Computer validation Guide

Final draft

oooo

Version 2

December 2002

Revision History:	
Version 1	August 2002
Version 2	References to 21 CFR part 11 in Chapter 6. “Legal Reference” – December 2002

Contents

2 Acknowledgement

This document was prepared by a Task Force consisting of representatives from various companies that participate in the Active Pharmaceutical Ingredients Committee of CEFIC.
We thank all the members mentioned below for their efforts, cooperation, creativeness, constructive comments and endurance. Without these elements this document would not have reached its present status.
The members of this Task Force are:

Lisa Näsman * (Astra Zeneca)
Claude Becker (Seloc/PCAS)
Gert Beets * (OmniChem)
Jean-Pierre Bovee (Aventis)
Gerben Canninga * (Diosynth)
Nigel Cryer (MSD/Merck)
William Cuijpers * (Diosynth)
Kees Piket (Solvay Pharmaceuticals)
Willy Verhaegen (OmniChem)

Allert Wiersema (DSM Anti-Infectives)

* these members left prior to finishing the document or joined later

3 Introduction

In the last decade, computerised systems have become a vital part in the manufacture of Active Pharmaceutical Ingredients.
Typical applications are Process Control Systems (DCS, PLC, SCADA), Laboratory Information Management Systems (LIMS), Laboratory Instrument Control Systems and Business Systems (ERP, MRP II).

cGMP regulations imply that the functionality's of those computerised systems, which have influence on the quality of the API, should be validated.

Validation shall demonstrate that the parameters defined as critical for its operation and maintenance are properly (adequately) controlled/managed.

It is essential that the validation is practical and achievable, adds value to the project, and is concentrated on the critical elements of the system.

This Guideline outlines the scope and legal requirements for the validation of computerised systems, chapter 7 gives a comprehensive methodology suitable for most situations within API production control and data handling situations.

For some specific cases the coverage may be less extensive and/or subsections may be merged depending on the criticality and the importance of the systems to be validated. Where specialist validation cases are to be handled chapter 8 gives the official guidance, references and industry guidelines.

4 Glossary

Refer to chapter 8.8

5 Scope

This guide is intended for use by manufacturers of Active Pharmaceutical Ingredients (APIs) and intermediates that use computerised systems for various parts of the process leading to the manufacture of an API or intermediate. It provides interpretation of existing cGMP guidelines related to the validation of quality critical computerised. These interpretations aim to be practical on one hand and acceptable for both the industry and authorities on the other. The emphasis is on explaining "what to do" and to a lesser degree in "how to do".
Whenever practical and feasible, attention will be paid to linking validation of computerised systems with other types of validation, like process validation and equipment validation.

Within this guide attention will be paid to two essential parts of computerised systems:
1. Infrastructure
2. Applications

When applying the contents of this guide it should be realised that not all computerised systems will contain all of the elements (a through e) mentioned below.
The following aspects will be covered:

a. hardware
b. operating system
c. network system
d. data base management system
e. system software
f. strategy
g. compliance
h. project plan
i. system life cycle
j. change control

Apart from the above-mentioned subjects, supporting activities as training of personnel, documentation and use of checklists will be covered. Attention will be given to the aspect of risk-analysis in relation to validation of computerised systems.

Note

Although no guidance will be included in this document related to electronic records and signatures (refer to 21 CFR part 11), this subject area must be considered in the URS (also see chapter 6).

6 Legal requirements

Computerised systems used in the manufacture of API's should be properly developed, validated and maintained to assure data and product integrity.
The newly developed guidance for the manufacture of API's (ICH Q7a) covers these requirements. It should be noted that according to the current understanding, 21CFR part 11 is not legally binding for API manufacturers; however it is advisable to consider the principles and recommendations contained in this document prior to validating computerized systems as required by ICH Q7a.

7 Guidance

7.1 Strategy

In today's business environment computerised systems are used more and more. It is critical to design and validate them so that they are fit for purpose and meet user as well as compliance requirements
There is a need for clarification of this very complex and often misunderstood area of compliance. This area is increasingly the domain of a few consultants and experts.
This document will provide clear transparent guidance for API-manufacturers.
It will help industry to redress the balance between too much, often ineffective, documentation with too little impact on quality assurance. This will bring about a cost effective, added value efficient and effective way of performing validation of computer systems that are maintained in compliance.
A strategy to achieve this will be set out in a pragmatic approach using a Validation Plan including the elements below.

7.1.1 Approach

1. The approach to validation of computer systems should be based on common sense and use techniques that are familiar within other areas of validation and also business.
2. It is important to establish the final objective of validation and to choose an approach where a positive response is given, every time the following questions are asked:
   - Will this have added value?
   - Is this the most efficient way?
   - Is this the most effective way?
   - Can we achieve 80 % of the objective with 20 % of the effort?
3. One way to assist with these decisions is to use simple flowcharts.

7.1.2 Analysis

A priority for validation activities can be established by analyzing a system inventory for the criticality, validation status, software category and system type. This analysis aids validation planning and prioritisation.

7.1.3 Inventory

For an effective approach the first make an inventory of existing and any proposed systems. In compiling the inventory an assessment should be made on the criticality of each system using a methodical approach. This list should be kept fully updated and the priorities should be assigned once the current inventory is completed.
This inventory could include classifications based on potential impact on product quality (e.g. critical, major, minor, none and further subdivide these into direct and indirect impact).
The inventory list (which can take the form of a spreadsheet or database) would normally include headings like:

- system name and version number
- system type, e.g. legacy system or new system and modules
- system owner
- system use, e.g. materials management, process control, analytical control etc.
- criticality, e.g. product quality, compliance, business
- validation status
- implementation date (actual or planned)
- development category (e.g. off the shelf, user developed etc.)
- software category e.g. spreadsheets, PLC's, process controls
- GAMP category
- CFR 21 part 11 e.g. compliant electronic records and signatures
- Last validation performance check.
- Priority (an outcome of risk analysis).

7.1.4 Risk Analysis

(Ref. ISPE baseline guide for qualification and commissioning)
A risk analysis all factors including safety, environment, product quality and financial should be taken into consideration. However the most important one is to define the criticality of the

system. Looking at the impact of the system on product quality, the validation status and the potential impact on the business can do this.
Product quality can be impacted directly, indirectly or not at all. Examples are as follows:

- direct impact: process control of final purification step, assay of finished product
- indirect impact: distribution list of finished products, equipment maintenance program

Once validated, all computerised systems must be maintained according to the System Life Cycle approach.
The cGMP approach to validation can be used in a total quality approach to computer systems involved in safety, environment, finance, however there is no legal requirement to do that and these systems should not be subject to cGMP inspection.

7.1.5 Economics

As API production usually takes place in a highly competitive environment it is of utmost importance to perform validation in an efficient and cost effective way. To that end each company has to decide how to execute validation.
Two main are used:

- to use own, well educated and trained personnel
- to hire consultants to guide and organise the validation task

The latter option should be considered especially for smaller companies. However, one has to realise that some in-house expertise is needed to stay in control of computer system activities and of the costs. It is a fundamental requirement that the company itself remains responsible for the ultimate results.
To assist in control of costs it is useful to recognise that not all computerised systems are in need of the same level of validation. Less critical systems should have appropriate level of documentation.

