuring the Validation Protocol is created (either by members of the System Validation Steering Team or the System Validation Team)
- approving the Validation Protocol
- establishing the System Validation Team
- holding regular project meetings at predefined intervals
- preparing the Validation Report
- certifying validation results
- establishing a periodic review, and if necessary, revalidation, for the system

Experts The System Validation Steering Team may rely on the opinion of experts within the System Validation Team when they certify the validation.

Consultants Consultants may be engaged if there are insufficient resources and expertise in the System Validation Team or the System Validation Steering Team or both.

Consultants would normally be requested by the System Validation Steering Team and approved by the Validation Committee (See, *Chapter 2 - Before You Start the Validation Process*).

A request to engage a consultant must include a detailed documented review of the prospective consultant's capabilities. Before engagement, a contract specifying scope, roles and

responsibilities must be signed.

System Validation Team

- Introduction** This topic looks at the System Validation Team. The System Validation Team *performs* the validation tasks for a specific system.
- Team members** The System Validation Team composition should reflect the requirements of the validation. This will vary according to the type of system in question.
- The System Validation Team should be made up of representatives from:
- Department in which the system is used
 - Information Resources
 - a User or Technical level (someone who is familiar with the system)
 - Problems and issues should be raised with the System Validation Steering Team
- Note** The System Validation Team will be guided by the System Validation Steering Team.
- Training** The System Validation Team should receive additional (documented) training as appropriate to the validation being performed. This will ensure that they have the required skills.
- Responsibilities** The responsibilities of the System Validation Team include, but are not necessarily limited to:
- identifying the required validation/qualification activities i.e.

prepare validation plan

- preparing the qualification protocols
- performing the validation/qualification activities
- collating, indexing and filing validation/qualification documentation
- preparing the Validation/Qualification reports

Chapter 5 Determine Validation Activities

Overview

Introduction This chapter looks at the second step in the validation process, determining the validation activities required for the specific system.

Note: When you have determined the validation activities, you should record them, along with approximate timings and responsibilities. This document is called the Validation Plan.

Validation activities The validation activities are the exact details or activities that will be required for each of the steps in the validation process.

Note: The type and number of validation activities will depend on the nature of the software and the actual computer system that is being validated.

Software categorization Software can be divided into different categories. The type of software category will determine the validation activities that are required for the software.

Development Life Cycle and Validation Activities

Introduction This topic provides a diagram of an example development life cycle and some associated validation activities.

The example life cycle is broken down into these phases:

- specification
- design
- construction
- testing
- installation
- acceptance testing
- operation

The validation activities are shown in relation to when they occur in the development life cycle.

Example: In the installation phase, the validation activity that occurs is the Installation Qualification.

Software Categorization and Validation Activities

Introduction To help determine validation activities for software, this guide groups software commonly found in systems into five categories and recommends the validation activities for each category.

Complex systems often have layers of software, and one system could exhibit several or even all of the categories.

Validation activities The table below outlines the validation activities that should be undertaken for the appropriate software category.

No.	Category Description	Validation Activities
-----	----------------------	-----------------------

1	Operating systems	Record the version.
2	Standard instruments, Micro controllers, Smart instrumentation	Record the configuration.
3	Standard software packages	Validate the application.
4	Configurable software packages	Audit the supplier. Validate the application and any bespoke code.
5	Custom built or bespoke systems	Audit the supplier. Validate the complete system.

Category 1 Category 1 is operating systems.

Established, commercially available operating systems which are used in pharmaceutical operations are considered validated as part of any project in which the application software operating on such platforms are part of the validation process (i.e. the operating systems themselves are not currently subjected to specific validation other than as part of particular applications which run on them). Well known operating systems should be used.

For validation record keeping, record the name and version number in the hardware acceptance tests or 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 could lead to a formal re-testing of the application, particularly where a major upgrade of the operating system has occurred.

Category 2 Category 2 is Standard Instruments, Micro Controllers and Smart Instrumentation.

These are driven by non user programmable firmware.

Examples: Weigh scales, bar code scanners, 3 term controllers.

This type of software is configurable and the configuration should be recorded in the equipment IQ.

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.

Category 3 Category 3 is Standard Software Packages. These are called Canned or COTS (Commercial Off-The-Shelf) configurable packages in the USA.

Examples: Lotus 1-2-3, Microsoft Excel and other spreadsheet packages.

There is no requirement to validate the software package, however new versions should be treated with caution.

Validation effort should concentrate on the application written with the package (reference should be made to Category 4), which includes:

- system requirements and functionality
- the high level language or macros used to build the application
- critical algorithms and parameters
- data integrity, accuracy and reliability
- operational procedures

As for other categories, change control should be applied stringently, since changing these applications is often very easy, and with limited security. User training should emphasize the importance of change control and the validated integrity of these systems.

Category 4 Category 4 is Configurable Software Packages. These are called custom configurable packages in the USA.

Examples: Distributed Control Systems (DCS), Supervisory Control and Data Acquisition packages (SCADA), manufacturing execution systems and some LIMS and MRP packages, database and document management applications.

(Note: In these examples the system and platform should be well known and mature before being considered in category 4, otherwise category 5 should apply.)

A typical feature of these systems is that they permit users to develop their own applications by configuring/amending predefined software modules and also developing new application software modules.

Each application (of the standard product) is therefore specific to the customer process and maintenance becomes a key issue, particularly when new versions of the standard product are produced.

This guide should be used to specify, design, test and maintain the application. Particular attention should be paid to any additional or amended code and to the configuration of the standard modules. A software review of the modified code (including any algorithms in the configuration) should be undertaken.

In addition, an audit of the supplier is required to determine the level of quality and structural testing built into the standard product. The audit needs to consider the development of the standard product which may have followed a prototyping methodology without a customer being involved.

Category 5 Category 5 is Custom built or bespoke systems.

