

Windchill System Validation

Technical Brief

Contents

| | |
|---|-----------|
| About This Document..... | 2 |
| Supported Releases and Products | 2 |
| Audience | 2 |
| Support Policy for Enterprise Deployment Resources..... | 2 |
| Challenge..... | 3 |
| Solution | 3 |
| Planning | 4 |
| Preparing..... | 5 |
| Hardware..... | 6 |
| Software | 6 |
| Data | 7 |
| Tests..... | 8 |
| Implementing..... | 9 |
| Installation and Configuration..... | 10 |
| Data Integrity Tests | 10 |
| Functional Tests | 11 |
| Solution Tests..... | 11 |
| Performance Tests..... | 12 |
| Scalability Tests | 14 |
| User Acceptance Tests | 15 |
| Other Tests..... | 16 |
| Governing..... | 16 |
| Maintaining..... | 17 |
| References | 17 |
| Deployment Diagrams | 18 |

About This Document

Use this document to help plan and implement validation activities in the phases leading up to a Windchill enterprise system deployment. The activities can help ensure project deployment goals are met and that well-informed production go-live decisions are made. You can use these methods by themselves or to supplement your existing standard validation practices.

Supported Releases and Products

Windchill® 10.0 & Windchill 10.1
Windchill PDMLink®
Windchill ProjectLink™
Pro/INTRALINK® 10.0 & Pro/INTRALINK 10.1

Audience

This document is intended for managers, system administrators, developers, and solution architects. It assumes that the reader has some working knowledge of the technology and concepts associated with the following:

- Information technology
- Enterprise systems
- System testing and validation
- Project management
- Operating systems, relational databases, LDAP, web servers, application servers, platforms, servers, storage, networking
- Windchill architecture, capabilities, and system administration

Support Policy for Enterprise Deployment Resources

This document is a resource that PTC has created to assist customers with implementation and systems integration on specified system configurations. This document is not PTC product documentation and it may not be updated in the same way as PTC product documentation. Before using this document, verify that this document is applicable to the versions of PTC products that you are using. Because of the almost infinite variety of system configurations that customers can have, PTC cannot and does not warrant that this document will be entirely accurate for your particular configuration. Using the information in this document may require specific knowledge, skills, or expertise with PTC or third-party products. If incorrectly implemented, some of the information provided can have significant negative consequences.

Should you need consulting help with this document, you may be able to engage the PTC Global Services organization or a PTC services partner who can assist your company with its deployment of Windchill. If you require such assistance, contact your PTC sales representative or authorized reseller.

If you would like to provide feedback to PTC on this document or have a specific technical question about this document or about a PTC product, contact PTC Technical Support for assistance. Support for this document is provided in English. PTC Technical Support will endeavor to reply within two business days to requests for support of this document.

Challenge

The challenge for customers is to gain confidence in their enterprise system deployment activities prior to global production go-live events. Completing a thorough review and test of the system, migrated data, and end-to-end usage scenarios is very important before releasing a system for production use.

Typical considerations and questions that need to be answered include the following:

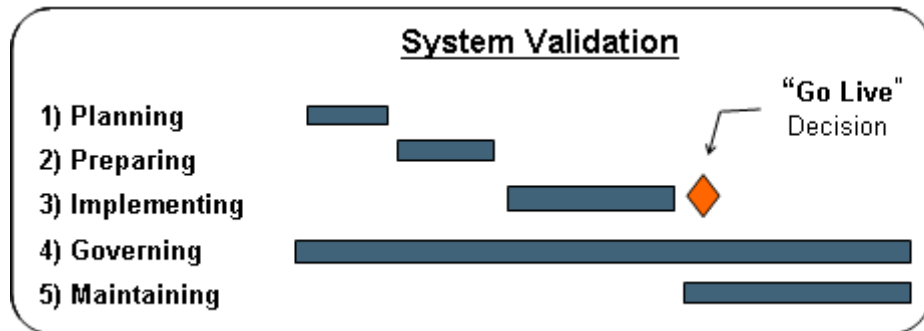
- Is data correctly upgraded/migrated from the source system?
- Are critical features and workflows performing as required?
- Does the new software version impact pre-existing behavior?
- Do workflows operate consistently using production data?
- Do all software applications interact as needed?
- Does the new hardware and/or configuration perform optimally?
- How does the system perform at peak user loads?
- How does the system perform using large data operations?
- Is there a performance impact due to any customizations?