7.2 Compliance

cGMP regulations imply that computerised systems that influence the quality of the API must be validated.

The depth and scope of validation depend on the criticality of the computerised functionality. This has to be established by means of a risk analysis at an early stage of the validation process.
Compliance critical key points to be considered include:

- Proven fit for purpose
- Access control /user management.
- Data integrity including: prevention of deletion, poor transcriptions and omission.
- Authorised / unauthorised changes to data and documents
- Critical Alarms handling (Process)
- Audit trails
- Disaster recovery / Back up and retrieval
- System maintenance and change control
- Training

Evidence of sufficient control of these issues should be demonstrated in the validation documentation.

This compliance must be integrated using the system life cycle approach (SLC), and clearly identified in the user requirements phase for any new computerised systems as detailed in chapter 0.
For existing systems, for which a life cycle model was not applied, a gap analysis must be undertaken against cGMP compliance issues. Identified issues must be tested and documented following a formal qualification plan/report.
For any identified non-conformances, the following alternatives should be considered:

- upgrading
- ensuring the requested control level through additional procedure (s) if the upgrading is not feasible
- replacing/upgrading the system where gaps are substantial and cannot be covered by the previous measures.

7.3 Project Plan

7.3.1 Description

The project plan is the backbone of any IT validation activity for any system. It describes the objectives, the organization, schedule, step-by-step activities and their chronology including milestones. Among these milestones are the deliverables.
It should address measures to control the project such as review and communication.
It is assumed that the major aspects covering GMP quality management system are in place.
A document describing the current computer validation situation should be available.
For the activities undertaken as part of the project plan see section 7.4.4.

7.3.2 Organisation

Special attention should be paid to the project organization.

7.3.3 Project Manager

The Project Manager is responsible for meeting objectives in terms of compliance with URS, while observing quality, time and costs requirements.

7.3.4 System Owner and Sponsor

The System Owner is the formal owner of the system and he is responsible for the validated status of the computerised system
The Sponsor provides the necessary investment and resources to support the project.

7.3.5 Users

Key users must be identified prior to writing URS. For instance when a project covers different specific areas, it is worthy to appoint a key user for each specific area.
They must approve the following documents:

- URS
- Functional/Design Specifications

They are involved in testing. It is key that the user has sufficient knowledge of the type of system so that they can become involved in designing the system. If there is a lack of knowledge it is critical that training be provided so that the user can provide an informed opinion.

7.3.6 Developer/ Supplier

The role of the Developer/ Supplier must be clear regarding the deliverables, document authorization, timing, change control. They must comply with all referenced quality standards.
The Developer/Supplier must provide the design specifications that must meet the URS.
Increasingly, suppliers are involved in executing part of the validation activities (early testing at supplier's site).
These aspects would be covered by the contract established between the customer and the supplier.

7.3.7 Site Computer Support

The site Computer Support defines or at least authorizes the hardware design, taking in account the compatibility of the existing systems, load, infrastructure.
Their role regarding installation and maintenance, including documentation must be defined.

7.3.8 Quality Unit

Quality Unit should be involved from the very beginning of the project. They must, review and approve all quality impacting documents.

7.3.9 Responsibility Matrix

Activities and responsibilities are assigned, for example by using a matrix, which lists all the deliverables versus contributors for each task. Responsibilities for writing, approving and authorizing should be assigned.

E.g.

Contributors Deliverables	System owner	Project Manager	QA	Supplier	Key User	IT Support
URS	A	R	A		W	W
IQ/OQ protocol		R	A	W	A	A
Test Scripts		R	A	W	A	A
Etc.						

Responsibilities:
- Writing: W
- Reviewing: R
- Approving: A

7.4 System Life Cycle

7.4.1 Introduction

This chapter details the actual validation activities to be performed to provide a computerised system validated to current standards.
Validation activities for computerised systems are divided into qualification of infrastructure (computers, system software and network) and validation of applications (including the

application software, interfaces to other applications, equipment and operational procedures) because of the differences in the approach required for each of the groups. The term computerised system is used in the text to designate the combination of both infrastructure and applications.

A Validation (Master) Plan should be developed according to company policies and internal procedures, including both infrastructure and applications.
Standard Operating Procedures (SOPs) should be in place together with a formal System Life Cycle Concept which describes all the relevant activities for creating and maintaining qualified infrastructure and application.

7.4.2 Software Categories (GAMP)

For applications the software development method or status can determine the validation effort. A very useful reference in this area is the GAMP guide.
The GAMP guide is an industrial standard (Ref 0) that defines five validation level categories for software as shown in the matrix below.
Categories 4 and 5 are the categories for which major validation efforts are required

Software categories according to GAMP Guide

GAMP category	Type	Application example	Infrastructure example	Remarks
1	Operating Systems, network software		VMS, MVS, UNIX, Windows NT	Established, commercially available operating systems which are used in pharmaceutical manufacture are considered validated as part of any project in which 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).
2	Standard Instruments, Micro Controllers, Smart Instruments	balances, pH meters, bar code scanners, PID controllers.	logic on an interface controller	These are driven by non-user programmable firmware. They are configurable and the configuration should be recorded in the equipment IQ.
3	Standard software packages	Office applications, spreadsheet, data base systems	Layered products as DBMS, ACL and communication packages	These are called Canned or COTS (Commercial Off-The-Shelf) configurable packages in the USA. Examples include Lotus 1-2-3, Microsoft Excel software (but not the spreadsheet itself since it includes generally calculations and eventually macros). There is no requirement to validate the software package, however new versions should be treated with caution.
4	Configurable software packages	LIMS, ERP (eg MRPII based), DCS, SCADA, MES, Chromatography data systems.	Users specific applications which are PLC based.	. These are called custom configurable packages in the USA. Examples include Distributed Control Systems (DCS), Supervisory Control and Data Acquisition packages (SCADA), manufacturing execution systems and some LIMS and MRP packages. 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 user process and maintenance becomes a key issue, particularly when new versions of the

GAMP category	Type	Application example	Infrastructure example	Remarks
				standard product are produced.
5	Custom built or bespoke systems	exclusively built solutions for a single or few customers	PLC with single purpose dedicated program	These are custom-build applications and include also the custom-build interfaces implemented when installing a configurable package. For these systems the full Life Cycle should be followed for all parts of the system. It should be noted that complex systems often have layers of software, and one system could exhibit several or even all of the above categories.

7.4.3 System Life Cycle Process

This section gives detailed guidance on the validation effort needed to establish documented evidence that a process will consistently perform according to its predetermined specifications and quality attributes. Depending on the complexity of the computerised system, or on the GAMP category, not all phases and/or activities have to be followed e.g. documentation can be combined (e.g. in the validation plan).
The activities and related output, which are described in the following sections, are not mandatory, but should be seen as an example to be adjusted for each specific situation.

The System Life Cycle concept describes all aspects of the life cycle of a computerised system that could consist of:

- planning;
- specification
- design
- construction
- testing
- implementation and acceptance
- ongoing operation;
- archiving of the system when replaced.

In the next chapters the validation activities are discussed step-by-step following the life-cycle concept.