Complex systems
Determining the category

<u>Development Activities</u>		<u>Validation Activities</u>
User Requirements Specification Functional Specification *	Specification	Validation Plan & Protocol
		Supplier Audit
		Specification Qualification
Hardware Design Specification Software Design Specification Software Module Design Specification Mechanical & Electrical Specifications Network Design Specification	Design	Design Qualification
Hardware Manufacture & Assembly Code Software Modules Equipment Manufacture & Assembly ** Network Manufacture & Assembly	Construction	Construction & Code Reviews
Hardware Testing Software Module Testing Software Integration Testing Equipment Testing **	Testing	Monitor Supplier***
Hardware Installation Software Installation Equipment Installation ** Network Installation Hardware Acceptance Testing Network Acceptance Testing	Installation	Installation Qualification
System Acceptance Testing	Acceptance Testing	Operational Qualification
		Performance Qualification
		Validation Report
Maintenance Change Control	Operation	Ongoing Operation (Controls & procedures)

Chapter 6

Write the Validation Protocol

Overview

Introduction This topic looks at the third step in the validation process, writing the Validation Protocol.

The Validation Protocol

Introduction This topic looks at the Validation Protocol.

The Validation Protocol describes the procedure to be followed to validate a specific computer system and how each step in the procedure will be documented to provide a high degree of assurance that the computer system will perform as intended.

The Validation Plan will be used as the basis for creating the Validation Protocol.

For the benefit of site management, the Validation Committee and supplier representatives, the Validation Protocol must also provide a high level description of the overall philosophy, intention and

approach.

Responsibilities The Validation Protocol should be written by the System Validation Team and approved by the System Validation Steering Team (or Validation Committee if a System Validation Steering Team does not exist).

Timing The Validation Protocol is written after the validation activities have been determined. The Validation Protocol should be written before the system validation activities are conducted and in accordance with the local SOPs.

Amendments Amendments to the Validation Protocol must be approved by the same process as the original Validation Protocol, and documented in a supplement or as a new version.

Existing system The Validation Protocol for an existing system is the same as for a new system. It may include extra considerations or requirements based on the fact that the system has been in use for some time.

Recommended sections The recommended sections for the Validation Protocol are as follows:

Introduction and overview

- Scope
- Validation procedures
- Responsibilities
- Acceptance criteria
- Certification and approval
- Protocol sign-off and acceptance

Note: You should use diagrams and flowcharts where appropriate.

Recommended Contents for the Validation Protocol

Introduction

This topic looks at the recommended sections and contents of the Validation Protocol.

Introduction and overview section

The introduction and overview section should be a general introduction to the system, its uses and the need for validation. It should describe the organization in which the system will be used and who will be responsible for the system. It should contain a general reference to the quality critical functions and business critical functions that the system performs.

Consider including the following information in the introduction and overview section:

- type of validation to be undertaken (prospective or retrospective)
- description of the process, plant or facility, to put the system in context
- system description
- purpose and objectives of system
- major benefits of system (or continued use of system)
- principle of operation
- quality critical functions
- business critical functions
- major components
- major interfaces
- system boundaries (for hardware, instrumentation and information flow)
- summary of supplier and system history (This may be documented in a separate section e.g. Supplier Selection)
- summary of similar systems
- years on market
- number of sites
- reference to any other documents which describe the system

- ownership/responsibilities for system
- development
- commissioning
- operations

Include diagrams and schematics where appropriate.

Scope section This section should describe the scope of the system to be validated. It should specify the following information:

- whether system is new or existing
- how the system should be or is to be used
- which parts of the system/application will and will not be validated
- justification for excluding parts of the system
- limitations and exclusions
- particular application functions
- unused package facilities, unused I/O ports
- components for future use which cannot be tested yet
- assumptions (e.g. correct operation, but not configuration, of operating systems or networking software)
- boundaries
- physical (e.g. area of site covered)
- logical (e.g. interfaces covered)
- items to be validated (e.g. control systems, interfaces, peripherals, instrumentation, services, other equipment)

If the system is part of a multi-component system, reference should be made to the Validation Protocols that will cover the remaining components.

Validation procedures section

This section should describe the activities that will be used to validate the system and the list of documents that will be generated for each activity. It should be detailed and cross-reference the standards for the documents to be written.

Responsibilities section

This section defines which parties (by name) will be responsible for each of the activities in the validation process and for generating the associated documentation.

Whilst the supplier or specific information departments may be responsible for performing parts of the validation, the end user department has the final responsibility for completing the validation.

Consider including an organizational structure chart for the validation project team.

Acceptance criteria section

This section describes the high-level acceptance criteria for each activity. It is the basis for the final certification of the system.

Specific acceptance criteria and expected results for each validation activity will be described in the qualification protocols.

Certification and approval section

This section describes how the validation will be certified or approved. This will be the basis for accepting the system for use in a validated state.

The personnel responsible for reviewing and accepting each document and for the validation as a whole should be identified.

Protocol sign-off and acceptance section

The final section of the protocol should contain the sign-off on the Validation Protocol by the validation steering team. This sign-off signifies the team's agreement to its contents.

Chapter 7

Specify the System Development Details

Overview

Introduction This topic looks at the fourth step in the validation process, specifying the system development details.

What You Should Specify to the Supplier

Introduction You should specify to the supplier (whether this supplier be an internal Company-IR group or a third party) that they must have:

- a good methodology in order to develop a system
- a formal quality management system for the development, supply and maintenance of the system

You should also make the supplier aware of the fact that the Company may audit.

These are the standards that should be in place in order to develop a system that can be validated.

Good methodology The supplier's system must be developed using a good methodology that uses a life cycle approach.

Note: A general description of a system life cycle methodology can be found in the IR Policy Manual and other references (See References in the *Reference Material* part of this document).

Formal quality management system The supplier's computer system must be developed using a formal quality management system.

Adherence to a quality management system should provide sufficient documentary evidence for subsequent acceptance by the

validation team.

Quality management procedures

The quality management system should include procedures associated with:

- documentation control
- project management
- quality planning
- life cycle definition
- testing
- configuration management
- programming/technical development standards

User Requirements Specification

Introduction

A good methodology and quality plan will ensure that a user requirements specification is developed.

This topic looks at the user requirements specification.