Solution

It is important to document and implement a company-specific validation plan to ensure that the enterprise solution meets the deployment goals of the project. A well-documented plan that produces traceable test results is invaluable when preparing to make go-live decisions. Additionally, you can use the test results for comparison with production use and during maintenance upgrade testing when needed.

Deployment validation activities include the following:

1. [Planning](#) – Create and approve a validation plan
2. [Preparing](#) – Setup for validation activities
3. [Implementing](#) – Perform validation testing
4. [Governing](#) – Report and track validation results
5. [Maintaining](#) – Perform periodic diagnostics



The sections that follow expand on the activities and are intended to provide guidance to help validate your Windchill System for production deployment. Information can be maintained in a single plan or a number of sub-plans may be created and referenced.

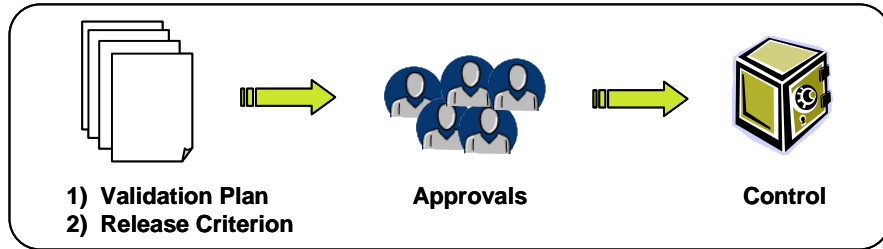
Planning

The validation plan must be detailed and approved by the team and key business stakeholders. A collaborative approval effort helps bring focus to the activity, ensures the proper setting of user and management expectations, and establishes an agreed acceptance criterion needed to make a well-educated go-live decision.

A typical validation plan can include the following topics. Detailed practices and considerations are described in later sections of this document.

- **Objective** – outlines the purpose, brief history, and goals of the project
- **Scope** – defines the configuration, software items and features to be tested
- **Strategy** – identifies the approach to test key features and characteristics
- **Environment** – specifies the required test configuration, data, and tools
- **Staffing** – names the responsible personnel and stakeholders
- **Schedule** – lists key milestones and planned start and end dates
- **Measurement and Criteria** – defines a means to gauge acceptance
- **Reporting and Tracking** – documents measured results over time
- **Governance and Risk Management** – defines oversight and contingencies

After the validation plan is approved, the document and its deliverables should be maintained and version controlled.



Preparing

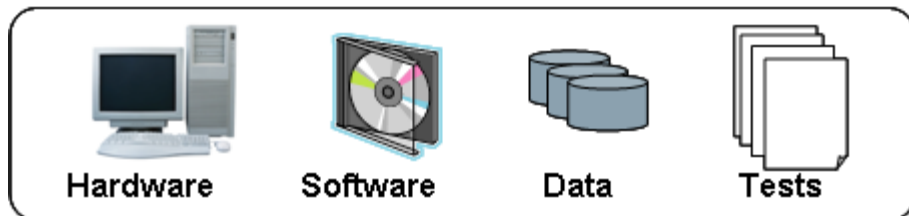
Typically, the validation or test system serves dual purposes; first it is used to pre-qualify the production deployment.

Then, after the production system has been deployed, the test system is used to qualify any planned changes (configuration or code) before applying them to the production environment.

Note: For these reasons, it is important that the test system preparation and setup closely match that which is in or will be in production.

Matching production and test system characteristics should include:

- [Hardware](#) types, operating systems, and patch level releases
- [Software](#) applications, compatible release versions, and customizations
- [Data](#) that is most recently re-hosted from the production environment
- [Tests](#) that represent functional, non-functional, and end-to-end production scenarios



The [PTC Technical Support website](#) and [Windchill Knowledge Base](#) should be actively referenced during system preparation activities. Standard documentation, TAN/TPI's, and suggested techniques can provide valuable information pertaining to all aspects of your planned deployment.

