Computer System Validation - It's More Than Just Testing

Introduction

Computer System Validation 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.

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.

What is Computer System Validation and Why is it Important?

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. [1]. The definition of Validation in this document is:

> Establishing documented evidence which provides a high degree of assurance that a specific process will consistently produce a product meeting its predetermined specification and quality attributes.

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.

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 (if any) 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 IT industry. In the IT industry, validation usually refers to performing tests of software against its requirements [2]. 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 [3, 4]. 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.

There are two key reasons why Computer System Validation is extremely important in the Life Science sector:

1. Systematic Computer System Validation helps prevent software problems from reaching production environments. As previously mentioned, a problem in a Life Science software application that affects the production environment can result in serious adverse consequences. Besides the obvious humanistic reasons that the Life Science sector strives to prevent such harm to people, the business consequences of a software failure affecting people adversely can include lawsuits, financial penalties and manufacturing facilities getting shut down. The ultimate result could be officers getting indicted, the company suffering economic instabilities, staff downsizing, and possibly eventual bankruptcy.

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. Failure to take corrective action in a timely manner can result in shutting down manufacturing facilities, consent decrees, and stiff financial penalties. Again, the ultimate result could be loss of jobs, indictment of responsible parties (usually the officers of a company), and companies suffering economic instabilities resulting in downsizing and possibly eventual bankruptcy.

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.

Relationship of Computer System Validation to the Software Development Life Cycle[1]

Computer System Validation is carried out through activities that occur throughout the entire Software Development Life Cycle (SDLC). The "V Diagram" (Figure 1) is widely used in the IT

---

[1] This section of the article draws freely from material that will appear in a new book by the author [5].

literature to emphasize the importance of testing and testing related activities at every step in the SDLC. The V-diagram is really a recasting of the oft-criticized "Waterfall" model of the SDLC. In fact the phases in the Waterfall Model are essentially the life cycle phases that appear on the left-hand side of the V-diagram. The V-diagram emphasizes the need for various forms of testing to be part of every step along the way. This avoids a "big-bang" testing effort at the very end of the process, which has been one of the main criticisms associated with the Waterfall model (or the way some have people have interpreted the Waterfall model).   The activities represented in the V-Diagram (labeled V &V in Figure 1) include Static Testing as well as Dynamic Testing activities. *Static Testing* (sometimes called Static Analysis) refers to inspections, walkthroughs, and other review/analysis activities that can be performed without actually executing the software. In *Dynamic Testing,* the software is actually executed and compared against expected results.   While many IT people use the term "testing" to mean dynamic testing, both dynamic and static testing activities are used in Computer System Validation to help ensure that the results of successive steps in the SDLC correctly fulfill the intentions of the previous step [4].

Different types of activities represented in the V-Diagram are sometimes distinguished by the terms Verification and Validation, words whose connotations in the IT world were discussed in the previous section. In some visualizations of the V-Diagram [see reference 3, for example], the term "Verification" is associated with the activities shown on the left hand side of the V and "Validation" associated with activities on the right hand side. At this point in time there are reasons why it may be preferable to avoid drawing this distinction. First, the IEEE definitions of these two terms have become so close [6] that it is hardly worth trying to articulate (or even remember) the difference. It is more productive to just call them V&V activities (as is done throughout the text of [6]).   Secondly, in companies regulated by the FDA and other regulatory bodies throughout the world, the term Validation is often used interchangeably with Computer System Validation when discussing the activities required to demonstrate that a software system meets its intended purpose.

In a sense, Computer System Validation has actually extended the V-Model and put a more user-driven spin on it.   As shown pictorially in Figure 2, Computer System Validation has several important features:

- Computer System Validation is driven by the "User". That is the organization choosing to apply the software to satisfy a business need is accountable for the Validation of that software. While the software supplier, the IT organization, the QA organization, and consultants can play important roles in a Computer System Validation, it is the User organization that is responsible for seeing that documented evidence supporting the Validation activities is accumulated.
- The User must write "User Requirements Specifications" (URS) to serve as the basis for a Computer System Validation. The URS provides the requirements the Computer System must fulfill for meeting business needs. A Computer System Validation cannot be done unless such a URS exists.
- The Supplier of the Computer System should provide Functional Specifications and Design Specifications, which satisfy the URS. Where such Specifications are not available for an existing system, they are sometimes developed by "reverse engineering" the functionality of the system.
- Users are involved in every step of the process (deeply involved for custom development, less for package based systems)
- A three level-structure is imposed on User Testing:
    – The Installation Qualification or IQ focuses on testing that the installation has been done correctly

- The Operational Qualification or OQ focuses on testing of functionality in the system installed at the User site
- The Performance Qualification or PQ focuses on testing that users, administrators, and IT support people trained in the SOPs can accomplish business objectives in the production environment even under worst case conditions.

How does a Life Science Company Determine What Needs to be Done in a Specific Computer System Validation?