General details

The user requirements specification:

- describes the functions that a system or system component must or should be capable of performing
- is generally developed by the user in the initial stages of a system development or system selection process
- is written in general terms and specifies what needs to be done, not how it will be done
- is independent of the specific application program (technically non specific) that will be written or purchased

Techniques to

The following techniques may be used to capture relevant user requirements:

capture requirements

- workshops (such as critical requirements analysis workshops)
- interviews
- presentations
- data modeling
- data flow diagrams

Relationship with PQ and SQ

The user requirements specification will be used as the basis for the development of the system acceptance test scripts / performance qualification test scripts (See, the topic *Performance Qualification* in *Chapter 8 - Perform Qualification Activities*).

The user requirements specification will be reviewed during the specification qualification (See, the topic *Specification Qualification* in *Chapter 8 - Perform Qualification Activities*).

Functional Specification

Introduction

A good methodology and quality plan will ensure that a functional specification is developed.

This topic looks at the functional specification.

General details

The functional specification, or system specification:

- describes in a high-level manner, the hardware, software and peripherals that make up the computer system as a whole (Note: In system development terms, this specification will form the basis of system testing.)
- describes how the specific system to be purchased or developed will meet the user and functional requirements
- describes the specific user requirements that will not be met by the system
- should include reference to the data model to be used
- should define the functionality that does not relate directly to the user interface (e.g. system interfaces)
- should define the non-functional requirements such as performance and availability.

Recommendation It is recommended that a system description should be included as part of the functional specification.

Produced by The functional specification is produced by the system developer/supplier.

When a functional specification is produced The functional specification may be produced:

- when a new application is being developed
- when the users need to be exposed to the system before finalizing their requirements

Prototyping The functional specification may be produced from a prototyping exercise in order to model the required user interface.

The use of prototypes should be carefully controlled (e.g. by time boxing and controlling the number of iterations) and maintained within the scope of the User Requirements Specification.

The agreed prototype forms part of the functional specification and can be used as the basis for a first pass conference room pilot.

Note Functional specifications can comprise mechanical, electrical, and software function specifications for systems embedded in manufacturing equipment.

Relationship with OQ and DQ The functional specification will be used as the basis for the development of systems acceptance test scripts / operational qualification test scripts.

The functional specification is reviewed as part of Design Qualification (See the topic *Design Qualification* in Chapter 8 - *Perform Qualification Activities*).

Design Specification

Introduction A good methodology and quality plan will ensure that a design specification is developed.

This topic looks at the design specification.

General details The design specification is a complete definition of the equipment or system in sufficient detail to enable it to be built.

This specification will form the basis of module/integration testing.

The design specification is reviewed in the:

Relationship with DQ and IQ

- Design Qualification
- Installation Qualification - The design specification is used to check that the correct equipment or system is supplied to the required standards and that it is installed correctly.

(See, *Chapter 8 - Perform Qualification Activities*).

Documentation

Introduction A good methodology and quality plan will ensure that several types of documentation are developed.

This topic looks at the following types of documentation:

- end-user documentation

- administration documentation
- system support documentation

End-user documentation

End-user documentation comprehensively describes the functional operation of the system.

This documentation should include:

- some means of problem solving for the user such as an index, trouble-shooting guide and description of error messages
- comprehensive drawings of the system, if applicable

End-user documentation is generally produced by the supplier or developer and should be updated each time the system changes.

Administration documentation

Administrator documentation is written for the administrator (the user who will maintain and administer the system).

This documentation:

- describes how to perform the administrator functions, such as:
- system configuration
- adding users to the system
- setting up levels of security
- setting up and maintaining master control records
- may be a special section of the end-user documentation or it may be provided as a separate document

Administration documentation is provided by the supplier.

System support documentation

System support documentation describes the system administration activities that are specific to the software.

These administration activities include:

- configuration of the environment

- installation
- maintenance documentation
- the running of batch jobs

System support documentation is provided by the supplier or developer for the system administrator.

Testing

Introduction A good methodology and quality plan will ensure that several types of testing are undertaken throughout the development life cycle.

This topic looks at the following types of testing:

- module testing
- integration testing
- system acceptance testing
- stress testing

Module testing

Module testing - sometimes known as unit testing - is testing at the level of a single functional routine or software module.

At a simple level, and independent of the system as a whole, unit testing verifies that the routine provides correct output for a given set of inputs.

Module testing is carried out to verify that the system performs as defined in a Design Specification (See, *Chapter 8 - Perform Qualification Activities*).

Integration testing:

- verifies that the system functions correctly as a whole
- proves that all software modules correctly interface with each other to form the software system as defined in the design specification and functional specification

Integration testing

Integration testing is performed on the fully built system, as it is to be used by the end-users. Data from other external systems may, however, be provided by "dummy" interfaces.

Example: A manufacturing resource planning system might be tested with data provided from a flat file that simulates the interface to the inventory system, without requiring the inventory system to be involved in the testing.

Similarly a process control system can be tested by "dummying" inputs and outputs from field instructions in the plant.

System acceptance testing

System acceptance testing is the testing of the system's interfaces to other systems in the computing environment.

It should cover both the testing of user requirements and system functionality. This not only ascertains that the system accepts data correctly from other systems, but also that it accurately passes data to downstream systems and correctly processes data within the system itself.

System acceptance testing is usually done separately from the integration testing in order to minimize the downtime and expertise requirements for the other systems.

The testing may be performed:

- at the suppliers (and then repeated at the user site)
- solely at the user site

Stress testing

Stress testing involves cataloguing the fact that the system fails in expected ways that are not catastrophic, yet are easily recognized as errors.

There are two categories of stress testing:

- entering data that is outside the range of acceptable data from the system and ensuring that the data is flagged as an error

- testing the system with a high volume 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. Based on this type of testing, the system developer will rate the system's maximum operational capacity

These two types of stress testing should be performed by the developer of the system as part of module testing and integration testing rather than as a separate activity.

Similar testing of the system related to the user's planned operation and environment should be included as part of the Performance Qualification (See, the topic *Performance Qualification* in *Chapter 8 - Perform Qualification Activities*).

Relationship with OQ and PQ

For a standalone computer system, the system acceptance testing broadly equates to OQ and part of PQ. Some aspects of performance qualification may need to be performed by the user after system acceptance testing (especially for configurable software packages).