Hardware

It is important to plan for, procure, and deploy hardware that supports the various activities needed to ensure a successful production deployment of Windchill. The test system environment is the primary focus here, but it is important to know that other environments that can also be utilized.

System environments and their intended purpose are specified in the table that follows. Sample deployment methods are included at the end of this document.

| Environment | Description and Purpose |
|---------------------|--|
| Production | Active, business, working environment |
| Sandbox | Area for unstructured evaluation of software |
| Development | Location where customization and unit testing occurs |
| Test System | Mirror of the production environment for validation |
| Upgrade / Migration | Practice data upgrade/migration environment |
| Training | System used for training end users |
| Production Backup | Clone of production for testing and staging upgrades |
| Disaster Recovery | Production data and system recovery |

* Environment needs depend on company-specific requirements

The test system is setup and maintained separately, but is similar to the production environment. It is important to document and control the definition, setup, and change of these environments. For these reasons, PTC recommends that you create and maintain a [Configuration Definition and Change Log](#) for each environment.

The Configuration Definition and Change Log, if properly maintained, is internally valuable and can also be used to communicate with PTC Technical Support personnel to help report and diagnose issues more quickly. Information such as Windchill configuration files, customized source code, patches and the log itself can be managed in Windchill ProjectLink or other source code control management systems.

Software

To document the system configuration, it is important to list all software applications, release versions and patches that are being targeted. Software releases must be compatible with the hardware and operating system release levels on which it is being deployed. Additionally, release version compatibility between interfacing software products, must be planned and documented as appropriate. Reference the [Windchill Deployment Planning Checklist](#) as a planning tool and resource.

If product customizations are part of the intended solution, you should require a formal definition of these development efforts and their installation. Customizations must also be carefully considered during validation and when appropriate, during the upgrade process.

Data

Preparation for system validation also requires that data is available in the fully configured software/hardware testing environment.

It is important to provide representative data that is used to support the initial functional testing. This data can help validate specific operational needs and general workflows.

Data is typically added to a system in the following ways:

- Manually created in the system
- Checked in from disk
- Imported or loaded using custom scripts

The data that is required should be specified in the test case documents that are derived from the validation strategy.

Example characteristics of the prepared data and configuration should include:

- Typical datasets
- Mission critical datasets
- Production attributes
- Production workflows
- Production users, groups, and access controls
- Extreme sized data
- Extreme complexity of data
- Customized application data (as needed)

For the more advanced phases of testing (for example: end-to-end solution, performance, and scalability), it is important to have the test system loaded with a recent copy of the complete production system data.

If you are re-hosting the production data from a practice upgrade or migration activity, be sure to perform data validation activities to confirm system integrity.

Test results using production data cannot be considered valid until a successful migration or upgrade (performed without errors reported) is confirmed.

Tests

Test case definitions are written in support of the validation plan. Test cases are specified that follow the proposed testing strategy. They include several important characteristics; they are independent of test category and are created for both manual and automated tests.

The following test case template includes some of the important characteristics that are necessary to properly perform and document the validation tasks:

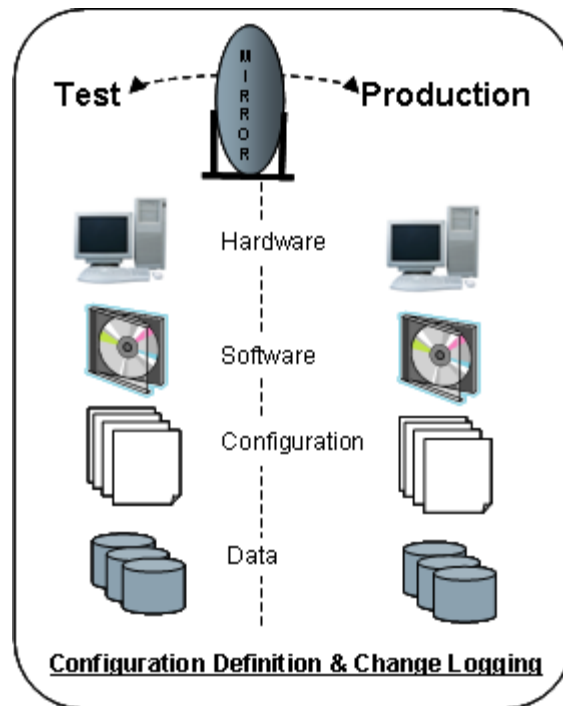
| Test # 1 | | | |
|---|------------------------|------------------|-----------------------------|
| Description: | | | Result : Pass / Fail |
| Requirement #: | Tester: | | Date : |
| User Role: | | | |
| Data / System Preconditions: <state of session and/or data at start of test> | | | |
| Step # | Execution Instructions | Expected Results | Actual Results |
| 1 | | | |
| 2 | | | |
| Data / System Post conditions: <state of session and/or data at end of test> | | | |