7.4.4 Planning

Typical activities and output in this phase are:

Activity	Output
Define business need/problem	Business justification/Problem description *
Assign project manager	
Define project-/validation team	
Describe main system requirements	(Main) system requirements *
Perform feasibility study	Results feasibility study *
Allocate project resources	
Write project plan	Project Plan

* May be incorporated in the project plan

7.4.5 Specification

Validation of a computerised system should demonstrate that the system meets predetermined specifications. Testing is needed in several stages of the Life Cycle.
Therefore, **documented detailed specifications need to be available for each stage of testing**.
In the Specification Phase the detailed user requirements specification (URS) and the acceptance criteria are specified, based on identified critical requirements.
Based upon this specification a supplier market survey may be performed to screen the possible candidate suppliers. Usually, a supplier has detailed information about an existing product in documents like the functional specification and/or the system design.

After approval of the DQ formal change control should be applied to all specification documents.
Consider parallel validation activities (see table below)
Typical activities and output in this phase, not necessarily in this chronological order, are:

Activity	Output
Develop validation plan	Validation plan
Define User Requirement Specification (URS)	User Requirements Specification (URS)
Develop acceptance criteria	Acceptance criteria *
Perform risk analysis	Risk analysis report
Develop acceptance test plan (IQ/OQ protocols, PQ protocol if applicable)	Acceptance test plan (IQ/OQ/(PQ) protocols) *
Request for proposal (i.e. quotation)	Request for proposal
Supplier review/audit	Review/audit report
Supplier selection	Supplier qualification
Draw-up of contract	Contract with supplier; with contractual requirements
Place order	Acquisition order

*If the software is developed, these items are part of the System Design and Programming Phase.
Generally, before supplier selection, the User Requirements Specification (URS) and the acceptance criteria should be specified. The URS should contain three types of requirements:

- process/user related requirements (detailing the required functionality's),
- technical/IT related requirements (including not only e.g., hardware and software requirements and required interfaces with other computerised systems, but also addressing the capability of the system to migrate data from previous as well as to future versions or systems),
- quality/QA related requirements (including GMP-compliance and all requirements from 21 CFR Part 11).

All requirements should be unambiguous, complete, verifiable/testable and consistent with each other. Preferably the URS are set up in a way that the traceability matrix can be built up from there (see appendix 8.6).

7.4.6 Supplier / Vendor selection

Whether the supplier is an outside company or an internal department, the supplier's ability to provide a system that can be validated should be a primary consideration. Knowledge of validation requirements and experience in providing systems for GMP systems are important selection criteria.

At least for Category 4 and 5 systems used for GMP activities, a supplier quality review, and if considered relevant an on-site audit, should be performed to assess the validity of potential suppliers. The supplier review and/or audit should cover company information, Quality Management System information, Software Development and Package information. The supplier selection process should be documented and any deviations from the requirements observed during an audit should be addressed.
For critical systems, this review and/or audit should be carried out before final supplier selection.

When a supplier has been selected, contractual requirements should be defined. These contractual requirements are usually a “blend” of user requirements and technical specifications.

7.4.7 Design and construction

This phase is only applicable to computerised systems that belong to GAMP category 4 or 5 and will be mostly the responsibility of the supplier.
In the Design Phase a Functional Specification will be developed when customisations on an existing product are needed, or a custom built system. This is a combined activity by both the supplier(s) and the customer. The total design of the system can be checked against the User Requirements Specification to check that all requirements are met. This check is often facilitated by a Design Qualification, DQ.

Typical activities and output in this phase are:

Activity	Output
Define functional specification	Functional specification
Design the system	Design specification
Design Qualification	DQ report (can be included in the URS traceability matrix)
Programming and module testing	Software; module test report
Audit supplier	Audit report
Supply final system description	System description (including hard/software diagrams)
Supply system installation procedure	System installation procedure
Supply system documentation	Manuals and user guides

The supplier is required to follow a development methodology, programming standards, and Change Control procedures during product development. For purchased systems (GAMP 4 category), (parts of) the design and programming may already have been done. In this case the supplier has to supply documented evidence that a development methodology, programming standards, and Change Control procedures during the development phase were followed and that adequate tests were performed.

In case the supplier is doing substantial programming, in this phase possibly an additional supplier audit with a special focus on the SLC, adherence to procedures and proper documentation may be useful.

7.4.8 Acceptance Testing

The objective of this phase is to take a decision on the formal acceptance of the system as delivered by the supplier. For developed systems this can be divided in two parts:
- Acceptance at the supplier site, Factory Acceptance Test (FAT);
- Acceptance at the customer site, Site Acceptance Test (SAT).

It can also be combined in a general Acceptance Test.
During this phase the installation and operation of the computerised system have to be qualified according to the Installation and Operational Qualification (IQ/OQ) protocols for the system. Although a qualification can be, at least partly, performed by the supplier of the system, the project team is responsible for the results.

The level of detail of in-house testing is depending upon the GAMP category and upon the testing done by the supplier.
Typical activities and output in this phase are:

Activity	Output
Installation Qualification	IQ report (FAT, SAT can be included)
Operational Qualification	OQ report (FAT, SAT can be included)
Training of users	Qualified personnel
Audit/review of IQ/OQ and if applicable (parts of) PQ	Internal Audit/review report
Updated IQ/OQ report to QA for approval	Final approved IQ/OQ reports

If IQ/OQ protocols are used which the supplier develops, these protocols should be reviewed and approved by the project team before starting the IQ/OQ. The IQ/OQ can be executed at the user's site, by the supplier or by a user representative or member(s) of the project team. Each test shall be documented in a way that it can be reconstructed. This can be achieved by creating log files, making printouts, using logbooks etc. If these options are not feasible or practicable, witness testing is allowed. Testers should sign for each test performed. Reviewers should sign for logical sets of tests. In the case of witness testing the witness should sign for the same steps as the tester. Witness testing is required when a vendor or contractor undertakes system testing.

Chronology of IQ, OQ and PQ is a critical compliance issue. The critical parts of the IQ should be finished before executing OQ of that particular part, but parallel activities for other parts are possible.
If any test or challenge does not meet the specification or other deviations are found, this should be documented and, if necessary covered by corrective actions (e.g. identification of the cause of the deviation, corrective actions and additional tests).

After installation, the Operational Qualification (OQ) verifies the functional specifications of any individual system or sub-system. If needed to confirm whether the system meets the contractual requirements, relevant parts of the PQ need to be performed in this phase. (Additional PQ testing against other user requirements is often still required after acceptance). Results from previous testing e.g. FAT can also be used in this phase.

These results from an IQ/OQ must be reported as a formal report (combined it is often called an acceptance testing report). At this stage the system is approved for handing over to user for PQ to take place.

7.4.9 Implementation and acceptance

Often at the time of acceptance of the system from the supplier, additional testing and/or documentation is required before the system can be released for use in the production environment.

Typical activities and output in this phase are:

Activity	Output
Develop implementation plan	Implementation plan
Performance Qualification (PQ)	PQ report
Develop procedures	Procedures
Complete system description	System description
Training of additional users	Qualified personnel
Write validation report	Validation report
Validation review	Internal Audit report

The computer system may be formally released for PQ. PQ will take place in a production environment. During the period of Performance Qualification of the system additional monitoring is undertaken. Depending on the nature of the system the PQ can consist of a monitoring period or process validation (e.g. production of validation batches).
The implementation plan specifies the actions for implementing the computer system in its operational environment:

- Necessary activities and other documentation as required for the ongoing operation phase
- Training of additional system users
- Remaining test activities, like production of Process validation batches

By approving the final report, the system implementation is completed.