For an embedded system, system acceptance testing is only part of OQ/PQ since other machine performance checks of components which do not form a direct part of the system will need to be performed.

Traceability

It is very important that direct traceability is established between the specification and the test performed i.e. a cross reference from the test script back to the section in the appropriate specification where the function is defined.

This traceability ensures that all parts of the software are tested, and clearly establishes the acceptance criteria for a given test.

Computer System Retirement

Introduction

Planned and organized execution of computer system retirement is essential for business and quality-critical systems to ensure continuity of data.

Stages in retirement

The stages in the retirement process depend on the definition of raw

process data for each system.

For regulatory systems, data must be archived either electronically or as some form of hardcopy (paper or fiche). Continued on-line access to data may be achieved by data migration to the replacement system, although this should be treated as a development project in its own right (design and testing of migration/translation tools, quality control of a proportion of transferred data, etc.).

A pre-retirement review of validation is necessary to ensure that the documentation and testing package is complete BEFORE the system is dismantled. The physical decommissioning process should be planned, ensuring overlap with the replacement system already operating successfully in its production environment.

**Consideration
for computer
systems
retirement
during
systems
development**

A key feature is that data archiving and retirement should be planned for at the initial requirements and design stages. Specifying the need for compact raw data prints including the full audit trail would help archiving. Ensuring the system can export all data types in a standardized, non-proprietary file structure would facilitate data migration.

Chapter 8

Perform Qualification Activities

Overview

Introduction

This topic looks at performing qualification activities, the fifth step in the validation process.

Types of reviews

These are the types of qualifications that can be performed:

- Supplier audit (this would be applicable to both internal and external suppliers)
- Source code review
- Specification qualification

- Design qualification
- Installation qualification
- Operational qualification
- Performance qualification

Note

At the end of the development process for a bespoke application, the supplier will usually perform a System Acceptance Test at the supplier site.

A request should be made to obtain this documentation for inclusion as part of the validation documentation.

Supplier Audit

Introduction

The supplier audit is usually undertaken for configurable software packages or custom built/bespoke software. It should be performed either:

- prior to the formal commitment to purchase (for configurable software packages)
- during the development process (for custom built/bespoke software)

Pre-qualification of suppliers

For projects where either a number of suppliers could potentially offer a packaged solution or a supplier is being selected for a custom activity, then a number of suppliers may be subject to a ‘pre-qualification’.

The pre-qualification may take the form of a visit or a questionnaire, which is sent to the supplier for their completion. The questionnaire would contain questions relating to the supplier’s organization and quality management system.

Responsibility

The System Validation Team will either:

- undertake the supplier audit, or
- check the results from the supplier audit

Note: The System Validation Team will check the results from the supplier audit when they perform the Specification Qualification (SQ) (See, *Specification Qualification* later on in this chapter).

Purpose

The purpose of the audit is to assess the supplier or development group's quality management system, specifically the development methodology and quality plan, to ensure that quality assurance is built into the software development process.

The audit should verify that the development methodology conforms to the system development standards specified as part of the validation process.

Recommendation It is recommended that you use the following Good Automated Manufacturing Practice (GAMP) Guide as the standard against which suppliers are assessed:

GAMP - "Supplier Guide for Validation of Automated Systems in Pharmaceutical Manufacture" Produced by the GAMP Forum.

GAMP

Although the GAMP document has been written specifically with manufacturing systems in mind it is recommended as a general reference as the principles contained within it are general and can be applied to the validation of systems of all types.

Other references

You should also refer to:

- ISO 9001: 1994 Quality Systems. Model for quality assurance in design/development, production, installation and servicing
- ISO 9000-3: 1991 Guidelines for the application of ISO 9001 to the development, supply and maintenance of software
- ISO 10011 Quality Systems Auditing
Part 1: Auditing
Part 2: Qualification criteria for Quality Systems auditors
Part 3: Managing an audit programme
- ANSI/IEEE Standards (detailed reference may be found in the Reference Material Section)
- The TickIT Guide - A guide to Software Quality System Construction and Certification using ISO 9001.

Audit details

The supplier audit should address the following topics:

- the supplier's development methodology and quality plan
- the documentation trail of a key product or component through the development life cycle to verify that the methodology has been followed
- project status reports or other internally generated documentation
- evidence of good testing procedures

Testing

Testing should be performed concurrently during the development of the system by the developer of the software, not solely by the purchaser or user at the end of the development process.

If there is sufficient evidence of testing, the testing procedures review can be considered part of the Operational Qualification.

Source Code Review

Introduction This topic covers source code review.

When to review source code	Use the table below to determine if Company should perform a source code review:	
	If ...	Then a source code review will...
	a supplier audit is not possible	be required
	there is satisfactory evidence in the supplier audit that the source code was developed in a high-quality manner and subject to review as part of the	not be required

development life cycle	
a supplement to the supplier audit is desired	<p>be required</p> <p>Note: The source code review should be a high-level review. It should use diagrams and charts of the software.</p>

Note Source code should be characterized to identify:

- key information, including version number and the language used
- other detailed information, if it will add value to the validation exercise

Recommendation It is recommended that you use the following Good Automated Manufacturing Practice (GAMP) Guide as the standard against which source code standards are assessed:

GAMP - "Supplier Guide for Validation of Automated Systems in Pharmaceutical Manufacture" Produced by the GAMP Forum.

Specification Qualification

Introduction The Specification Qualification (SQ) is a technical, quality and commercial review of the tender/requirements package.

Note: Bespoke systems will require a full SQ whereas off-the-shelf systems will require a much simpler SQ.

Purpose The purpose of the SQ is to show that the controls required to specify the design have been addressed and agreed by the user, and where appropriate the in-house implementation group.

Documents to The following documents may be reviewed during the SQ:

be reviewed

- User Requirements Specification
- Functional Specification (if available at this time)
- Supplier Audit Report
- Supplier Contract
- Commercial and Purchasing Specifications

Acceptance criteria for the SQ will be defined in the Validation Protocol. The protocol will define which documents should be prepared, the standards they must meet and the approval status required.