Test case descriptions and results can be documented in a test case management application or simply recorded in Microsoft Word or Excel files. The test case descriptions and results can be located in one or several files or repositories, but they should be centrally managed and controlled artifacts.

Key definitions and considerations from the test case template include:

- Test Case ID – unique identifier for traceability and reporting
- Description – clear description of purpose and feature being tested
- Requirement – traceable “feature ID” or “standard” being tested
- Pre-conditions – all input required to successfully execute the test
- Instructions – defined step by step test execution procedure
- Expected result – step by step anticipated result to “pass”
- Actual results – step by step actual recorded result to “pass/fail”
- Post-conditions – final expected output or state of system after test
- Pass/Fail – disposition of test result as recorded by the tester

Test results are compiled and reported in accordance with the validation plan.



A properly planned and prepared validation system should closely match that of the target production environment – hardware, software, configuration and data.

Implementing

Validation testing is performed to help ensure the production system successfully operates when deployed. During implementation, the test strategy is carried out using the test cases, system, and data that have been prepared for this activity.

The strategy can specify the following featured test cases and activities:

- [Installation and Configuration](#)
- [Data Integrity Tests](#)
- [Functional Tests](#)
- [Solution Tests](#)
- [Performance Tests](#)
- [Scalability Tests](#)
- [User Acceptance Tests](#)

Achieving successful results from the validation process is critical. It is especially important that all test cases relating to the release criterion (from the validation plan) be executed, recorded, and yield acceptable results. The release criterion is tied to the success of the project as defined and approved by key stakeholders.

The following sections describe some of the proven test methodologies and details that can be used in the enterprise system validation process.

Installation and Configuration

Although system setup, installation, and configuration is done as part of validation preparation, it is extremely important to document the steps followed for repeatability and to track any ongoing changes that can be required as a result of the validation process. Documented changes can include:

- Adding and removing debug settings
- Tracking system and application patch histories
- Tracking configuration changes
- Tracking customization release updates

Record the initial configuration and all changes in a [Configuration Definition and Change Log](#). After making any changes, run appropriate system tests called short checks to confirm stability of the operational baseline. Then before beginning or continuing to implement the validation test strategy, run specific tests to validate the changes that were applied.

Data Integrity Tests

Ultimately, a full copy of production data is needed to be validated in the test system. When moving to a new release of Windchill, re-hosted data can come from the results of a practice upgrade (data coming from an earlier release of Windchill) or migration (data coming from a dissimilar system – Pro/INTRALINK 3.x, 3rd party or proprietary system).

When re-hosting the production data from a practice migration activity, you must run the [Windchill Diagnostic Utility \(WINDU\)](#) as a first step to ensure the migration activity was successful. For example, complete the following steps:

1. Run the WINDU script.
2. Review the results from the log file that is created.
3. Report any errors to PTC Technical Support.

Test results using this production data cannot be considered valid until a successful migration of the data (performed without errors reported) is confirmed.

The test plan and strategy should include additional data validation activities. Selecting a good cross section of mission critical data is important. Manual testing can be augmented by automated techniques if desired. Publishing viewables and automating the execution of CAD Data Management operations should be considered and will help provide increased confidence.

Functional Tests

A traditional focus of the validation process includes functional testing. Functional testing is performed to ensure that the operational aspects of an application are performing as designed and documented. Categorize and prioritize functional test cases for execution. Typically, there are many test case files that are assigned to IT and participating end users.