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 (such as [7]) and IT standards (such as [6]). 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 (sometimes called Current Good Manufacturing Practice or cGMP)
- GLP: Good Laboratory Practice
- 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 [8].

The FDA has been striving to make its Quality System regulations consistent with the requirements for quality systems contained in applicable international standards. This includes the International Organization for Standards (ISO) ISO 9000 : 2000 "Quality Management Systems" and the ISO/CD 13485 ''Quality Systems—Medical Devices—Supplementary Requirements to ISO 9001''. So companies who follow these standards will find that Computer System Validation is well harmonized with their individual Quality Systems.

The GAMP Forum (Good Automated Manufacturing Processes Forum) focuses on the application of GxP to the IT environment. The GAMP Guide for Validation of Automated Systems [7] is said to be the most widely used, internationally accepted, guideline for validation of computer systems. The ISPE (International Society for Pharmaceutical Engineering) and the GAMP Forum jointly produce the GAMP Guide.

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:

1. The type of software that is being validated, e.g. Information Management, Business System, PLC or SCADA, Process Control, Platform/Infrastructure, etc. must be considered. GAMP defines categories of software and the validation strategies that correspond to these categories.

2. Whether the software is off-the-shelf, configurable or custom developed impacts the Validation. The more customized the software, the more

comprehensive the Validation due to the deep involvement the User should have in the supplier's software development processes.

3. In addition to performing Validations on computer systems that are brand new to the organization, a company may need to perform a validation on an existing legacy system that has never been validated. (a "retrospective" validation). Also; a significant change to a previously validated system may require a "revalidation". An individual Life Science company may establish slightly different approaches for prospective validations, retrospective validations, and revalidations.

4. Business and Compliance Risks associated with the specific Computer System should be used to determine validation priorities. Validations (and the associated testing, in particular) should focus on the areas with the highest risks.

A Typical Computer System Validation

As discussed in the previous section, 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 [see reference 9, for example]. 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[2]:

Document Name   Function of Document in Validation

Document Name	Function of Document in Validation
User requirements Specification (URS)	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.
Validation Plan	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 This will generally involve referencing the organization's Quality Management System documentation that deals with such issues as Configuration Management, Change Control, and System Retirement.
Project plan	Details the tasks and time line for the project.

---

[2] If associated equipment and instruments were involved, additional documents would be generated documenting aspects of the hardware qualification and commissioning.

Documentation justifying Selection of System including Supplier Audit Report	Outlines the reasons for choosing the system including the results of auditing the supplier's quality management system
Functional Specifications	Detailed specifications showing the functions that the system performs
Design Specifications	Detailed specification showing how the system performs the functions documented in the Functional Specifications
Supplier Test Plans and Results	Documentation of Supplier Testing
Task Reports	Documentation of Design/Specification/Testing Reviews, Walkthroughs, and Inspections
Traceability Matrix	Analysis document that shows mapping between URS, Functional Specs, Design Specs and test cases in IQ, OQ, PQ (see below)
Risk Assessments	A Risk Assessment (sometimes called Failure Mode and Effects Analysis), is an analysis of failure scenarios associated with each of the functions and sub functions of a system. Each failure scenario is examined for potential business impact and likelihood of occurrence in order to determine the relative risks associated with each function and sub function of the system. Risk assessments may need to be performed at multiple strategic points in the SDLC.
Network and Infrastructure Qualification	Documentation that shows that the network and infrastructure hardware/software supporting the application System being validated has been installed correctly and is functioning as intended [10]
Installation Qualification (IQ) Scripts and Results	• Test cases for checking that System has been installed correctly in User environment • Results of executing scripts • Deviations from expected results (if any)
Operational Qualification (OQ) Scripts and Results	• Test cases for checking that System does what it is intended to do in User environment • Results of executing scripts • Deviations from expected results (if any)
SOPs (Standard Operating Procedures), Training Material, and Training Records	Documented procedures for users, system administrators, and IT related functions such as Backup & Restore and Archiving. Training records must be kept to show the appropriate people were trained in the correct operation of the system.

Performance Qualification (PQ) Scripts and Results	• Test cases for checking that System does what it is intended to do with trained people following SOPs in the production environment even under worst case conditions • Results of executing scripts • Deviations from expected results (if any)
Validation Report	The Validation Report includes: • A review of all activities and documents against the Validation Plan • evidence that deviations (if any) have been addressed and the system is validated • the plan for ongoing activities to maintain validation status

Relationship Between Computer System Validation and 21 CFR Part 11

In 1997, the FDA added rule 21 CFR Part 11 to the Code of Federal Regulations [11]. This regulation introduces specific controls on the use of electronic records and includes strict administrative controls on electronic signatures. These controls deal with:

1. Making electronic records suitable for supplanting paper records.
2. 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 [12]. All computer systems in this category must have technical and administrative controls to ensure:

1. The ability to generate accurate and complete copies of records
2. The availability of time-stamped audit trails
3. The protection of records to enable accurate and ready retrieval
4. Appropriate system access and authority checks are enforced