7.4.10 Ongoing operation;

Once a computer-related system has been validated, the validated state has to be maintained. This requires an adequate maintenance system and (a) Standard Operating Procedure(s) that are (is) incorporated in the relevant Quality Management System.

The following issues need to be covered as applicable in (a) procedure(s):

- Use of the system
- Security
- Back up and Restore
- Disaster recovery
- Contingency planning
- Business continuity
- Preventive maintenance
- Corrective maintenance (problem reporting)
- Change Control (including configuration management); also see chapter 8.5 Change Control
- Audit trail (equivalent to GMP alteration of data)
- Training
- Periodic evaluation
- Archiving
- System retirement (may be addressed in a much later stage)

7.4.11 Use of system

In this procedure the main tasks and responsibilities of the users should be defined. If needed, detailed instructions on how to use the system can be included.

7.4.12 Security

There needs to be proper access control procedures on three levels:

1. Wide and/or Local Area Network level
2. System, or Application level
3. PC level.

Items that need to be covered include how access is controlled, a password policy and audit trails. At all three levels there should be continuously updated lists of approved users and their authorisation levels.

7.4.13 Back up and Restore

The following items need to be covered by these documents:

- back-up and restore procedures
- frequency of back up
- verification of the ability to retrieve a back up data and files
- at least two generations or two copies of back-ups should be kept, unless other measures are taken to prevent back-up versions from damaged.
- back up copies should be stored separate from the system in a way that it is highly unlikely that both the original and the back-up copy/copies can be damaged.
- availability of the back up within an appropriate period
- for systems with a low back-up frequency, back ups should be checked for accessibility, durability and accuracy at a frequency appropriate for the storage medium, which should also be specified, but at least once a year for critical systems.
- in case of no low frequency back-ups change control should ensure the availability and integrity of back ups by restoring the data on a regular basis (particularly after changes to the system have been made).
- even with high frequency back-ups, prove the restore system works e.g. one time per year. There is no need to test the full system; this can be done by randomly selecting one or a few files to restore on a special area.

Note: if the same tapes are used, the tapes may be getting worse without noticing it.
The procedures have to be carried out, controlled and documented.

7.4.14 Disaster recovery

Disaster recovery procedures should be available for the most common disasters, often power failure or hard disk failure. The maximum downtime of the system should be documented, including the measures to meet that. If possible, disaster recovery procedures should be tested.

7.4.15 Contingency planning

In case of complete destruction of the hardware, software and data files, the knowledge and back ups of the system should be available to build up a complete new system. It should be documented whether and if so how the process is continued in case of a disaster (unavailability of the system).

7.4.16 Business continuity

Measures should be taken to ensure availability of source code in case the system supplier stops business for any reason. This should be addressed in Service Level Agreements and/or in Escrow agreements.

7.4.17 Preventive maintenance

Items to be covered are:

- List of critical system components
- There should be a system in place (e.g. Service Level Agreement SLA) which ensures that maintenance takes place in time. For hardware components the date of the last and/or next maintenance should be easily visible or retrievable.
- All maintenance should be documented (e.g., logbook)
- In case preventive maintenance leads to changes, the change control procedures should be followed.

7.4.18 Corrective maintenance (Problem reporting)

Each problem should be registered under a unique number or code, mentioning the problem, the date, the hard ware (registration number) the chosen solution, by whom it was handled etc.
If the solution of a problem leads to a change in hardware or software, the procedures for change control should be followed.

7.4.19 Change control

Refer to section 8.5

7.4.20 Audit trail

Computer generated and time stamped audit trails that independent record the date and time operator entries and actions that create modified or delete electronic records. Record changes shall not obscure previously recorded information.
The audit trail should be searchable and be secured from any changes.
It must be able to interrogate by dates, time, persons, type of change and reasons for change.

7.4.21 Training

All users of the system should be trained. This training should be documented and where applicable evaluated. Users should be informed about current standards or changes of the system. The training responsibilities should be defined.

7.4.22 Periodic evaluation

At predefined intervals (e.g. once a year) assessments should be made of the performance of the systems using the data from the change control and problem reporting documentation. A decision should be made and documented on the possible need for changes to and/or revalidation of the system. Decisions on periodic evaluation should be approved by at least the system owner and QA.

7.4.23 Archiving

All documentation generated in the Operation and Maintenance and Change Control procedures should be properly archived.
Data and the necessary software to retrieve those should be archived.

7.4.24 Retirement phase

At a certain point in the computer system's life cycle circumstances can occur which force a decision to retire the computer system. This decision will initiate the Retirement Phase (and probably an Planning Phase for system replacement).
In case of system retirement the following steps should be taken:

- Set up a data preservation plan which could include one of the following options:
- make sure that a new system will be able to retrieve data from previous systems
- preserve previous applications
- archive hard copies (when allowed)
- Completion of system documentation and validation dossier
- Execution of the data preservation plan
- QA audit on the preservation documentation

7.4.25 Infrastructure

Qualification of the infrastructure e.g. of the local area network, contains the following elements:

- high level documentation of the network e.g. security, reliability and availability.
- installation documentation including current schematic diagram.
- configuration management (an up to date inventory of hardware and software [incl. versions] components)
- monitoring of the performance of the infrastructure

Testing of infrastructure is normally included in the functional testing of the application (e.g. loop testing, process control systems etc).

7.4.26 Validation deliverables and activities

The system life cycle model as defined in this document serves as the backbone for the validation process.
Depending on the complexity and the GAMP category of the computerised system, activities and/or related output may be omitted, rationally combined or further subdivided.

Category 1 Operating Systems

Well-known operating systems should be used. 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 program of the application, particularly where a major upgrade of the operating system has occurred.

Category 2 Standard Instruments, Micro Controllers, Smart Instrumentation, Embedded software

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 Standard Software Packages

There is no requirement to validate the software package. Validation effort should concentrate on the application, 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 Configurable Software Packages

The life cycle should be used partially 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/review of the supplier is required to determine the level of quality and structural testing built into the standard product. The audit/review needs to consider the development of the standard product, which may have followed a prototyping methodology without a user being involved. The European Guide to Good Manufacturing Practices, Annex 11, requires that the development process is controlled and documented.
A Validation Plan should be prepared to document precisely what activities are necessary to validate an application, based on the results of the audit and on the complexity of the application.

Category 5 Custom Built or Bespoke Systems

For these systems the full Life cycle should be followed for all parts of the system. An audit of the supplier is required to examine their existing quality management systems and a Validation Plan should then be prepared to document precisely what activities are necessary, based on the results of the audit and on the complexity of the proposed bespoke system.

Guidance on validation documentation required for each GAMP category of software is given in the table below. GAMP 1 operating systems are not included in this table, as only version control should be applied. Wherever possible and practicable documents or items listed below may be combined. If documents are combined, it should be clear which of the indicated items are actually covered in which document(s).
The activities listed below are not necessarily in a chronological order.