Functional test case considerations include:

- Commonly used features
- Critically important features
- Historically problematic use cases
- Expected code fixes & improvements
- Extreme use cases
- Negative test cases

When customizations are planned, it is important to perform an out-of-the-box (OOTB) baseline test in the area being changed before the customizations are applied. This is important to help guard against or troubleshoot any inadvertent functional and performance regressions with the applied customizations.

After customizations are installed, unit tests (first created and run on the development systems) can be run on the test system in combination with other tests that together will ultimately make up the complete functional test cases for the production environment.

As part of completing the functional test execution, results are logged for analysis and potential action items are assigned in support of making the final release decision.

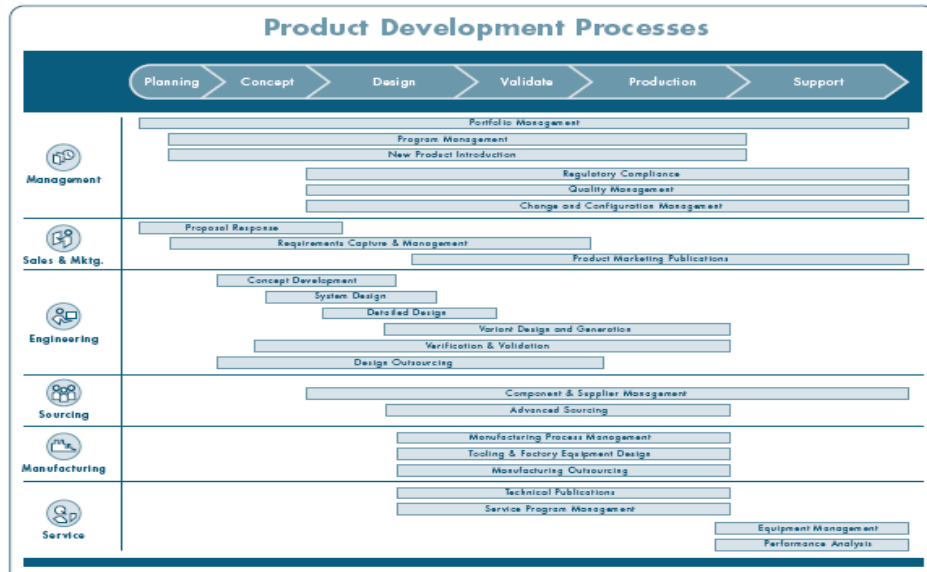
Solution Tests

Ultimately, solution testing ensures that all system components and applications interact properly to provide complete, production worthy results. Solution testing involves end-to-end testing of scenarios that mimic all life cycle phases of production activities. For this reason, all system architecture, third party components, and data must be installed so that the test environment closely matches that of the targeted production environment.

Solution testing and environmental considerations include:

- Business process scenario definitions
- Workflows, lifecycle states, and transition rules
- Attributes and soft type definitions
- Object initialization rules and numbering
- Template and folder definitions and structure
- Users, groups, roles, and team assignments
- Access control logic and security
- Multi-Product interoperability (functions and data)

PTC is focused on addressing the business process needs of our customers. To that end, the [PTC Business Process Landscape](#) is defined to closely resemble customer practices. These processes are defined to help focus PTC development and testing efforts to ensure that customer needs are met.



Because solution testing most closely resembles that of production use, you should create scenarios that target standard company practices and important user acceptance needs. Pre-qualifying user needs in a multiuser environment can help streamline the release process and facilitate user adoption.

Solution test case scenarios can be documented using the test case template format previously discussed. Likewise, you should compile and log test results for analysis, and assign potential actions items in support of making the final release decision.

Performance Tests

Performance testing helps define the means by which to measure system performance. Use the information obtained from this section to develop performance testing practices.

Performance testing is not intended to define the methods to design, configure, or tune the system for validation or production use. There are many other factors, materials, and tools that support the configuration and tuning process. PTC Global Services and certified deployment partners provide offerings and experts to assist in this activity. There are also links to performance tuning and configuration related information in the [References](#) section of this document.

Performance testing can be used to validate both individual operations, as well as complete end-user scenarios. In most cases, performance is measured to assist in system tuning activities and to guard against regression. In other cases, there may be published improvements that are being targeted. Testing can help determine how the new published performance goals can be achieved.