Recommendation It is recommended that you use the following Good Automated Manufacturing Practice (GAMP) Guide as the standard against which the above documents are reviewed:

GAMP - "Supplier Guide for Validation of Automated Systems in Pharmaceutical Manufacture" Produced by the GAMP Forum.

Supplier audit If a supplier audit has not been done, then the validation team may be required to undertake one.

SQ Report The results of the SQ should be documented in the SQ Report.

The report should also include:

- details of tasks done during the SQ
- supporting documentation
- version numbers or dates of documents reviewed

Design Qualification

Introduction The Design Qualification (DQ) is a review of system design and

documentation for technical content and quality. It is usually performed prior to installation at the site.

Note: Bespoke systems will require a full DQ whereas off-the-shelf systems will require a much simpler DQ.

Purpose

The purpose of the DQ is to:

- confirm that the system development life cycle has been followed
- ensure that the individual elements of the system have been designed and proven
- ensure that the user and supplier agree that the integration of all the elements meet the User Requirements Specification, Functional Specification, Design Specification and Quality plan

Documents to be reviewed

The following documents may be reviewed during the Design Qualification:

- Functional Design Specification
- Flow Diagrams
- System Diagrams
- User Manuals
- System Manager Manuals (Administration Documentation)
- Design phase implementation and test documentation
- Drawings
- Material and equipment lists
- Quality plans
- A listing of any deviations from the User Requirements Specification for the system developed
- Compliance matrix (i.e. list of functions with reference to their quality critical nature)
- Change control records
- Source code development review

Note: Although not all the documentation may be available,

sufficient documentation *must* exist in order to verify the system at the DQ phase.

Some or all of the requirements of the DQ will be met if a supplier audit is performed.

Acceptance criteria for the DQ will be defined in the Validation Protocol. The protocol will define which documents should be prepared, the standards they must meet and the approval status required.

Recommendation It is recommended that you use the following Good Automated Manufacturing Practice (GAMP) Guide as the standard against which the above documents are reviewed:

GAMP - "Supplier Guide for Validation of Automated Systems in Pharmaceutical Manufacture" Produced by the GAMP Forum.

DQ Report The results of the DQ should be documented in the DQ Report. The report should also include:

- details of tasks done during the DQ
- supporting documentation
- version numbers or dates of documents reviewed

Installation Qualification

Introduction The Installation Qualification (IQ), is the process of demonstrating that the system hardware and software has been installed according to the manufacturer's specifications and that all deliverables are properly accounted for.

The IQ is achieved by writing an IQ protocol and documenting the installation to ensure it meets the acceptance criteria defined in the protocol.

Note: For the purpose of this guide, the IQ covers both hardware, system software and application software.

System software System software (the operating system) is validated by default (because the application will not function without the system software) when the application software is qualified and the system is validated.

It does *not* need to be validated independently of the application software.

However, you do need to qualify the installation of the system software, and the operation and maintenance of the software.

Qualifying new applications When a new application is installed on a platform that has been validated for a different application, then an IQ only needs to be performed on the new application.

The new IQ should refer to the original IQ for the hardware and system software.

IQ Protocol An IQ protocol should be written.

The IQ protocol describes how the hardware and system and application software should be installed and should contain:

- system description, functional and design specifications as appropriate

- manufacturer's documented recommendations for installation
- manufacturer's instructions for unloading and installation
- post-installation procedure, if any
- associated equipment (Note: associated equipment should be qualified)
- associated conditions, e.g. environmental and ergonomic conditions
- associated documentation and deliverables including but not limited to:
 - ◇ List of approved deliverables
 - ◇ Manuals, operations, administration, technical and maintenance
 - ◇ Patch notes and/or release notes
 - ◇ Structures list (database, fields, types, sizes, keys, links and referential relationships)
 - ◇ SOP List and SOPs (Application, Systems, Environment, Training, IT specific and Corporate Software/Hardware critical SOPs.)
 - ◇ Screen shots
 - ◇ File list (application, shared, 3rd party installation,
 - ◇ Module or component list
 - ◇ Media list and storage locations
 - ◇ Report list
 - ◇ Registry changes list
 - ◇ Functional Requirements
 - ◇ User Requirements
 - ◇ Project Plans
 - ◇ Flow diagrams and/or schemas
 - ◇ User and Functional Matrixes
 - ◇ Pre-requisition descriptions and Matrixes

- ◇ Training procedures and employee training records
- ◇ Errors, dialogs and operator interface messages
- ◇ Audits and audit findings
- acceptance criteria
- responsibilities and roles

Qualification scripts may be written to support the requirements of the protocol.

Acceptance criteria for the IQ will be documented in the IQ Protocol and the qualification scripts, and will be based on the functional and design specifications along with the installation instructions.

IQ results

The IQ results should include the following:

- documentation of the installation procedure followed by the installer
- notes made by the installation technician during installation
- records of data entered or received from the system during installation
- obtaining original installation and diagnostic media, documentation and deliverables
- variances from the IQ Protocol

IQ summary report

If the results are extensive, then a summary report indicating whether the installation was performed according to the protocol should be written.

The report should be written by a knowledgeable and responsible person who has reviewed all of the actual results.

The report should state:

- whether or not the installation procedure was followed (as shown by the documentation generated by the person performing the installation)
- whether the environmental and ergonomic recommendations made by the supplier were met

- any deviations from the manufacturer's procedures or recommendations (these should be explained and justified)

Existing systems

For existing systems, any available documentation from the installation period should be gathered for the IQ. If there is no documentation available, then the IQ should contain this information:

- hardware and software description and configuration
- operating parameters
- environmental controls
- confirmation that the above items meet the supplier's /manufacturer's recommendations

Operational Qualification

Introduction The Operational Qualification (OQ) is a test to ensure that each component of a computer system performs as intended within representative or anticipated operating ranges.

It is equivalent to the testing performed by the supplier during the development process (i.e. module and integration testing) and to a system acceptance test at the completion of the system development process.

Note: OQ should also be performed on non-software components. However, that is outside the scope of this guide.

Testing as part of the development process If the tests performed by the supplier during the development process were satisfactorily performed and documented by the supplier, then the OQ requirement may be satisfied and the OQ may be waived.

