COMPUTER SOFTWARE VALIDATION IN PHARMACEUTICALS

Atul Bendale*, Nirav Patel1, D. P. Damahe1, Sachin B. Narkhede1, Anil G. Jadhav1, G. Vidyasagar2

1Smt. B. N. B. Swaminarayan Pharmacy College, Vapi, Gujarat, India
2Dean (Pharmacy Department) Kutch University, Bhuj, Gujarat, India

Received 20 August 2011, Accepted 07 October 2011

Abstract
To validate the computer system and computer assists software in pharmaceutical field. It is the process by which all aspects of a process (including computer systems) are shown to meet all quality requirements, and comply with applicable rules and regulations regarding product quality, safety and traceability. It is the technical discipline that pharmaceutical and life science companies use to ensure that each information technology application fulfills its intended purpose. For a process supported by a computer system, we can say that computer system validation provides documented proof that the system will repeatedly and reliably do what it is designed to do, is "fit-for-purpose", and complies with the applicable rules and regulations. Good computer system validations have many advantages like improve quality assurance, reduce other validation cost and time, improve GMP compliance and 21 CFR part 11 regulation which impact on product quality, safety, identity or efficacy that subject to GxP rules. It is likely that the future will see convergence of computer system validation terminology and techniques as a common technical discipline across other industry sectors as well.

Keywords: Computer Software Validation, Validation in Pharmaceuticals, 21 CFR 820.30 (g)

Introduction:
Validation is "establishing documented evidence that provides a high degree of assurance that a specific process will consistently produce a product meeting its pre-determined specifications and quality attributes." (FDA 1987) [1] A properly designed system will provide a high degree

* Corresponding author. Tel: +918000701337
E-mail address: atulbendale123@gmail.com (Atul Bendale)
© Pharmaboon. All rights reserved.
of assurance that every step, process, and change has been properly evaluated before its implementation. Testing a sample of a final product is not considered sufficient evidence that every product within a batch meets the required specification. Regulations and laws affect how pharmaceutical products are manufactured, ordered and received, handled in warehouses, distributed, processed and tested in laboratories, and also affect a multitude of other manufacturing, quality assurance, marketing and research and development activities. In essence, regulations control many aspects of how a pharmaceutical business is run. It is essential that all Computer systems involved in these and other processes are validated, and are supported and controlled by procedures and documentation that keep them in compliance.[2]

It is the technical discipline that life science companies use to ensure that each information technology application fulfills its intended purpose. Stringent quality requirements in FDA regulated industries impose the need for specific controls and procedures throughout the Software Development Life Cycle (SDLC). Evidence that these controls and procedures have been followed and that they have resulted in quality software (software that satisfies its requirements) must be documented correctly and completely. These documents must be capable of standing up to close scrutiny by trained inspectors since the financial penalty for failing an audit can be extremely high. More importantly, a problem in a life science software application that affects the production environment could result in serious adverse consequences, including possible loss of life.[2]

The activities involved in applying the appropriate controls/procedures throughout the SDLC and for creating the necessary trail of documented evidence are all part of the technical discipline of computer system validation. As we will discuss in this article, software testing is a key component in this discipline. However, computer system validation, involves more than what many it people consider to be software testing.[3,4]

**Regulatory Requirements for Software Validation:**
The FDA’s analysis of 3140 medical device recalls conducted between 1992 and 1998 reveals that 242 of them (7.7%) are attributable to software failures. [5,6] Of those software related recalls, 192 (or 79%) were caused by software defects that were introduced when changes were made to the software after its initial production and distribution. Software validation and other related good software engineering practices discussed in this guidance are a principal means of avoiding such defects and resultant recalls. Software validation is a requirement of the quality system regulation, which was published in the Federal register on October 7, 1996 and took effect on June 1, 1997.[7,8] validation requirements apply to software used as components in medical devices, to software that is itself a medical device, and to software used in production of the device or in implementation of the device manufacturer's quality system.