To measure performance improvements or guard against regression, you must have performance test baseline results from the current production release. You can obtain performance baseline test results using both manual (see [Windchill Creo Data Management Performance Benchmark](#)) and automated testing approaches (see [Windchill Single User Performance Tester](#)). You can record the precise timing for manually executed tests using system based tools.

In addition to specifying tests using extreme sized data (commonly referred to as stress testing), you should also measure the performance of the most common activities that users do each day. Capture and maintain these results for reference using standard reports. For example, the following table outlines tasks of interest and color codes the delta time to visually show where there are improvements (green) and degradations (red):

| Execution Tasks | Details and commentary | Prod. version A (baseline – sec. | Prod. version B (target – sec.) | Delta Time (+/- %) |
|---------------------|------------------------|--------------------------------------|--------------------------------------|-------------------------|
| Operation 1 | | | | - % |
| Operation 2 | | | | - % |
| Scenario 1 | | | | |
| - step 1 | | | | - % |
| - step 2 | | | | + % |
| Scenario 1 Total | | | | - % |

When capturing performance baseline results, it is important to document the exact system environment conditions so that the conditions can be re-established for release comparison testing. The [Configuration Definition and Change Log](#) can be a convenient place to store and reference this information.

Important baseline test environment conditions include:

- Product release versions
- Product maintenance and patch updates
- Server specifications and configurations
- Server load (real or simulated)
- Client specifications and configurations
- Test location and network (LAN/WAN) conditions
- Test data (typical and extreme scale)

Maintaining performance baselines is extremely valuable and can be used to assist in the following specific activities:

- **Upgrading to a new release** – Before moving to the new release of an existing Windchill deployment, baseline the performance measurements of the current test and production environments. Existing results can then be compared with those of the new target release environments.
- **Maintaining the current release** – Once in production, take periodic performance measurements against the production server to ensure performance reliability. Consistently take these measurements from the various locations, at specified times, using the same hardware, data, and test cases that were used to generate baseline results. Ideally, automation can be leveraged to ensure consistency, repeatability, and a lower cost to implement.
- **Migrating from another product** – When moving to Windchill from another PLM solution, it is not meaningful to measure productivity gains operation by operation. In some cases, there are potential scenario differences as well as the need to account for workflow benefits moving to a full featured, enterprise PLM solution.

If desired, the approach documented and published in [Using the PTC Day-in-the-Life Performance Benchmark](#) can be used to estimate overall productivity gains by moving to Windchill.

In summary, each company has users who perform like sets of tasks in their day-to-day activities and, therefore, can be grouped into specific user profiles. For each group, document the tasks and weigh them based on frequency of use. You can then use this information to calculate their day-in-the-life experience.

After measuring the performance of each operation, weighted averages can be applied to the appropriate user profiles. The results can be used to provide an overall productivity comparison between the legacy system and target Windchill deployment. This method more accurately captures true productivity gains in moving to a new Windchill solution.

PTC Global Services and certified deployment partners have tools to help facilitate these types of complex studies and deployments.

Scalability Tests

Windchill is designed and proven to be a fully scalable enterprise solution. In this context, PTC works closely with each of the supported platform vendors to determine the resource requirements needed with increasing Windchill system utilization.

You can perform scalability testing to ensure that the hardware system architecture specified is properly sized and configured to meet your planned utilization needs. Use the appropriate Windchill hardware sizing guide when designing the Windchill architecture for a specific hardware platform and release. You can access the hardware sizing guides from the PTC Reference Documents Web site as described in the [References](#) section, later in this brief.

After the system is deployed and configured, the scalability testing strategy can be carried out. Although you may want to simulate full multiuser loading of the validation system, simulating this is not practical without automated tools and accurate utilization estimates of the production environment. That being said, PTC experiences have shown that a properly sized and deployed configuration handles the production load as designed.

If multiuser production load simulation is desirable, PTC Global Services and certified deployment partners have tools and practices to help facilitate these types of simulations. In such cases, JMeter (open source) and LoadRunner (from HP) have been utilized.