Information about the types of testing should be obtained during:

supplier audit

Systems Qualification (SQ)

Design Qualification (DQ)

The OQ may not be waived for the following systems:

operational control systems

automated equipment with embedded PLCs which are connected to manufacturing process control and monitoring instrumentation

OQ Protocol An OQ Protocol should be written.

An OQ Protocol describes the approach used to qualify the system. It should include:

- execution instructions
- qualification scripts
- qualification data and data set-up requirements
- justification for the choice of data
- expected results
- resolution procedure for unexpected results
- acceptance criteria for qualification - this will be based on the appropriate and corresponding design specifications

Qualification scripts

Qualification scripts, or test plans, describe the step-by-step procedure for qualifying the system. The procedure may be broken down into multiple discrete scripts for ease of use.

The scripts should verify that the system performs as intended against the system specification created during the development process. They should include information about test conditions such as:

- security
- screen flow
- data validation
- data updates

There should be a direct reference between the test script and the specification against which the testing is being performed.

Qualification data

The data used with the qualification scripts should be identified.

The datasets should include:

- data that is typical of the data used in a live situation
- unacceptable data
- data at the boundaries of the acceptable range

Qualification results When the OQ scripts are executed, then the results should be recorded, signed and dated by the executor. Screen prints or electronic logs should be retained to verify the results. Automated testing tools may be used where appropriate to record the results.

If expected results and actual results vary, then the discrepancies should be clearly identified and backed up with the records of action taken to resolve them.

Summary reports If the OQ generates extensive documentation, then a summary report should be written.

This report may be reviewed by the System Validation Steering Team instead of reviewing all the raw data. (Note: You should still retain the raw data.)

The report should be written by a knowledgeable and responsible person who reviews all the raw data.

The summary report should include this information:

- whether or not the qualification scripts were followed
- whether or not the expected results were attained
- description of any deviation from expected results
- any follow-up activities to correct any deviations
- statement of whether the operational qualification results meet the acceptance criteria
- justification for the acceptance of the validation of the system

Existing systems Any testing or qualification performed on an existing system during its development should be reviewed. This includes a review of the operating history, including problem logs of the system.

If there are a significant number of software-related problems, then a more extensive OQ may be required. If there are few problems, then an OQ may not be required.

Performance Qualification

Introduction The Performance Qualification (PQ) ensures that the total system performs as intended in the specified operating range. The total system includes all hardware and software components, associated equipment, people and procedures that make up the system. The execution process is conducted using company specific pre-defined dataset or actual live data.

PQ is not validation Performance Qualification is not the same as validation. In earlier literature and in the industry, Performance Qualification was often called validation or validation testing. The two processes should not be confused. Performance Qualification is one process in a series of processes, which make up validation.

During the development process a system acceptance test will have been performed - either at the supplier site or at the user site. This system acceptance test forms part of the PQ.

The PQ should always be performed at the user site (and may involve repetition of all or part of the system acceptance tests as required) and will include testing specific to the user environment and defined ways of working.

PQ Protocol A PQ Protocol should be written.

The PQ protocol describes how the PQ should be performed. It should contain:

- description of the use of the system in the context of the work environment
- references to SOPs or other user documentation, and the user requirements/functional specification as required
- qualification scripts
- qualification data
- justification for the choice of data

- expected results
- data set-up requirements
- testing procedure
- resolution procedure for unexpected results
- acceptance criteria for qualification

Qualification scripts The qualification scripts describe the procedures to verify the performance of the system against the User Requirements Specification. They should simulate the operation of the system in a live situation, using all system components and operating procedures.

Scripts may be broken down into multiple steps for ease of use.

There should be a direct reference between the test script and the specification against which the testing is being performed.

Setting operational capacity limits

The PQ should also include testing the system against (but not exceeding) its operational capacity.

Note: If the system is expanded to operate at a higher capacity this type of testing should be repeated.

The operational capacity should be set by the user but should not exceed the rated capacity provided by the supplier.

Qualification data

The data to be used with qualification scripts must be identified. The choice of data should be justified in terms of its suitability for demonstration purposes. It should simulate the data used in live situations.

Abnormal data or data which is outside the operating ranges should also be tested to ensure that it is handled correctly in the system.

If a new computer system is to be implemented as part of a computer-controlled process in a manufacturing environment, process validation requirements should be considered.

PQ results When the scripts are executed, then the executor should record, sign and date the results. Screen prints or electronic logs should be retained to verify the results. Where appropriate, automated testing tools may be used to record results.

If discrepancies between expected and actual results are identified, then they should be resolved. Action taken to resolve discrepancies should be documented.

Summary report If the PQ generates extensive documentation, then a summary report should be written.

This report may be reviewed by the System Validation Steering Team instead of reviewing all the raw data. (Note: You should still retain the raw data.)

The report should be written by a knowledgeable and responsible person who reviews all the raw data.

The summary report should include this information:

- whether or not the qualification scripts were followed
- whether or not the expected results were attained
- description of any deviation from expected results
- any follow-up activities to correct any deviations
- statement of whether the performance qualification results meet the acceptance criteria
- justification for the acceptance of the validation of the system

Existing systems A PQ Protocol should be developed for an existing system in the same way as for a new system. The PQ should cover all of the functions outlined in the user requirements specification.

Historical information may be used in lieu of performance qualification scripts, data and expected results. However, the actions taken, the data used and the results obtained when the historical data was generated must be clear.

In addition, effective change control must have been in place to ensure that the system has not changed since the historical data was

generated.

If there is not enough data for some or all of the functions, then the gaps must be qualified as for a new system. The protocol must specify which system functions are qualified on historical information and which ones will be qualified as new.

Chapter 9

Develop/Review Controls and Procedures

Overview

Introduction This chapter looks at developing and reviewing controls and procedures, the sixth step in the validation process.

If the computer system is a new one, then you will need to develop the controls and procedures or check the suitability of existing generic procedures applicable to the site or department.

If the computer system is an existing one, then you will need to review the controls and procedures and update them if required.

