

# Validation Life Cycle

## Introduction

The life cycle approach to validation is a system engineering principle that ensures quality is built into the system and standards are established for the processes. The Validation Life Cycle starts when a system is determined to require validation and ends when it is retired. Validation is a good business practice and is required for all computer systems used in regulated areas. The scope of the validation effort may vary based on the components of the application (i.e., purchased, in-house, combination), complexity of the application, type of release (i.e., new, upgrade, emergency) and the application's business/regulatory criticality factor.

Validation is a planned and systematic quality assurance process. It provides documented evidence that:

- The system fulfills the business requirements.
- The system complies with regulatory requirements and company policies/procedures.
- The system is developed according to quality software engineering principles.
- The system operates as intended, throughout its lifecycle.
- Post-retirement, archived data associated with the system is maintained securely and can be retrieved successfully throughout the record retention life cycle.

## Phases

The Validation Life Cycle takes place within the environment of a System Development Life Cycle (SDLC) as explained below (**Figure 1**). The phases and their corresponding activities/deliverables are elaborated on page 2 of this document.

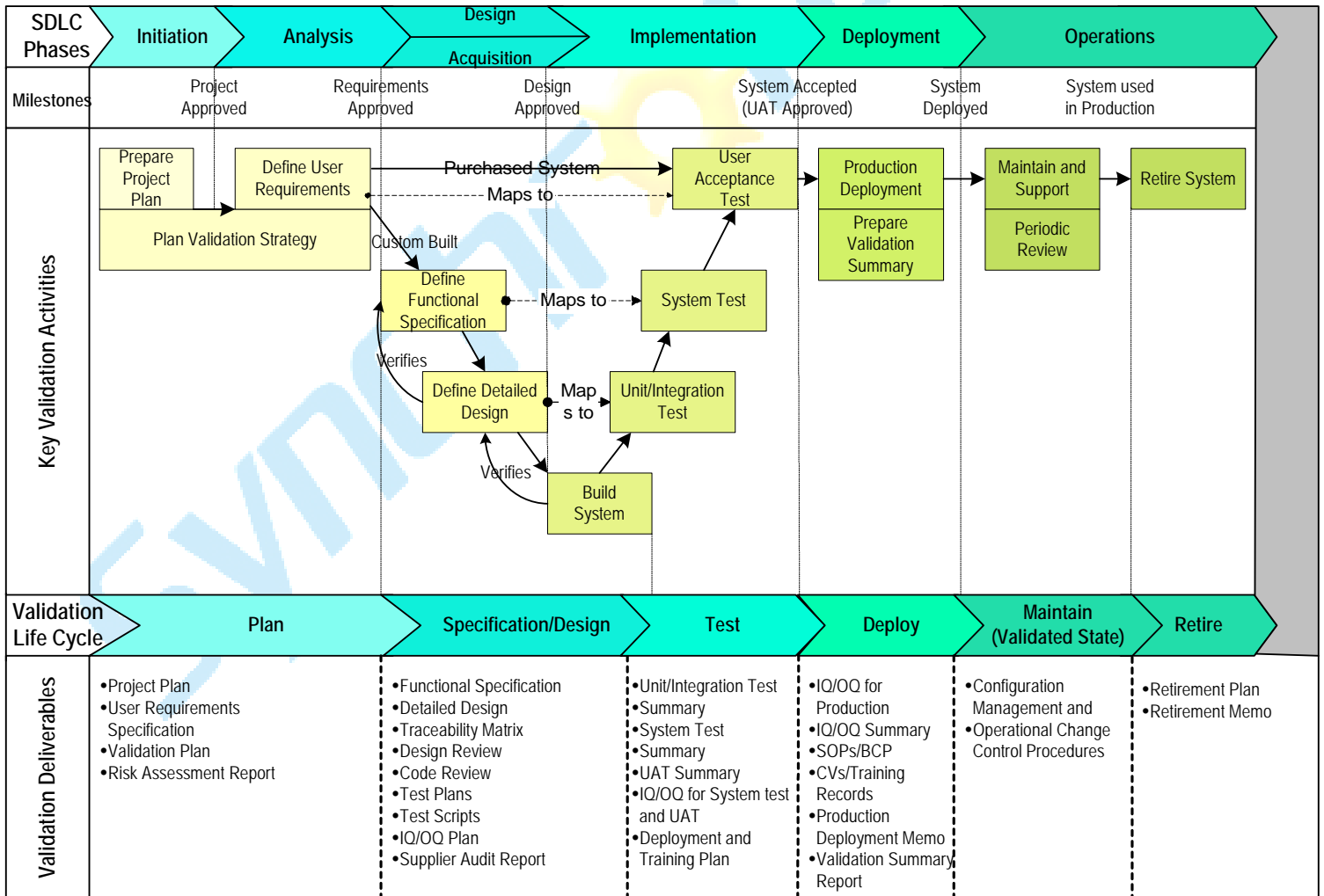


Figure 1

## 1. Plan (Validation)

- Prepare Project Plan; perform a risk assessment for validation determination.
- Assemble validation team; create the Validation Plan. *(The Validation Plan is a detailed plan that outlines the entire validation process. It typically, provides a system overview and details about the validation approach, scope of validation, Validation/SDLC activities, acceptance criteria, deliverables and roles/responsibilities. While Quality Assurance normally maintains oversight over the entire validation process the roles/responsibilities of the individual deliverables/tasks should be clearly defined).*
- Define the User Requirements Specification.

## 2. Specification/Design

For purchased components, perform a formal assessment of the vendor's quality assurance practices; this may involve a vendor audit.

- Define the Functional, Design Specifications. Conduct design reviews and establish traceability between the User Requirements and Functional Specification.
- Develop the system (may involve coding, customization and/or configuration of components). Follow coding standards and conduct code reviews.
- Develop Unit and Integration Test scripts (based on the Design Specification).
- Develop the System Test Plan and base the scripts on the Functional Specification. Establish traceability between the Functional Specification and System Test cases. For purchased systems, the above activities are normally performed by the vendor.
- Develop the User Acceptance Test (UAT) Plan and base the scripts on the User Requirements. Establish traceability between the User Requirements and UAT cases.

## 3. Test

- Perform Unit/Integration tests and summarize results. Review test evidence and summary before commencing with System Test execution.
- Review and approve System Test Plan. Perform Installation Qualification/Operational Qualification (IQ/OQ) for the System Test environment. Complete System testing (functional testing) and summarize results. Review test evidence and summary before UAT execution.
- Review and approve UAT Plan. Perform IQ/OQ for the UAT environment. Complete UAT execution and summarize results. Review test evidence and summary before production deployment.
- Update traceability matrices, as required.
- Finalize Deployment and Training Plans.

For purchased components, the vendor normally performs the unit/integration and system testing; the results are reviewed by the user.

## 4. Deploy

Update and verify the necessary documentation and Standard Operating Procedures (SOPs) including: user manual(s); Business Continuity Plan (BCP); SOPs for backup/recovery, archiving/restoration, change control and configuration management, security, disaster contingency, periodic review and incident management. Perform data migration (if required). Issue production memo and notify users. Summarize validation activities and perform Production IQ/OQ.

## 5. Maintain (Validated State)

Archive the validation package. Document changes to the system according to change management procedures. Perform re-validation based upon risk assessment. Update documentation and perform periodic reviews.

## 6. Retire

Execute System retirement according to the procedures outlined in the formal Retirement Plan. Perform data archiving/data migration and verify retirement activity. Retain records per applicable retention policies. Issue Retirement Memo.