Activities / output	GAMP 2, 3	GAMP 4	GAMP 5
Business need		+	+
(Main) system requirements		+	+
Results feasibility study		+	+
Project Plan		+	+
Validation plan	or IQ/OQ	+	+

Activities / output	GAMP 2, 3	GAMP 4	GAMP 5
	protocol		
User Requirements Specification (URS)	+	+	+
Acceptance criteria	+	+	+
Risk analysis report	If applicable	+	+
Acceptance test plan (IQ/OQ/(PQ) protocols) *)	Or Validation Plan	+	+
Request for proposal		+	+
Supplier review/audit report		+	+
Contract with supplier; with contractual requirements		+	+
Acquisition order		+	+
Functional specification		+	+
System design specification		+	+
DQ report (can be included in the URS traceability matrix)		+	+
Software; module test report		Optional	+
Supplier audit report on system development		Optional	+
System description (including hard/software diagrams)		+	+
System installation procedure		+	+
Manuals and user guides	If applicable	+	+
IQ report (FAT, SAT can be included)	+	+	+
OQ report (FAT, SAT can be included)	+	+	+
Internal Audit/review report	If applicable	+	+
Final approved (IQ/OQ reports[1]	+	+	+
Implementation plan		+	+
PQ report		+	+
Procedures	+	+	+
Training record	If applicable	+	+
Validation report	+	+	+

8 Appendices

8.1 Practical checklists for computer validation

This checklist is a quick reference; for more information reference is made to the relevant chapters

1. [1] The results from an IQ/OQ can be reported as a formal report (combined it is often called an acceptance testing report). At this stage the system is approved for handing over to user for PQ to take place. PQ will take place in a production environment. During the period of Performance Qualification of the system additional monitoring is undertaken. Depending on the nature of the system the PQ can consist of a monitoring period or process validation (e.g. production of validation batches).

VALIDATION PLAN PARAGRAPHS	INFRASTRUCTURE	APPLICATION
1) Introduction	– short description of the system – the position of the application system related to other systems – schematic overview of the system	
2) Scope	– describe the purpose of the validation process – describe ownership, customer and supplier role – subject of qualification, appointment to GAMP category	
3) Changes to the documentation	mention the changes from the previous version of this document.	
4) Validation team, members and responsibilities		
Team members/participants	X	X
Authorisation of protocols and documents	X	X
Who does the testing / who does verification	X	X
Responsibilities for the member	X	X
Authorisation of documents	X	X
Resources	X	X
5) Validation Strategy / Activities		
Verification of URS	X	X
GAMP categories		X
Risk analysis	x	X
References to URS – DQ	X	X
Testing criteria / acceptance criteria	X	X
References to applicable SOP's – change control	X	X
IQ, OQ and PQ protocols	X	X
Testing activities	X	X
IQ, OQ activities	X	X
Documentation of findings, reporting	X	X
Creation of infrastructure and application manuals	X	X
PQ activities		X
Review validation file	X	X
Validation report	X	X
6) Planning of activities	X	X

Design Qualification (DQ)	INFRASTRUCTURE	APPLICATION
System requirements in the URS to be covered by the succeeding specification requirements	Hardware, network, I/O cards, work stations, servers, alarms	Application software
Detailed URS including users specific configurations	X	X

Design Qualification (DQ)	**INFRASTRUCTURE**	**APPLICATION**
Compliance with relevant standards (GMP's, metrology…)	X	X
Ensure equivalency with old system (minimum)/ what if analysis	X	X
Organisational coherence	X	X
Internal and external links with other systems	X	X
Global software compatibility		X
System flexibility (without reconfigurations)	X	X
Alarms	X	X
Back-up and restoring	X	X
General design, system overview		X
GAMP categorisation		
Installation Qualification (IQ)	Documented conformance to the URS, to the DQ and to the supplier's specifications	
Component, system instrument list	X	
Software verification	X	X
Completeness and conformance to the DQ, the specifications and the order	X	X
	X	X
Observance of the relevant standards	X	X
Manuals (M&I etc), documentation and versions check	X	X
Supplier's tests	X	
Environment requirements	X	X
Installation and installation testing		
Operational Qualification (OQ)	Testing shall be carried out in real operation environment conditions	
Training	X	X
User manual(s) and SOP's	X	X
Calibrations (if applicable)	X	
Test all the critical functionalities to specifications	X	X
Data integrity testing	X	X
Alarms testing	X	X
Back up and restore testing	X	x
Access control	X	X
System change control in place	X	X
Data collection and review		X
Robustness, limit testing	X	X
Global review, reporting and conclusions		X
Performance Qualification (PQ)	Upon positive issue of previous phase, in actual operating conditions	
Actual operation conditions and limits		X
Data collection		X
Reliability checks against DQ	X	X
Reliability checks against manual or previous system		X
Actual impact / effect on product quality	X	X
Correction of defects (within the change control frame)	X	X
Suitability of SOP's	X	X

Design Qualification (DQ)	INFRASTRUCTURE	APPLICATION

Appendix 2 Traceability Matrix

8.2 User Requirement Specification Traceability Matrix

Purpose: This matrix provides assurances that the user requirements were fully covered in the validation documentation. The matrix also allows quick verification that the all the relevant functionality was tested during validation.

8.3 Definition

The first two columns identify the section number and title in the User Requirements Specifications. The remaining columns identify the document and the specific section where each user requirement was verified.

URS Spec Reference	User Requirements Specification Description < Document reference >	Validation and Project Plan < Document reference >	Functional Specifications < Document reference >	Installation Qualification < Document reference >	Operational Qualification < Document reference >	User Acceptance Testing < Document reference >	Performance Qualification < Document reference >

Document approval
System owner (name, date, signature)
Quality Assurance (name, date, signature)

8.4 Change Control

8.4.1 Scope

During the process of development, construction, validation, use and termination of computerised systems an SOP/system for controlling changes should be in place.
The purpose is to ensure that any change to the computerised system application is controlled in such a way that it will remain compliant with the GMP regulations. This can include hardware, software, authorisations, training and documents.
It must be considered very important that a global view is taken when dealing with change control.

It is advised to identify the category of change.
E.g. categories 1 to 3 below pertain to the application and would be classified as **routine change** and can be controlled by a simple procedure.
Categories 4 and 5 below deal with new, modified or additional processes which may have impact on product quality. For these cases a more careful control of the change is required and the need for revalidation should be reviewed.

Examples of different levels of changes:

1. giving new rights to a user,
2. entering a new kind of raw material in an ERP
3. replacing a hard disk
4. installing a bar code based identification system in a warehouse
5. connecting a LIMS to an ERP

8.4.2 Change control system

There should be procedures in place ensuring the following:

- Request for change (reason and description, identification)
- Change evaluation and authorisation (impact analysis; authorisation by system owner or delegate and QA-function, if applicable)
- Implementation and testing (testing/validation efforts should be based upon the impact analysis)
- Change completion, evaluation and approval (update documentation and formal release)

If based upon the impact analysis the change is minor and no testing or documentation updates are required, this should be documented.

The system should be kept as simple as possible: a procedure using change request forms can be adequate.