Controls and Procedures

Introduction Controls and procedures ensure that the system retains its validated status in daily use. They must be approved and implemented before the system can be certified as validated.

Note: The creation and revision of control procedures must be controlled and approved according to site/departmental SOPs.

Responsibility The responsibility for each control and procedure rests with the department responsible for that activity.

Scope Controls and procedures should (as a minimum) cover the following areas:

- problem reporting
- change control
- back-up
- recovery
- business continuity
- operating procedures
- security administration
- database administration
- purge and archive procedures
- output controls

These procedures are outlined in the rest of the topics within this chapter. The precise controls and procedures required would be determined when you determine the validation activities that will be undertaken (step 2 in the validation process).

Training For every procedure that has been identified in the above areas, there should be documented evidence that the people affected by the procedure have been trained on its use.

Training provides a degree of assurance that the procedures are being followed.

Training must be repeated any time that a significant change is made to a procedure.

For procedures that are not routinely used, for example recovery procedures, training should include testing of the procedures. This will ensure that the procedure is adequate and that it can be followed.

Records should be kept of all training/testing activity.

Problem Reporting and Change Control Procedures

Introduction This topic looks at the problem reporting and change control procedures that should be in place.

If these procedures are not in place, then they must be developed, approved and implemented.

Problem reporting A problem log should be kept and a procedure should be written for logging, tracking and resolving system problems.

The procedure should reference the change control procedure to be followed if the problem resolution involves a change to the system.

Change control All changes to the system must be controlled and documented by a formal change control procedure.

The procedure must include steps for:

- assessing the impact of the change
- testing
- controlling the implementation

Note: Testing must ensure that the change is put in place with no adverse effect to the functioning of the system.

The change control procedure should be in place prior to the start of the qualification activities to ensure that any changes required during the validation are captured and processed in the correct manner.

Change control procedure

The change control procedure should cover changes to the:

- hardware
- system software (operating system)
- application software

- data

This should include the process by which application owners are to be notified of changes.

Changes to hardware

Changes to hardware should be controlled so that appropriate testing and documentation upgrades are performed.

Users should be informed of the change and, where appropriate, be involved in testing and approving the change.

Any external consultants making changes to hardware should be aware of this procedure and comply with it.

Changes to system software

Although the system software (operating system) does not require formal validation, it does require a change procedure to be developed.

This is because changes to system software can have an impact on the validated system that runs on it.

Changes to the system software should be performed in a controlled manner according to a written procedure with appropriate testing and notification to end users.

Note:

System software does not require formal validation for two reasons:

- there is usually extensive industry experience with system software
- system software is validated indirectly through the validation of the application

Changes to application software

Changes to application software should be controlled so that they are only implemented after the appropriate re-validation has been performed.

This would include:

- testing
- re-qualification

- training
- documentation

Users should be involved in performing the re-validation and approving the change.

Revision control should be in place to ensure that the version of software can be uniquely identified in the present and in the past.

Changes to data

Changes to master data or control data that sets the parameters or configuration of the system should be restricted to authorized personnel and should be tracked. A written procedure should describe how this is to be done.

The degree of control that is placed on changes to master data depends on the effect that data has on the functioning of the system.

Changes to historical data in the database should be covered under Database Administration procedures (*See, Operating and Administration Procedures* in this chapter).

Back-up, Recovery and Business Continuity Procedures

Introduction

This topic looks at the back-up, recovery and business continuity procedures that should be in place.

If these procedures are not in place, then they must be developed, approved and implemented.

Back-up procedures

Procedures should be in place for performing regular back-ups of the system.

Back-up procedures should include the following information:

- back-up frequency (this should be appropriate to the criticality of the system)
- back-up medium

- specific back-up procedures
- how the back-ups will be identified and stored
- maximum amount of data that could be lost due to the back-up schedule
- specifications for testing the procedures

Recovery procedures

Procedures should be in place for describing how the system will be restored using the back-ups should it be necessary.

Recovery procedures should include the following information:

how the appropriate back-ups are to be identified

specific recovery procedures (including partially completed transactions)

specifications for testing the procedures

Business continuity

A business continuity plan should be available to describe how the business can continue to operate in the event of the system being unavailable.

The plan should include:

- contingency procedures
- disaster recovery procedures

Contingency procedures

Contingency procedures describe how the functions provided by the system can be performed manually in the event of the system being unavailable.

Contingency procedures should include:

- manual operating procedures
- the tracking and reconstruction of manual events to update the system when it is restored
- a reflection of how long the business could operate without the system before there is a danger of material financial loss to the business

Disaster recovery procedures

Disaster recovery procedures describe how the organization can obtain alternate computer resources in the event of the primary computer system being unavailable.

These procedures would be followed if the system were down for longer than manual procedures could feasibly be used.

The disaster recovery procedures should include:

- alternate hardware and software
- all peripheral equipment
- necessary communication lines

Operating and Administration Procedures

Introduction This topic looks at the operating procedures, security administration procedures and database administration procedures.

If these procedures are not in place, then they must be developed, approved and implemented.

End-user operating procedures

End-user operating procedures describe how the users are to use the system in their daily jobs.

These procedures differ from the end-user documentation supplied by the supplier or developer in that they are specific to the company, the department, and the task at hand.

The end-user documentation may be referenced to avoid duplicate description of standard system functions.

For a given system there may be multiple procedures for use within different departments or for different tasks.

Systems personnel operating procedures

Operating procedures for systems personnel describe any routine system maintenance activities.

Written procedures are required for activities such as:

start-ups

shut-downs

job scheduling

systems logs

problem logging

Written procedures should describe how security features are authorized, implemented and maintained. They should also designate responsibility for each of these activities.

Security administration procedures

Security administration covers:

- physical security of the system
- security features of the operating system
- application specific security features, such as application access, transaction authorization and audit trails

Database administration procedure

A written procedure should describe how the application database should be administered.

It should contain a secure method for making changes to the database, including the authorization and tracking of changes.

Purge and Archive Procedures

Introduction This topic looks at the purge and archive procedures.

If any of these procedures are not in place, then they must be developed, approved and implemented.

Procedures Purge and archive procedures describe how data will be copied, stored in archives and deleted from the system.