**Importance of computer system validation**
The FDA states that process validation is “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” 22. It is the process by which all aspects of a process (including buildings, equipment, and computer systems) are shown to meet all quality
requirements, and comply with applicable rules and regulations regarding product quality, safety and traceability. For a process supported by a computer system, we can say that computer system validation provides documented proof that the system (e.g. Hardware, software, peripherals and network) will repeatedly and reliably do what it is designed to do, is "fit-for-purpose", and complies with the applicable rules and regulations. Computer software validation must show that the system operates predictably according to its specifications, and that conclusion is supported by formal, documentary evidence because regulators will not take your word for it. The ultimate goal of any computer system validation project is to realize and sustain compliance, while ensuring the peak performance and functionality of those systems. Computer software validation what to validate, when and is a sound business practice that supports quality assurance, and promotes responsible and profitable operations. Computer software to validate, when and provides the evidence that a computer system does what it is intended to do according to the system specifications and operating procedures.

A key source document providing FDA guidance on the general topic of validation is "general principles of validation, food and drug administration" from the center for drug evaluation and research [17] Validation, as described in this document, is aimed at manufacturers of pharmaceuticals and medical devices who must demonstrate that their processes produce consistent product quality. It applies to all processes that fall under FDA regulation, including, but not limited to, computer systems. For example, validation applies to pharmaceutical manufacturing processes which include checking, cleaning, and documenting that all equipment used in manufacturing operates according to predetermined specifications. Computer system validation (or computerized system validation as it sometimes called in the literature) is the result of applying the above definition to a computer system:

"Establishing documented evidence which provides a high degree of assurance that a computer system will consistently produce results that meet its predetermined specification and quality attributes."[13]

Note that a "computer system" in the life sciences sector is more than computer hardware and software. It also includes the equipment and instruments linked to the system as well as the trained staff that operate the system and/or equipment using Standard Operating Procedures (SOPs) and manuals.

As applied to computer systems, the FDA definition of validation is an umbrella term that is broader than the way the term validation is commonly used in the industry. In the industry, validation usually refers to performing tests of software against its requirements[17]. A related term in the it world is verification, which usually refers to inspections, walkthroughs, and other reviews and activities aimed at ensuring that the results of successive steps in the software development cycle correctly embrace the intentions of the previous step [18,19].

As we will see below, FDA validation of computer systems includes all of these activities with a key focus on producing documented evidence that will be readily available for inspection by the FDA. So testing in the sense of executing the software is only one of multiple techniques used in computer system validation.[23]
There are two key reasons why computer system validation is extremely important in Pharma-sector:

1. Systematic computer system validation helps prevent software problems from reaching production environments.

2. FDA regulations mandate the need to perform computer system validation and these regulations have the impact of law. Failing an FDA audit can result in FDA inspectional observations ("483s") and warning letters.

A key point to be gleaned from 1 and 2 above is that not only do FDA regulated companies need to do computer system validation, but they need to do it right. Cutting corners on doing a validation might save a little money in the short term but these savings will look minute and inconsequential when compared to the potential costs and impacts of not doing the validation correctly.[7]

Need of validation [21]
Computers and automated equipment are used extensively throughout all aspects of research and development, laboratory testing and analysis, product inspection and acceptance, production and process control, environmental controls, packaging, labeling, traceability, document control, complaint management, and many other aspects of a pharmaceutical company's operations. Increasingly, automated plant floor operations can involve extensive use of embedded systems such as programmable logic and digital function controllers, statistical process control, SCADA, and robotics. In addition, software tools are frequently used to design, build, and test the software of computer systems. Other commercial software applications, such as word processors, spreadsheets, databases, and flowcharting software may be used to implement systems. All computerized equipment, systems, applications, tools and embedded systems that affect, monitor, or control product safety, quality, efficacy, or purity are subject to one or more of good manufacturing, laboratory, or clinical practice (GxP) and other applicable regulations and hence computer software validation.