8.5 Matrix

Refer to section 7.3.9

8.6 Benefits

The benefits of computer validation are undoubted, however the bureaucracy of documentation often overshadows them.

Business benefits:
Strong project management will minimise re-engineering/designing and reduce redesign costs. The project discipline which computer validation brings can help to ensure that the team is in control of automated system.
By highlighting the critical phases it can be ensured that the project team spends the most effort and time on the most critical systems. Well defined User Requirements documentation provide clear indications of future requirements and make vendors aware of them.
Computer validation also brings discipline to an idealistic environment

Compliance benefits:
By following the system life cycle approach the following benefits will be achieved.
You will:

- meet regulatory expectations for computerised systems and ensure that the system is fit for purpose.
- ensure the system has data that is secure and control with protection against fraud, mistakes, system errors.
- use audit trails and passwords to assist control of the system.
- ensure that changes do not cause system failure by controlling and testing changes.
- meet new regulations by including them in user requirements, e.g. Electronic records and signatures
- ensure that suppliers understand the compliance requirements prior to design
- by version controlling help with debugging and disaster recovery.
- by limit testing ensure that the system performs as required and will not fail during times of stress or when unusual entry combinations are undertaken.

8.7 References

8.7.1 FDA (Web site http://www.fda.gov)

- http://www.fda.gov/ora/compliance_ref/cpg/cpgdrg/cpg425-100.html
- http://www.fda.gov/ora/compliance_ref/cpg/cpgdrg/cpg425-200.html
- http://www.fda.gov/ora/compliance_ref/cpg/cpgdrg/cpg425-300.html
- http://www.fda.gov/ora/compliance_ref/cpg/cpgdrg/cpg425-400.html
- http://www.fda.gov/ora/compliance_ref/cpg/cpgdrg/cpg425-500.html
- http://www.fda.gov/ora/inspect_ref/igs/csd.html
- http://www.fda.gov/ora/inspect_ref/igs/gloss.html

  - 21 CFR Part 211 G8 (a) and (b) - equipment
  - 21 CFR Part 211 180 (a) (c) (d) (e) – Documentation, Records
  - 21 CFR Part 11 – Electronic Records & Electronic Signatures.
  - FD & C Act section 704 (a) – Software inspection.

8.1 FDA Guidelines

CPGs	FDA Compliance Policy Guides on Computerised Drug Processing
CPG 7132a.07	Input / Output checking
CPG 7132a.11	CPG cGMP Applicability to hardware and Software
7132a.12 CPG	Responsibility of suppliers

General principles of software validation. Draft Guidance; Version 1.1. (June 1997)

8.7.2 ICH Q7a Good Manufacturing Practice for Active Pharmaceutical Ingredients (current guideline); chapter 12

8.7.3 G.A.M.P. (Guide for Validation of Automated Systems); current version

8.7.4 IEEE (Institute of Electrical and Electronics Engineers) 730, 828, 829, 830, 1012; (including guidance on software quality and Development).

8.7. 5 ISO Standards: ISO/CEI/2207, CEI 9126 – 94, ISO 12119

8.8 Glossary

Action Levels: Levels or ranges distinct from product specifications which, when deviated from, signal a drift from normal operating conditions and which require actions.

Alert or Warning Levels: Levels or ranges which, when deviated from, signal a potential drift from normal operating conditions but which do not necessarily require action.

Application: See application software

Application Software: (PMA CSVC) a program adapted or tailored to the specific user requirements for the purpose of data collection, data manipulation, data archiving or process control.

Archiving: The provision to ensure the long-term retention requirements for the type of data held and the expected life of the computerised system. System changes must provide for continued access to and retention of the raw data without integrity risks.

Audit: (ANSI N45.2.10-1973) an activity to determine through investigation the adequacy of, and adherence to, established procedures, instructions, specifications, codes, and standards or other applicable contractual and licensing requirements, and the effectiveness of implementation.

Auditee: The organisation to be audited.

Auditor: A person qualified to perform quality audits.

Audit Trail: For the purpose of computerised systems, audit trail means a secure, computer generated, time-stamped electronic record that allows reconstruction of the course of events relating to the creation, modification, and deletion of an electronic record. The data must be able to be interrogate, sorted by date, time, reason for change, person.

Automated System: Term used to cover a broad range of systems, including automated manufacturing equipment, control systems, automated laboratory systems, manufacturing execution systems and computers running laboratory or manufacturing database systems. The automated system consists of the hardware, software and network components, together with the controlled functions and associated documentation. Automated systems are sometimes referred to as computerised systems; in this Guide the two terms are synonymous.

Back-up: Provisions made for the recovery of data files or software, for restart of processing, or for use of alternative computer equipment after a system failure or a disaster (see restore).

Bespoke: A system produced for a customer, specifically to order, to meet a defined set of user requirements.

Bug: (ANSI/IEEE) A manifestation of an error in software (a fault).

Calibration: (PMA CSVC) Demonstration that a particular measuring device produces results within specified limits by comparison with those produced by a reference standard device over an appropriate range of measurements. This process results in corrections that may be applied to optimise accuracy.

Certification: (b. ANSI/ASQC A3 1978)

- Documented testimony by qualified authorities that a system qualification, calibration, validation or revalidation has been performed appropriately and that the results are acceptable.

- The procedure and action by a duly authorised body of determining, verifying, and attesting in writing to the qualifications of personnel, processes, procedures, or items in accordance with applicable requirements.

CGMP: (Code of Federal Regulations) Abbreviation for current Good Manufacturing Practice.

Change Control: (PMA CSVC) A formal system by which qualified representatives of appropriate disciplines review proposed or actual changes that might affect a validated status. The intent is to determine the need for action that would ensure and document that the system is maintained in a validated state.

Change Note: A document specifying the details of an authorised change request.

Change Plan: A plan defining the details of the authorised change request, defining actions, responsibilities and procedures.

Compiler: (ANSI/IEEE): A program used to translate a higher order language into its re-locatable or absolute machine code equivalent.

Computer devices: The combination of computers (servers, clients) and equipment providing computerized facilities

Computerised System: (PMA CSVC) A process or operation integrated with a computer system.

Computer Hardware: (PMA CSVC) Any physical element used in a computer system.

Computer System: (PMA CSVC) A group of hardware components and associated software designed and assembled to perform a specific function or group of functions.

Configuration: The documented physical and functional characteristics of a particular item or system. A change converts one configuration into a new one.

Configuration Management: (ANSI/IEEE) The process of identifying and defining the configuration items in a system, controlling the release and change of these items throughout the system life cycle, recording and reporting the status of configuration items and change requests, and verifying the completeness and correctness of configuration items.

Contingency plan: A document that describes how work is continued after total failure of the system for a period of time
NOTE: *In combination with this plan, there should be a Disaster recovery plan. (See: "Disaster recovery plan")*

Control software: application software such as utilities and tools which is used to control and manage computerized systems

Construction Qualification: Documented evidence to show that those constructional aspects of a facility that can affect product quality have been constructed in accordance with the approved specification.

Control Parameters: Those operating variables that can be assigned values that are used as control levels.

Control Parameter Range: Range of values for a given control parameter that lies between its two outer limits or control levels.

Critical Process Parameter: A process-related variable which, when out-of-control, can potentially cause an adverse effect on fitness-for-use of an end product.