These procedures should include:

- a systematic method for determining which data to archive
- a method for recording what data was archived
- a description of where the archived data is to be stored
- a description of how the archived data will be found and retrieved when required
- the retention period for archived records
- the testing specifications for all purge and archive procedures
- the testing specifications required for testing data recall from archive after software or hardware changes have been made
- a specification of the appropriate archive media (this will depend on the retention period of the data)

Output Controls Procedures

Introduction This topic looks at output controls procedures.

If any of these procedures are not in place, then they must be developed, approved and implemented.

Procedures If a computer system produces outputs (such as reports) of a sensitive or proprietary nature, or that must be controlled from a regulatory standpoint, then there should be controls over the logical and physical outputs of the system.

A procedure should be written that:

- describes the controls that are in place
- specifies how access to sensitive output may be authorized

Chapter 10

Certify the System

Overview

Introduction This chapter looks at the second to last step in the validation process, certifying the system.

The objective of the certification is to verify that all validation deliverables have met the acceptance criteria that were described in

the Validation Protocol.

Validation Report

The System Validation Team should produce a Validation Report that describes:

- what was done
- the results obtained
- any special considerations regarding the use of the system that were identified during the validation process
- whether the procedure as described in the Validation Protocol was followed, and if not followed then what was the deviation and why
- whether or not the acceptance criteria were met
- the documentation that was generated
- the location of the documentation generated
- the retention period for the documentation

Independent experts may review the results of specific validation activities and the summary report may incorporate their findings.

Document Retention

Documentation associated with the validation must be retained at least for the lifetime of that system.

Special consideration should be given to regulatory requirements for document retention, especially for computer systems holding information, which may pertain to patient safety.

Acceptance Criteria

Detailed acceptance criteria will have been defined for the individual qualification activities, whilst high-level acceptance criteria will have been defined in the Validation Protocol for the validation as a whole.

It should be noted that in some instances not all of the acceptance criteria for a validation will be met. If this is the case, then the Validation Committee are responsible for determining whether or not the system can be certified for use.

The Validation Committee may certify the system for use provided:

- that there is an action plan for resolution of the issues

- appropriate manual procedures are in place to supplement the computer system if necessary

Detailed guidance cannot be provided on this topic, as it will always be a matter of professional judgement of the personnel involved.

Certification review

The authorized parties, as identified in the Validation Protocol, review the Validation Report to confirm that the procedures were followed and that all acceptance criteria were met.

Approval for use of a computer system once the validation is complete is the responsibility of site or departmental management.

If the Validation Report is accepted, then the reviewers certify the system as validated by signing either the Validation Report or a certification form.

If the implementation date for the system is different from the certification date, then this must be indicated.

Existing system

The process for certifying an existing system is the same as for a new system.

Chapter 11

Review Periodically

Overview

Introduction This chapter looks at the last step in the validation process,

reviewing periodically.

When a system is certified, the control procedures should ensure that it continues to operate as validated. However, small changes over time can cumulatively affect the performance of a system.

The system should therefore be reviewed periodically to provide additional assurance of validation. The interval between reviews will depend on the level of use of the system and its criticality - critical systems will probably be reviewed annually.

It is the responsibility of the user/system/application owner to ensure that the periodic review is performed on the system/application.

The management of the periodic review requirements for all systems will be achieved through the site/departmental Validation Master Plan.

Review contents

The review should identify:

- who will perform the review
- frequency of review
- criteria for determining if re-validation is required
- review procedures

Scope

The review should cover:

- hardware and software specification
- problem log (to identify recurring problems or unresolved major problems)
- change control log
- maintenance logs
- work practices (to ensure SOPs still reflect usage)
- system security (to ensure user access rights are appropriate)
- identification of areas particularly sensitive to change
- identification of environmental changes which may significantly

impact the system

- ability of applications to support existing systems to the year 2000

Change control

When you review the change control log, you should:

- determine the total number of changes
- select a percentage of those changes and audit the change control records and associated documentation for completeness

Review report

The review findings should be documented in the Review Report. The report should conclude that the system is still validated, or requires re-validation, either wholly or partly e.g. some components of the system require re-validation.

Note: If additional validation is required, a Validation Protocol should be written.

Example periodic

Checklists can be very useful for reviewing periodically.

**review
checklist**

The table below illustrates an example periodic review checklist.

System Name :		
Date of original validation certification or last annual review:		
List Periodic Review Team :	1. 2. 3. 4. 5.	
Date of Periodic Review		
<u>Task</u>	<u>Pass / Fail</u>	<u>Comments</u>
1. Does the hardware system description accurately reflect the current hardware configuration.		
2. Does the software system description accurately reflect the current software configuration.		
3. Review the system problem reporting log (hardware and software) for the last year and address the following issues: 3.1 Are there any recurring problems on the system. 3.2 Are there any major problems awaiting resolution.		

<u>Task</u>	<u>Pass / Fail</u>	<u>Comments</u>
<p>4. Review the system problem reporting log (hardware and software) for the last year and address the following issues:</p> <p>4.1 Number of problems reported.</p> <p>4.2 Number of problems unresolved.</p>		
<p>5. Review system security and address the following issues:</p> <p>5.1 Do all users defined on the system still require access.</p> <p>5.2 Cross reference the access requirements of a certain percentage of users with their training records to ensure that they match.</p>		
<p>6. Review change control log for the last year and address the following issues:</p> <p>6.1 Total number of changes.</p> <p>6.2 Number of major changes.</p> <p>6.3 % changes to review</p> <p>6.4 Documentation required to support changes present.</p>		
<p>7. Discuss with the system users whether any changes have occurred to their way of working in the last year. Ensure that all system SOPs reflect the current use and operation of the system.</p>		

<u>Task</u>	<u>Pass / Fail</u>	<u>Comments</u>
<p>8. Review the maintenance logs of any equipment associated with the computer system under review and address the following issues:</p> <p>8.1 Has the appropriate periodic maintenance occurred.</p> <p>8.2 Are the calibration records up to date.</p> <p>8.3 Is there a current maintenance agreement for this item of equipment.</p>		
Report Reference		