Examples of Systems That Need To Be Compliant [8-10]:
The following list of examples identifies some of the systems that subject to the industries rules and regulations.
• The use of wireless technology to improve sales productivity is fast becoming the norm
• On-line customer service is powerful vehicle to improve customer satisfaction and reduce operating costs.
• Systems used to manage clinical trial data, whether provided internally or out sourced to a third party vendor, require validation.
• All record keeping related to raw materials, product and sample distribution, test sample management and clinical trial data, including any web-enabled interfaces;
• Systems that manage GxP training for employees;
• Because facility plans are required in an FDA submission and must meet specific facility requirements for manufacturing drugs, cad systems or any system that maintains engineering
plans and drawings of a facility might also require validation.

- Legacy systems are common sources of non-compliance with 21 CFR part 10
- The integration of computer systems has greatly improved productivity and visibility within operations. However, these hybrid systems that now include GxP functions and non-GxP functions in the same system must also be validated to ensure integrity and quality.

**Fit For Purpose Computer System Validation.**[20]

The meaning is the activities defined in the validation plan have been completed and the necessary actions to maintain the system in its validated state are taken, the system is validated. This is predicated on the validation plan being "fit for purpose". Completion of the activities defined in the validation plan will often require that:

- Validation plan / quality plan has been followed without major non-compliances;
- The system specifications provide for all GxP requirements to be met through automated processes and sops - GxP requirements may be implicit;
- The system is shown to comply with 21 CFR part 11 (e-records and e-signatures);
- The system is shown to perform consistently and correctly as specified;
- Data at start-up is validated;
- All documents have been written reviewed and approved by demonstrably

**Regulators For Computer System Validation** [22]

Computer software validation in an organization involved in any aspect of the pharmaceutical or medical device industry is a multi-disciplinary activity. Ultimately, senior company management may be held to account by the regulators should a system not be validated, though the effects of that may well cascade down the organization. Though the FDA and EU rules, regulations and guides identify quality assurance and quality control units as being responsible for the various processes that assure product or device safety, efficacy, strength, quality and purity, they do not allocate individual responsibilities for computer software validation. However, computer software validation is essentially a Quality Assurance activity, and that means that QA or its equivalent must at least be satisfied with the validation process at all stages, and signify that by signing the supporting documentary evidence that the validation has been conducted and completed to their satisfaction.

**Principles of software validation** [22,23]:

This section lists the general principles that should be considered for the validation of software.

**Requirements:** A documented software requirements specification provides a baseline for both validation and verification. The software validation process cannot be completed without an established software requirements specification.23

**Defect Prevention:** Software quality assurance needs to focus on preventing the introduction of defects into the software development process and not on trying to “test quality into” the software code after it is written.

**Time and Effort:** To build a case that the software is validated requires time and effort. Preparation for software validation should begin early, i.e., during design and development planning and design input. The final conclusion that the software is validated should be based on evidence collected from planned efforts conducted throughout the software lifecycle.
Software Life Cycle: Software validation takes place within the environment of an established software life cycle. The software life cycle contains software engineering tasks and documentation necessary to support the software validation effort. In addition, the software life cycle contains specific verification and validation tasks that are appropriate for the intended use of the software. This guidance does not recommend any particular life cycle models – only that they should be selected and used for a software development project.

Plans: The software validation process is defined and controlled through the use of a plan. The software validation plan defines “what” is to be accomplished through the software validation effort. Software validation plans are a significant quality system tool. Software validation plans specify areas such as scope, approach, resources, schedules and the types and extent of activities, tasks, and work items.

Procedures: The software validation process is executed through the use of procedures. These procedures establish “how” to conduct the software validation effort. The procedures should identify the specific actions or sequence of actions that must be taken to complete individual validation activities, tasks, and work items.

Software Validation after a Change: Due to the complexity of software, a seemingly small local change may have a significant global system impact. When any change (even a small change) is made to the software, the validation status of the software needs to be re-established. Design controls and appropriate regression testing provide the confidence that the software is validated after a software change.