Customer: The pharmaceutical customer or user organisation contracting a supplier to provide a product. In the context of this document it is synonymous with User.

Database: (ANSI) Collection of data fundamental to a system.

DCS: Distributed Control System.

Dead Source Code: (K.G. Chapman):

1. Superseded code from earlier versions. Avoided by using quality software development standards;

2. Residue from system modification. Avoided by effective configuration change controls;

3. Rarely used code that appears dead such as:

   - modules in some large configurable programs;
   - certain diagnostic programs that are intended to be inactive until needed.

Removal of code in category 3 leads to serious potential future problems. "Idle code" can be "parked" in libraries until needed.

Debugging: (IEEE) The process of locating, analysing, and correcting suspected faults.

Design Specification: (a. GAMP Forum, b.IEEE)

a. This is a complete definition of the equipment or system in sufficient detail to enable it to be built. This links to Installation Qualification which checks that the correct equipment or system is supplied, to the required standards and that it is installed correctly.

b. The specification that documents the design of a system or system component to satisfy specified requirements.

Design Qualification (DQ): Formal and systematic verification that the requirements defined during specification are completely covered by the succeeding specification or implementation.

Disaster recovery plan: A document that lists all activities required to restore a system to the conditions that prevailed before the disaster (e.g., power failure) occurred.
NOTE: *In combination with this plan, there should be a Contingency plan. (See: "Contingency plan")*

Documentation: (ANSI N45.2.10-1973) Any written or pictorial information describing, defining, specifying, reporting or certifying activities, requirements, procedures, or results.

Electronic Signature: A computer data compilation of any symbol or series of symbols executed, adopted, or authorised by an individual to be the legally binding equivalent of the individual's hand-written signature.

Edge-of-Failure: A control parameter value that, if exceeded, means adverse effect on state of control and/or fitness for use of the product.

Electronic Approval: An input command requiring restricted entry made under a level of higher authorisation, which signifies an act of approval.

Electronic Identification: (eID) An electronic measure that can be substituted for a hand-written signature or initials for the purpose of signifying approval, authorisation or verification of specific data entries.

Electronic Verification: An input command that enables a designated user, or the computerised system itself, to electronically signify verification or endorsement of a specific step, transaction or data entry. Source of the electronic verification may be made visible or invisible to users of the data.

Embedded System: A system, usually microprocessor or PLC based, whose sole purpose is to control a particular piece of automated equipment. This is contrasted with a standalone computer system.

ERP: Enterprise Resource Planning

Executive Program: (ANSI/IEEE/ASO) A computer program, usually part of the operating system, that controls the execution of other computer programs and regulates the flow of work in a data processing system.

External Quality Audit: A systematic and independent examination to determine whether quality activities and related results comply to a documented Quality Management System and whether this documented Quality Management System is implemented effectively and is suitable to achieve the contractual requirements placed by the customer.

FAT: Factory Acceptance Test (acceptance testing at the supplier's site).

Flow Chart: (ANSI/IEEE) A graphical representation of the definition, analysis or solution of a problem in which symbols are used to represent operations, data, flow, and equipment.

Frozen Software: This is software which is under configuration control and may not be altered without change control.

Functionality: See functional requirements.

Functional Requirement: (ANSI/IEEE) A requirement that specifies a function that a system or system component must be capable of performing.

Functional Testing: Also known as "BLACK BOX" testing, since source code is not needed. Involves inputting normal and abnormal test cases; then, evaluating outputs against those expected. Can apply to computer software or to a total system.

GAMP: Good Automated Manufacturing Practices

Hardware Acceptance Test Specification: (see Installation Qualification below). Documented verification that all key aspects of hardware installation adhere to appropriate codes and approved design intentions and that the recommendations of the manufacturer have been suitably considered.

Hardware Design Specification: (APV) Description of the hardware on which the software resides and how it is to be connected to any system or equipment.

High Level Review of Software:

> Purposes: determine if programs meet design specs as described by such documents as modular flow diagrams, HIPO charts, pseudo code and operating manuals.
>
> Characteristics: involves comparing design specs and acceptance criteria with cognitive mechanisms which depict the program in terms more easily understood by automation practitioners and non-computer scientists.
>
> Uses: quality acceptance review by QA software auditing, for example to complement "walk-through", for inspections, and for troubleshooting problems.

Infrastructure: Control software, system software, computers and network.

Installation Qualification [IQ]: (PMA CSVC) Documented verification that all key aspects of [software and] hardware installation adhere to appropriate codes and approved design intentions and that the recommendations of the manufacturer have been suitably considered.

Integration Testing: (IEEE) An orderly progression of testing in which software elements, hardware elements, or both are combined and tested until the entire system has been integrated.

Interface: (ANSI/IEEE) A shared boundary. To interact or communicate with another system component.

Legacy system: Software applications and computerised systems that have been working for many years and have never been validated. For systems that are critical to a regulated process a retrospective evaluation should be performed.

Life Cycle Concept: (PMA CSVC) An approach to computer system development that begins with identification of the user's requirements, continues through design, integration, qualification, user validation, control and maintenance, and ends only when commercial use of the system is discontinued.

LIMS: Laboratory Information Management System.

Loop Testing: Checking the installed combination of elements characterising each type of input/output loop.

Low Level Review of Software:

Purposes:

- detect possible coding errors
- determine adherence to design specs
- determine adherence to standards
- implement path analyses

Characteristics:

- requires highly trained experts who are familiar with software/hardware systems on which program is based

- to conduct low-level line-by-line source code inspection requires a team of experts working no more that two 2 hour sessions/day; this means about 100-150 lines of code per man-day (1.5 million lines = 40 man years)

Use: mainly during software development

Machine Code: (ANSI/IEEE) A representation of instructions and data that is directly executable by a computer (machine language).

Major Change: (PMA CSVC) A change to a validated system that, in the opinion of change-control reviewers, necessitates a revalidation of the system.

Minor Change: (PMA CSVC) A change to a validated system that, in the opinion of change-control reviewers, does not necessitate a revalidation of the system.

Modularity (Software): (ANSI/IEEE) The extent to which software is composed of discrete components such that a change to one component has minimal impact on other components.

MRP: Material Requirements Planning.

MRP II: Manufacturing Resource Planning.

Network: (a. ANSI/IEEE, b. GAMP Forum)

a. An interconnected or interrelated group of nodes.

b. An interconnected communications facility. A Local Area network (LAN) is a high bandwidth (allowing a high data transfer rate) computer network operating over a small area such as an office or group of offices.

Operating Environment: All outside influences that interface with the computer system.

Operating System: (a. PMA CSVC, b. ANSI/IEEE)

a. A set of programs provided with a computer that function as the interface between the hardware and the application programs.

b. Software that controls the execution of programs. An operating system may provide services, such as resource allocation, scheduling, input/output control, and data management.

Operational Qualification [OQ]: (PMA CSVC) Documented verification that the equipment-related system or subsystem performs as intended throughout representative or anticipated operating ranges.

Performance Qualification [PQ]: Documented verification that the process and/or the total process-related system performs as intended throughout all anticipated operating ranges.

Planned Change: (PMA CSVC) An intentional change to a validated system for which the implementation and evaluation program is predetermined.

