Automated System Validation

By: Daniel P. Olivier & Curtis M. Egan

In today's technical environment validation practices are both a requirement and an important tool in the medical and pharmaceutical industry. This paper will clarify the FDA and ISO requirements for computer system validation and provide helpful insight into conducting a validation project that increases confidence in quality and minimizes the cost for implementation.

Background

There are several factors that are increasing the visibility to the need for validation of automated systems used in support of pharmaceutical or medical device production and for software used in medical devices. These factors include:

a. FDA non-compliance observations for failure to validate computer systems used for production or used to support implementation of the quality system functions are becoming much more frequent today.
b. To stay competitive, manufacturers must increase reliance on computer systems to assist in the development, manufacturing and documentation of medial devices and pharmaceutical products.
c. Use of real time data captured from computerized systems is needed to implement six sigma and continuous improvement processes that are have been embraced to provide increased productivity and product quality.

As increased automation is embraced, the challenge then becomes how validation activities can be accomplished in a manner that meets the regulatory requirements but also integrates a business benefit that supports increased confidence in the reliability of data produced from these systems and minimizes the time and cost for conduct of the validation activities. This paper will define a general approach for conduct of validation and suggest strategies to satisfy these seemingly diverse objectives.

Approaching a Validation Project: The Validation Plan

It is important to lay the foundation for a validation project early, as in many cases there will be multiple individuals working together towards the same goal. Communication and scheduling are paramount to maintaining the project flow. One tool that can be employed to help set the foundation for the project is a validation plan.

A validation plan is a document that will outline the tasks, documents, and responsibilities of those involved in the project. This document will define roles and responsibilities and identify the activities and documents required to successfully complete the project. A validation plan should also include the scope of the project; this could include information on the portions of the system that are to be addressed functionally (most importantly those that require validation to satisfy regulatory requirements).

Defining System Requirements

The first phase of the validation is establishing requirements; the requirements are captured in a Requirements Specification. The requirements will be the basis for the validation protocols giving the framework for intended and functional use. The requirements will also provide the basis for the tester to define test steps for how the automated system is to function. As the requirements specification is the basis for the test protocols, it is essential that the requirements be clearly defined and testable.

Changes in the scope of the requirements will cause a waterfall effect throughout the documentation for the project. It is important to clearly define the requirements early to avoid rework that will lead to extended timelines and added costs.

Safety Risk Analysis

Safety concerns can also be factors in conducting validation activities. Safety risk analysis is becoming increasingly recognized by the FDA as an important method to support definition of the essential requirements for validation processes and medical device products. Failure modes and effects analysis (FMEA), fault tree analysis (FTA), and hazard and operational analysis (HAZOP) are different safety analysis techniques that can be used to identify safety related requirements for the system to be validated. The safety related requirements as defined in the safety risk analysis should be captured in the Requirements Specification where they establish the basis for testing.

A risk assessment can also be used to identify more detailed validation (testing) based on the criticality of the safety risk. This requires the validation team to consider and identify safety considerations as validation protocols are created. It is essential to ensure that the test protocols verify not only correct system performance, but also address the potential system failure cases (safety risks) to ensure that the system fails to a safe state if a failure does occur. The following are some areas that should be considered in addressing a safety risk analysis for production systems or applications used in support of a quality system requirement, much more detailed risk categories can be defined for medical device products addressing user errors and product misuse and abuse conditions.

- Identification of hardware components where failures can lead to an unsafe product.
- Identification of processing algorithms that are safety related (i.e.).
- Identifications of sensors, alarms, and interlocks where failure could result in the inability to detect or prevent a bad product or operator injury.
- Identification of operator command sequences that can lead to an out of specification product.
- Identification of “critical control points” that capture out of range conditions that could lead to a safety related failure.
- Identification of lost or corrupted data that can lead to a failure to make appropriate product disposition decisions.
- Identification of system interfaces that may be a source of lost of corrupted data.