Validation Coverage: Validation coverage should be based on the software’s complexity and safety risk – not on firm size or resource constraints. Validation documentation should be sufficient to demonstrate that all software validation plans and procedures have been completed successfully.

Independence of Review: Validation activities should be conducted using the basic quality assurance precept of “independence of review.” Self-validation is extremely difficult. When possible, an independent evaluation is always better, especially for higher risk applications.

Flexibility and Responsibility: Specific implementation of these software validation principles may be quite different from one application to another. Software is designed, developed, validated, and regulated in a wide spectrum of environments, and for a wide variety of devices with varying levels of risk.

FDA regulated medical device applications include software [7]:
Is a component, part, or accessory of a medical device; Is itself a medical device; or Is used in manufacturing, design and development, or other parts of the quality system.

In each environment, software components from many sources may be used to create the application (e.g., in-house developed software, off-the-shelf software, contract software,
shareware). In addition, software components come in many different forms (e.g., application software, operating systems, compilers, debuggers, configuration management tools, and many more). The resultant software validation process should be commensurate with the safety risk associated with the system, device, or process. Software validation activities and tasks may be dispersed, occurring at different locations and being conducted by different organizations. However, regardless of the distribution of tasks, contractual relations, source of components, or the development environment, the device manufacturer or specification developer retains ultimate responsibility for ensuring that the software is validated.

**Role of a life science company to determining the needs of specific computer system validation [20]**

The way an individual company approaches computer system validation is based on the company's interpretation of FDA regulations and FDA guidance documents as well as their efforts to adopt industry best practices. Best practices include life science industry group guidelines. Some of the FDA regulations provide rules on the quality system under which life sciences companies must operate known as the "regulated GxP environments". GxP is an umbrella term that covers: GMP (good manufacturing practice), GLP (good laboratory practice) and GCP (good clinical practice).

These codes/quality systems are sometimes referred to collectively as the predicate rules. Depending on the software application, different predicate rules may apply. For example, there are specific regulations that cover medical device software (21 CFR 820.30 (g)). Guidance on validation of medical device software is provided in an FDA paper called general principles of software validation: final guidance for industry and FDA staff [19]. In addition to the FDA regulations, FDA guidance documents, and best practices that apply, there are other factors/variables that affect what needs to be done in a specific computer system validation:

- The type of software that is being validated
- Whether the software is off-the-shelf, configurable or custom developed impacts the validation
- Business and compliance risks associated with the specific computer system should be used to determine validation priorities.

**A typical computer system validation (actual process)[19]:**

Computer system validation is definitely not a "one size fits all" procedure; the approach that an individual company may take to a specific validation depends on the rules, guidance, best practices, and characteristics of the system being validated. On the other hand there are some strong similarities between the activities in most computer system validations and the type of documentation produced. In fact one way to get a good understanding of computer system validation is to take a look at the type of documents that would be accumulated. The following is a list of the documents that might result from the validation of a computer system application to be used in a GxP sensitive environment:
Validation Master Plan and Project Plan: [25]
All validation activities should be described in a validation master plan which should provide a framework for thorough and consistent validation. A validation master plan is officially required by annex 15 to the European GMP directive. FDA regulations and guidelines mandate a validation master plan; however, inspectors want to know what the company’s approach towards validation is. The validation master plan is an ideal tool to communicate this approach both internally and to inspectors. It also ensures consistent implementation of validation practices and makes validation activities much more efficient. In case there are any questions as to why things have been done or not done, the validation master plan should give the answer.

Within an organization a validation master plan can be developed for: Multiple sites, Single sites, Single locations, Single system categories, Department categories, e.g., for development departments.