Manual multiuser testing cannot practically utilize the full system capacity but is valuable to help gain confidence in the system. User training and acceptance testing activities should be performed to apply a base level loading on the system. During these periods, the following types of tests can be performed while concurrently monitoring system behavior and performance.

- **Large Scale Operations** – Even under nominal loading conditions, be sure to test any large scale use cases that will be performed in production. Large CAD operations, batch viewable creation, report generation, advanced searches, and releases to ERP are a few examples to consider.
- **Extreme Use Cases** – There are also times when users can initiate inadvertent or unintended actions that can create larger than normal system loading. These scenarios give rise to what is called negative testing and should be anticipated wherever possible.
- **Local and Remote Sites** – Testing from both LAN and WAN environments is also significant when confirming the scalability of specific operations, independent of geographical location. In such instances, network monitoring tools can also help to ensure the test environment conditions and can assist in troubleshooting potential problems.

User Acceptance Tests

In order to ensure a successful production rollout process, user acceptance testing (UAT) cycles should be planned. During these test periods, teams of future users are assigned a combination of formal tasks and free time with the system. User results and feedback are collected and acted upon as necessary.

User acceptance testing is generally most successful when key stakeholders (end users) take part in defining the use cases and release criterion early in the validation planning process. Defining clear and unambiguous criterion up front eliminates subjectivity and emotions from the release decision process.

Whenever possible, user acceptance test cases should be pre-run in earlier test phases to ensure success and a good first impression when users are formally exposed to the system.

Other Tests

Methods to test additional architectural features should also be considered if they are applicable to your configuration.

- Security
- Replication
- Backup/Recovery
- Failover
- Decommissioning

Governing

A formal governance process should be used to help monitor and control the system validation plan activities. A reporting schedule for the review team and business stakeholders should be maintained to ultimately oversee the progress in achieving the go-live release criterion.

Example criterion defined in the System Validation Plan can include;

| “Go-live” Criterion Description | Current Status | “Go-live” Criteria |
|--|----------------|--------------------|
| # of Open “Blocking” Issues (P0) | | 0 |
| # of Open “High Priority” Issues (P1) | | <10 |
| % of Data Integrity Tests Completed (passing) | | 100% |
| % of Functional Tests* Completed (passing) | | 100% |
| % of Solution Tests* Completed (passing) | | 100% |
| % of Performance Tests* Completed (passing) | | 100% |
| % Scalability Test* Completed (passing) | | 100% |
| % Documented Installation / Configuration Baseline | | 100% |
| % User Acceptance Tests (UAT) Completed (passing) | | 100% |
| % Other Criterion* (passing) | | 100% |

*Those identified as mission critical scenarios and use cases

In support of the review and reporting activity, detailed test case results, issues lists and summary reports can be archived in a test case management tool or in Windchill ProjectLink. Formal management and control of this information can help ensure that proper tracking, notification, traceability and approval processes are achieved.

Throughout the system validation process and into production, it is important to report and manage risks that threaten the quality, content and schedule of the implementation. For this reason, it is important to hold these regularly scheduled reviews to identify and mitigate risks leading up to the successful release and during the production use of the Windchill solution.

Maintaining

The test system and environment remains a critical asset to leverage even after the Production system has been deployed. It is important to plan, deploy and test any changes (configuration or code) on the test system before applying them to the production environment. Again, the Configuration Definition and Change Log (or similar) should be used to formally document any system changes.

Finally, production system monitoring should also include taking periodic performance measurements, at the various locations, using the tests that were used to initially qualify the system. Taking these proactive steps will help ensure that the system as qualified when released will remain responsive and productive as the system data characteristics and usage evolves.

References

The following table identifies additional materials that may be of interest:

[Configuration Definition and Change Log Template](#)

[Hardware Sizing Guidelines - HP-UX Platform](#)

[Hardware Sizing Guidelines - IBM AIX Platform](#)

[Hardware Sizing Guidelines - Linux Platform with Oracle Database](#)

[Hardware Sizing Guidelines - Microsoft Windows Platform with Oracle Database](#)

[Hardware Sizing Guidelines - Microsoft Windows Platform with SQL Server Database](#)

[Hardware Sizing Guidelines - Sun Solaris Platform](#)

[Windchill Deployment Planning Checklist](#)