PLC: Programmable Logic Controller.

Policy: (PMA CSVC) A directive usually specifying what is to be accomplished.

Procedure: (PMA CSVC) or **Standard operating Procedure (SOP):** A directive usually specifying how certain activities are to be accomplished.

Process System: (PMA CSVC) The combination of process equipment, support systems (such as utilities), and procedures used to execute a process.

Product: Any computer system supplied by the supplier to the customer as the result of an agreed contract between the two parties.

Prospective Validation: The validation of new or recently installed systems following a Life Cycle Concept (see PMA definition).

Proven Acceptable Range (PAR): All values of a given control parameter that fall between Acceptance Criteria (ANSI/IEEE): The criteria a software product must meet to successfully complete a test phase or to achieve delivery requirements.

Pseudo code: (ANSI/IEEE) A combination of programming language and a natural language used for computer program design.

Qualification Protocol: A prospective experimental plan that when executed is intended to produce documented evidence that a system or subsystem has been properly qualified.

Quality Assurance [QA]: (a. ANSI/IEEE, b. Dr J M Juran):

a. A planned and systematic pattern of all actions necessary to provide adequate confidence that the item or product conforms to established technical requirements.

b. The activity of providing, to all concerned, the evidence needed to establish confidence that the quality function is being performed adequately.

Quality Control [QC]: (Dr J M Juran): The regulatory process through which industry measures actual quality performance, compares it with standards, and acts on the difference.

Quality Function: The entire collection of activities from which fitness for use is achieved, no matter where these activities are performed.

Quality Plan: A plan created by the supplier to define actions, deliverables, responsibilities and procedures to satisfy the customer quality and validation requirements.

Quality Management System: The organisational structure, responsibilities, procedures, processes and resources for implementing quality management.

Range Testing: Checking each input output loop across its intended operating range.

Raw Data: Any work-sheets, records, memoranda, notes, or exact copies thereof, that are the result of original observations and activities and which are necessary for the reconstruction and evaluation of a work project, process or study report, etc. Raw data may be hard/paper copy or electronic but must be known and defined in system procedures.

Re-qualification: (PMA CSVC) Repetition of the qualification or a portion thereof.

Requirement: (ANSI/IEEE)

- A condition or capability needed by a user to solve a problem or achieve an objective
- A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed document. The set of all requirements forms the basis for subsequent development of the system or system component

Restore: Recovery of data files or software from a back-up, for restart of processing, or for use of alternative computer equipment after a system failure or a disaster (see back-up).

Retrospective Validation: (PMA CSVC) Establishing documented evidence that a system does what it purports to do based on an analysis of historical information.

Revalidation: Repetition of the validation process or a specific portion of it.

SAT: Site Acceptance Test; Acceptance testing at the customer's site.

SCADA: Supervisory Control And Data Acquisition.

Security: (IEEE) The protection of computer hardware and software from accidental or malicious access, use, modification, destruction, or disclosure. Security also pertains to personnel, data, communications, and the physical protection of computer installations.

Shall: This word is used in the example procedures throughout the appendices so that they may be used without alteration.

Should: The stated requirement is strongly recommended.

Simulation: (ANSI/IEEE/ISO) The representation of selected characteristics of the behavior of one physical or abstract system by another system. In a digital computer system, simulation is done by software; for example, (a) the representation of physical phenomena by means of operations performed by a computer system, (b) the representation of operations of a computer system by those of another computer system.

SLC: System Life Cycle (see life cycle concept).

Software: (PMA CSVC): A collection of programs, routines, and subroutines that controls the operation of a computer or a computerised system.

Software Life Cycle: (ANSI/IEEE) The period of time that starts when a software product is conceived and ends when the product is no longer available for use. The software life cycle typically includes a requirements phase, test phase, installation and checkout phase, and operation and maintenance phase.

Source Code: (PMA CSVC) An original computer program expressed in human-readable form (programming language), which must be translated into machine-readable form before it can be executed by the computer.

Specification Qualification: A documented evaluation of the detailed specification, carried out for the purpose of confirming compliance with the User Requirement and Functional Specifications and providing the detailed design documentation required for subsequent stages of validation (e.g. Installation and Operational Qualification) and ongoing operation of the facility or system in compliance with regulatory requirements related to product quality.

Standalone System: A self-contained computer system which provides data processing, monitoring or control functions but which is not embedded within automated equipment. This is contrasted with an embedded system, the sole purpose of which is to control a particular piece of automated equipment.

Structural Testing: (Bluhm, Meyers, Hetzel) Examining the internal structure of the source code. Includes low-level and high-level code review, path analysis, auditing of programming procedures, and standards actually used, inspection for extraneous "dead code", boundary analysis and other techniques. Requires specific computer science and programming expertise.

Sub-contractor: Any organisation or individual used by a supplier to provide material or services which are embodied in the product to be supplied.

Sub-program: A self contained program unit which forms part of a program. Sub-programs are sometimes referred to as procedures, subroutines or functions.

Supplier: Any organisation or individual contracted directly by the customer to supply a product.

System: An assembly of units consisting of one or more micro processors, associated hardware and all layers of system and application system.

System Acceptance Test Specification: Documented verification that the automated system or subsystem performs as defined in the Functional Specification throughout representative or anticipated operating ranges.

System Software: (ANSI/IEEE) Software designed for a specific computer system or family of computer systems to facilitate the operation and maintenance of the computer system and associated programs, for example, operating systems, compilers, utilities.

System Specifications: (PMA CSVC proposed) Describes how the system will meet the functional requirements.

Technical infrastructure: the operating system plus layered software and libraries, use to control the computer hardware and provide services to application software.

Tester: A person performing the test.

Testing: (IEEE) The process of exercising or evaluating a system or system component by manual or automated means to verify that it satisfies specified requirements or to identify differences between expected and actual results.

Testing: Structural & Functional: Both forms of testing are essential. Neither form of testing can be exhaustive. Structural testing should occur chiefly during software development.

Test procedure: A procedure which when executed successfully provides documentary evidence that part of the automated system works as specified.

Unplanned (Emergency) Change: (PMA CSVC) An unanticipated necessary change to a validated system requiring rapid implementation.

URS: User Requirements Specification.

User: The pharmaceutical / chemical industry customer or user organisation contracting a supplier to provide a product. In the context of this document it is, therefore, not intended to apply only to individuals who use the system, and is synonymous with Customer.

Utility Software: (ANSI/IEEE) Computer programs or routines designed to perform some general support function required by other application software, by the operating system, or by system users.

Validation: "Establishing documented evidence which provides a high degree of assurance that a specific process will consistently produce a product meeting its pre-determined specifications and quality attributes". - FDA Guidelines on General Principles of Process Validation, May 1987.

Validation Plan: A plan created by the customer to define validation activities, responsibilities and procedures.

Validation Protocol: (PMA CSVC) A prospective experimental plan that when executed is intended to produce documented evidence that the system has been validated.

Witness: A person observing the test and results.

Worst case: (FDA 1987) A set of conditions encompassing upper and lower processing limits and circumstances, including those within standard operating procedures, which pose the greatest chance of process or product failure when compared to ideal conditions. Such conditions do not necessarily induce product or process failure.