Computer validation master plans should include: Introduction with a scope of the plan, e.g., sites, systems, processes, Responsibilities by function, Related documents, e.g., risk management plans, Products/processes to be validated and/or qualified, Validation approach, e.g., system life cycle approach, Risk management approach with examples of risk categories and recommended validation tasks for different categories, Vendor management, Steps for computer system validation with examples on type and extent of testing, for example, for IQ, OQ and PQ, Handling existing computer systems, Validation of macros and spreadsheet calculations, Qualification of network infrastructure, Configuration management and change control procedures and templates, Back-up and recovery, Error handling and corrective actions, Requalification criteria, Contingency planning and disaster recovery, Maintenance and support, System retirement, Training plans (e.g., system operation, compliance), Validation deliverables and other documentation, Templates and references to sops, Glossary.

Relationship between computer system validation and 21 cfr part 11:[26]
In 1997, the FDA added rule 21 CFR part 11 to the code of federal regulations. This regulation introduces specific controls on the use of electronic records and includes strict administrative controls on electronic signatures. These controls deal with- Making electronic records suitable for supplanting paper records and making an electronic signature as secure and legally binding as a handwritten signature.
Regardless of whether or not a company uses electronic signatures, 21 CFR part 11 impacts all companies that use computer systems that create records in electronic form associated with the GxP environment. All computer systems in this category must have technical and administrative controls to ensure:- The ability to generate accurate and complete copies of records, The availability of time-stamped audit trails, The protection of records to enable accurate and ready retrieval, Appropriate system access and authority checks are enforced.

Validation of automated process equipment and quality system software:[28,29]
The quality system regulation requires that “when computers or automated data processing systems are used as part of production or the quality system, the [device] manufacturer shall validate computer software for its intended use according to an established protocol.” This has
been a regulatory requirement of FDA’s medical device good manufacturing practice (GMP) regulations since 1978. Computers and automated equipment are used extensively throughout all aspects of medical device design, laboratory testing and analysis, product inspection and acceptance, production and process control, environmental controls, packaging, labeling, traceability, document control, complaint management, and many other aspects of the quality system. Increasingly, automated plant floor operations can involve extensive use of embedded systems in:- Programmable logic controllers; Digital function controllers; Statistical process control; Supervisory control and data acquisition; Robotics; Human-machine interfaces; Input/output devices; and Computer operating systems.

Software tools are frequently used to design, build, and test the software that goes into an automated medical device. Many other commercial software applications, such as word processors, spreadsheets, databases, and flowcharting software are used to implement the quality system. The device manufacturer has latitude and flexibility in defining how validation of that software will be accomplished, but validation should be a key consideration in deciding how and by whom the software will be developed or from whom it will be purchased. The software developer defines a life cycle model.

**Special considerations [29]**
Commercial software used in electronic recordkeeping systems subject to part 11 needs to be validated, just as programs written by end users need to be validated. We do not consider commercial marketing alone to be sufficient proof of a program’s performance suitability. The end user is responsible for a program’s suitability as used in the regulatory environment. However, the end user’s validation approach for off-the-shelf software is somewhat different from what the developer does because the source code and development documentation are not usually available to the end user. End users should validate any program macros and draft guidance for industry - not for implementation. Other customizations that they prepare. End users should also be able to validate off- the-shelf software by performing all of the following: 10.1 end user requirements specifications end users should document their requirements specifications relative to part 11 requirements and other factors, as discussed above. The end user’s requirements specifications may be different from the developer’s specifications. If possible, the end user should obtain a copy of the developer’s requirements specifications for comparison. Where source code is not available for examination, end users should infer the adequacy of software structural integrity by doing all of the following: Conducting research into the program’s use history.

This research should include:

(1) Identifying known program limitations;
(2) Evaluating other end user experiences; and,
(3) Identifying known software problems and their resolution; evaluating the supplier’s software development activities to determine its conformance to contemporary standards. The evaluation should preferably be derived from a reliable audit of the software developer, performed by the end user’s organization or a trusted and competent third party. Draft guidance for industry - not for implementation.
Functional Testing of Software
End users should conduct functional testing of software that covers all functions of the program that the end user will use. Testing considerations discussed above should be applied. When the end user cannot directly review the program source code or development documentation (e.g., for most commercial off-the-shelf software, and for some contracted software,) more extensive functional testing might be warranted than when such documentation is available to the user. More extensive functional testing might also be warranted where general experience with a program is limited, or the software performance is highly significant to data/record integrity and authenticity. Note, however, we do not believe that functional testing alone is sufficient to establish software adequacy.