From the point of view of Computer System Validation, 21 CFR Part 11 has two key impacts. First, it affirms that the FDA expects all computerized systems with GxP electronic records to be validated (just in case this was not obvious before). Secondly, 21 CFR Part 11 says that when you do a Validation of a particular Computer System, items 1 through 4 above automatically become part of the requirements for the System. This means that every Computer System Validation must assess whether the system being validated satisfies requirements 1 through 4 above and must identify deviations, if any, and corrective actions. Since FDA regulated companies are anxious to avoid deviations in their Validations wherever possible, most companies in the Life Science sector are currently in a proactive mode of assessing all of their systems for 21 CFR Part 11 compliance and addressing deviations through procedural remediation, technical remediation (e.g. software upgrades), or replacement of non-compliant systems with 21 CFR Part 11 compliant systems.

Summary and Conclusions

A Computer System Validation is a set of activities that FDA Regulated companies must conduct for each of their GxP sensitive computer systems. 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.

Today, 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

1. *General Principles of Validation, Food and Drug Administration*, Drug Evaluation and Research; Rockville, Maryland; 1987; (http://www.fda.gov/cder/guidance/pv.htm).

2. Kaner, Cem; Falk, Jack; Nguyen, Hung Quoc; *Testing Computer Software*; New York: John Wiley & Sons; 1999.

3. Perry, William E.; Rice, Randall W.; *Surviving the Top Ten Challenges of Software Testing*, New York: Dorset House; 1997

4. Deutsch, Michael S. ; *Software Verification and Validation: Realistic Project Approaches*, New Jersey: Prentice-Hall: 1982

5. Petschenik, Nathan; *System Testing with an Attitude;* New York: Dorset House; expected to be published 2004.

6. *IEEE Standard of Software Verification and Validation*; Institute of Electrical and Electronics Engineers (IEEE), New York; 1998

7. GAMP 4 Guide for Validation of Automated Systems (2001), *International Society of Pharmaceutical Engineers (ISPE)*, Tampa, FL and Brussels, Belgium

8. *General Principles of Software Validation: Final Guidance for Industry and FDA Staff* , U. S. Department of Health and Human Services, Food and Drug Administration, Center for

Devices and Radiological Health, Center for Biologics Evaluation and Research (http://www.fda.gov/cdrh/comp/guidance/938.html).

9. McDowall, R. D.; *Chromatography Data Systems III: Prospective Validation of a CDS;* (http://www.21cfrpart11.com/library.htm)

10. McDowall, R. D.; *Qualification of Computer Networks and Infrastructure;* American Pharmaceutical Review, Summer 2001 Issue.

11. *21 CFR Part 11 Electronic Records; Electronic Signatures; Final Rule*; Department of Health and Human Services, Food and Drug Administration; Federal Register, Vol. 62, No. 54, March 20, 1997.

12. Lande, Victoria V., *An Update on Electronic Records and Compliance*, Seminar presented by NuGenesis Technologies, 2001.

Figure 1

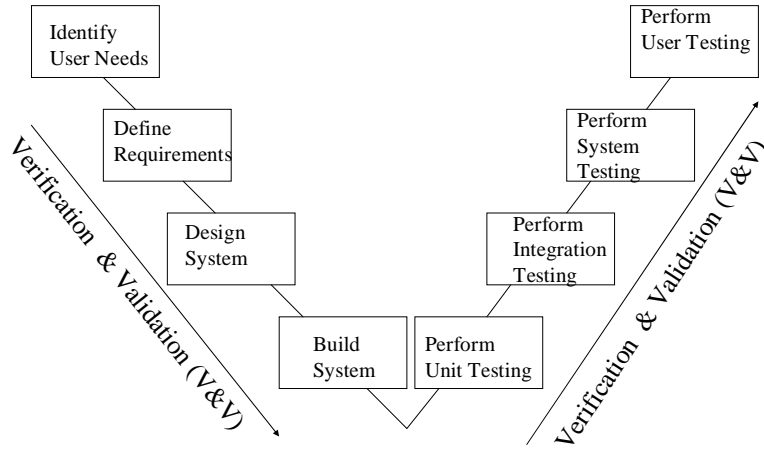## "V" Diagram

Identify User Needs

Define Requirements

Design System

Build System

Perform Unit Testing

Perform Integration Testing

Perform System Testing

Perform User Testing

Verification & Validation (V&V)

Verification & Validation (V&V)

Figure 2

Computer System Validation
in FDA Regulated Industries

Create User Requirement Specifications (URS)

Define Functional Specifications

Design System

Build System

Perform Unit Testing

Perform Integration Testing

Perform System Testing

Perform Installation Qualification

Perform Operational Qualification

Perform Performance Qualification

V & V

V & V

= User has primary responsibility

= Supplier has primary responsibility but User involved (deeply involved for custom development) and responsible for documenting evidence of quality