Conducting Document Reviews

Periodic reviews are a necessity to ensure quality documents. Document reviews can be formal or informal, formal design review records should be retained for medical device products to fulfill the requirement for design reviews (21 CFR 820.30 (e)). These reviews should focus on the intent and content of the document, and ensure that any applicable regulatory requirements are adequately addressed by the application.

Validation Protocols

A validation protocol consists of executable test cases based upon the system requirements. There should be ample opportunity and instruction for the test engineer to provide objective evidence of the proper functioning of the system.

Validation test planning is necessary to identify the items to be tested and resources needed for testing including test equipment and documentation. Any tests that require external facilities or personnel, such as those that might be required for environmental testing should be identified and planned for accordingly.

For software, test planning should include an analysis of the type of testing to be performed. In general three levels of testing can be defined, including the following:

a) **Unit testing:** Unit testing is to address the specific implementation of a given program unit such as a function or class. This level of testing is also considered to be "white box" testing as the tests are defined based on being able to look into the program structure as though through a white or clear box.

   The emphasis on unit testing is to verify the correctness of the program unit as a standalone entity. All program units should be unit tested when they are initially coded. Unit tests for functions that support safety related requirements should be formally documented. Documentation for non-safety related requirements may be informal as defined in the Validation Plan. Other unit verification practices such as code reviews should also be used to verify the correctness of program units.

b) **Integration testing:** Integration testing is to test the integration of program units and subsystems. These tests are also to address interface data and protocols including loss of communication and corrupted data as well as the correctness of data passed between program units or subsystems.

   Dedicated integration test procedures may not be required for all development or validation efforts. For smaller development efforts, system test procedures may adequately address integration issues and separate integration test procedures may not be required. The Validation Plan should identify the integration tests that require formal documentation.

c) **System validation testing:** System validation testing is to ensure that the product conforms to defined user needs and intended uses as defined in the system level specifications and the safety risk analysis. To ensure completeness of testing, traceability of all requirements to test procedures is necessary. In addition, system validation tests are to address the following:

i. Compliance with applicable regulatory standards
ii. Tests in an actual or simulated use environment

The level of criticality of the software will be the primary guide as to the level of formality of unit and integration test procedures. Appropriate FDA and industry standards and guidelines should be referenced to support these decisions.

The validation protocol should provide detailed instructions for the software test engineer, outlining the method and pass/fail criteria for testing each system requirement. The validation protocol should include descriptions of special test equipment or test code, hardware setup, or any other details necessary to perform the tests. We suggest that the procedure be written as a form with the intent of each test being signed off by the tester, and test results documented. Further instructions to provide objective evidence should be included as well, such as indicating when screen captures should be made, when printouts should be generated and retained, and when discrete results of the test step are to be recorded.

Validation Summary Report

The validation summary report should indicate in a clear and concise manner the findings of the validation activities. Exception conditions, deviations from the written protocol, failures, and other notations relevant to the validation activities should be noted in the report. This report can be used by the validation team to determine the status of the project and identify any additional modifications, or further testing that would need to be conducted to support operational use of the system.

Validation Traceability Matrix

The purpose of a trace matrix is to ensure that all of the requirements have been tested, and show the test section where the requirements are tested. The Trace Matrix should also include a column to show reference numbers from the Safety Risk Analysis where applicable.

A sample safety and functional requirements traceability matrix is shown below.

Requirement Number	Requirement Description	Test Number	Safety Risk Reference
3.2	Start up RAM check	4	12, 13
3.3	Patient data entry	5, 6	
3.4	Temperature controls	8, 11	35, 36

All the requirements should have an entry and if a requirement is tested via some other method such as clinical trials or code inspection, it should also be noted in the matrix.

If some of the requirements incorporate design issues that cannot be functionally tested, they can be verified through code inspection, unit tests, or other method. These verification methods should also be documented and traced to the applicable requirements.

Summary

A validation project is an important step to providing your company with high confidence in the quality of your automated systems and assuring a defensible position for regulatory audits. The documents and activities mentioned in this article will assist you in developing, executing, and documenting a validation package that will give you a level of assurance that your system functions as expected, and is validated in compliance with the FDA regulatory requirements.