The Internet: We recognize the expanding role of the internet in electronic recordkeeping in the context of part 10. Vital records, such as clinical data reports or batch release approvals, can be transmitted from source to destination computing systems by way of the internet.

Internet Validation: We recognize that the internet, as computer system, cannot be validated because its configuration is dynamic. For example, when a record is transmitted from source to destination computers, various portions (or packets) of the record may travel along different draft guidance for industry - not for implementation 14 paths, a route that neither sender nor recipient can define or know ahead of time. In addition, entirely different paths might be used for subsequent transfers.

Conclusion
Systematic computer system validation helps to prevent software problems from production environment. A problem in a Pharmaceutical software application which affect the production environment can result in serious adverse consequence and also affect the product quality and business firm like lawsuits, financial penalties which ultimately results the company suffering from economic instabilities, staff downsizing and possibly eventual bankruptcy. FDA regulation mandate the need to perform computer software validation and these regulations has the impact of law. Failing in FDA audit can result in FDA inspectional observation and warning letters. And failure to take corrective action in a particular timing can results in shutting down manufacturing facilities, consent decrees, and stiff financial penalties. So computer software validation is very important for pharmaceutical companies and laboratories.

A computer system must be validated at the time of installation, before and during any project is running, any change in the software or computer system. A validation must be done by a qualified person who has completely information regarding the system and project to be done. Good computer system validations have many advantages like improve quality assurance, reduce other validation cost and time, improve GMP compliance and 21 CFR part 11 regulation which impact on product quality, safety, identity or efficacy that subject to GxP rules.

A typical computer system gives documented evidence like URS, validation plan, project plan, documentation justifying system selection including supplier audit report, functional specification, design specification, supplier test plan and results, task reports, traceability matrix,
risk assessment, network and infrastructure qualification, installation qualification strips and results, operational qualification scripts and results, SOP, training material an training records, performance qualification scripts and results, and finally validation reports. For validation of computer system a special validation master plan is developed for particular system and project. A special consideration of computer software validation includes validation of OTS software and internet validation.

**Table 1: Validation cost overcome by computer software validation [12]**

<table>
<thead>
<tr>
<th>Sr. No.</th>
<th>Cause</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Starting computer software validation at a late stage of an implementation project - this often necessitates the repetition of sometime and labour intensive project activities such as rewriting or improving specification documents, and rewriting of test specifications</td>
<td>Start validation at the start of the project - Design, development, build and test, the incremental costs due to validation will be minimized and all validation activities will also have value in the project itself. Apart from document and test management procedures and part 11 compliance, if an organization already has and respects a formal quality management system for system development, support, and maintenance, the project activities required will probably not be significantly different to those for an un-validated system.</td>
</tr>
<tr>
<td>2</td>
<td>Not focusing the validation on the most important issues and possibly validating more than is necessary</td>
<td>The second can be avoided by assessing the risk to quality and compliance up-front and continually during the project, and thereby define and justify the scope and extent of validation, and exclusions from it. In this way unnecessary validation activities can be eliminated and time and resources saved.</td>
</tr>
<tr>
<td>3</td>
<td>Required formal, approved test results, and difficulties of using automated test tools.</td>
<td>It is often the case that the first use of an automated test tool in a project (i.e. The first time the tests are written and used), the time and resources for this phase of the project may be greater than is the case for manual testing (both test writing and execution). Until recently there has been no possibility of automating testing in a gxp context. However, the recent development of e-signature, audit trails, and e-record protection in some test tools has made this a possible option. So the choice is now a computer software validation and project management decision, whereas previously it was solely a computer software validation issue.</td>
</tr>
</tbody>
</table>

**Table 2: Function of document in validation**

<table>
<thead>
<tr>
<th>Sr. No.</th>
<th>Document name</th>
<th>Function of document in validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>User requirements specification</td>
<td>Defines clearly and precisely what the user wants the system to do and states any constraints (e.g. Regulatory) under which the system must operate.</td>
</tr>
<tr>
<td>2</td>
<td>Validation plan</td>
<td>Defines the objectives of the validation and the activities, procedures and responsibilities for accomplishing the objectives of the validation. The validation plan should also deal with the approach for maintaining the validation status and issues like configuration management, change control, and system retirement.</td>
</tr>
<tr>
<td>3</td>
<td>Project plan</td>
<td>Details the tasks and time line for the project.</td>
</tr>
</tbody>
</table>
Summary

The objective of these activities is to document evidence that each computer system will fulfill its intended purpose in a GxP production, laboratory, or research operation. The intention is to avoid software problems that could have serious impact. Dynamic testing of the software is an important part of the computer system validation. But computer system validation is more than just this type of testing. Computer system validation requires a comprehensive set of equally important static testing activities that need to be conducted throughout the SDLC. This includes a variety of analyses, audits, walkthroughs, reviews, and traceability exercises. Documentation must be accumulated that demonstrates that these activities have been performed effectively.

Now a day, the term computer system validation refers specifically to the technical discipline used in the life sciences sector to help ensure that software systems meet their intended requirements. Through its regulations/guidance on computer system validation, the FDA has shaped it testing and analysis processes to match the needs and requirements of the industries it governs. As a result, computer system validation has become an integral part of doing business in FDA regulated environments. It should be noted, however, that significant progress has been made in achieving consistency and harmonization between FDA regulations/guidance on computer system validation and relevant international it standards and best practices. It is likely that the future will see convergence of computer system validation terminology and techniques as a common technical discipline across other industry sectors as well.

References:
4. European commission enterprise directorate-general (2001), final version of annex 15 to the guide to good manufacturing practice, qualification and validation, Brussels. European commission enterprise directorate-general, Page No. 90-96
5. International conference on harmonization (ICH) of technical requirements for the registration of pharmaceuticals for human use, validation of analytical procedures: definitions and terminology, Geneva (1996), Page No. 748-832
7. U.S. FDA - guidance for industry (draft) analytical procedures and methods validation: chemistry, manufacturing, and controls and documentation, 2000, Page No. 223-243
8. 21 code of federal regulations (CFR) part 820, Page No.76-97
10. 21 CFR 820 quality system regulation, Page No. 670-693
11. US FDA: guideline on general principles of process validation (1987), Page No. 52-67
12. Draft guidelines for the validation of blood establishment computer system, September 28, 1993, Page No. 432-463
14. General principle of software validation; final guidance for industry and FDA staff; January 11, 2002, Page No. 345-367
16. Blood establishment computer software validation in the user facility guideline, FDA, 2007, Page no, 25-1042
17. General principles of validation, food and drug administration, drug evaluation and research; Rockville, Maryland; 1987, Page No. 345-378
18. Perry, William e.; rice, Randall w.; surviving the top ten challenges of software testing, New York: Dorset house; 1999, Page No. 123-154
21. Petschenik, Nathan; system testing with an attitude; NEW YORK: Dorset house; expected to be published 2004
22. IEEE standard of software verification and validation; institute of electrical and electronics engineers (IEEE), NEW YORK; 1998, Page No. 1034-1065
23. General principles of software validation; final guidance for industry and FDA staff, document issued on: January 11, 2002, Page No. 345-367
24. Gamp 4 guide for validation of automated systems (2001), international society of pharmaceutical engineers (ISPE), Tampa, Fl and Brussels, Belgium, Page No. 567-599
27. General principles of software validation; final guidance for industry and FDA staff, document issued on: January 11, 2002, Page No. 345-367
29. Guidance for industry. 21 CFR part 11; electronic records and electronic signature validation, Page No. 1027-1094