[Windchill Creo Data Management Performance Benchmark](#)

[Windchill Single User Performance Tester for Creo Data Management Operations](#)

[Using the PTC Day-in-the-Life Performance Benchmark](#)

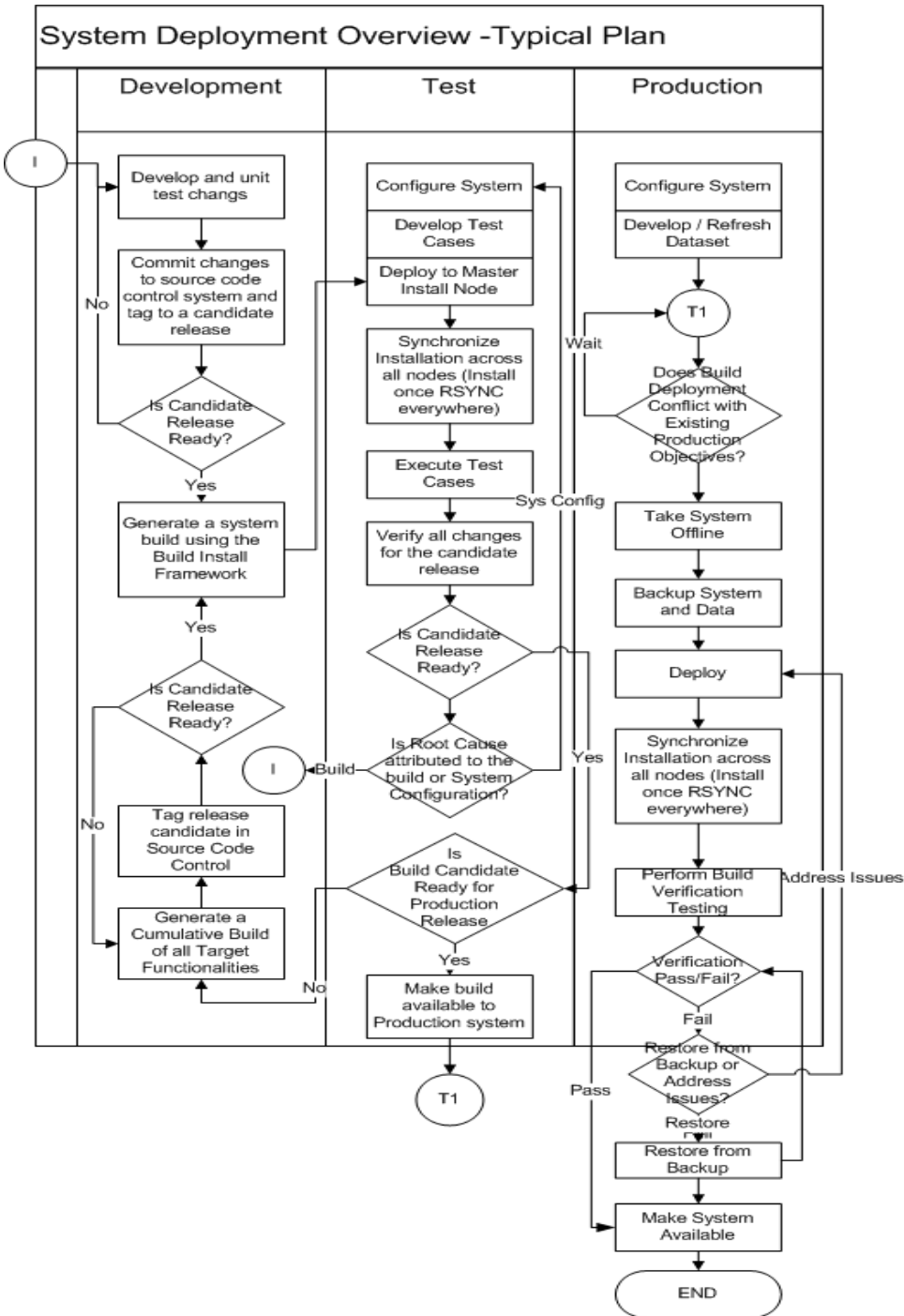
Reference documents can be found on the PTC Technical Support Web site at:

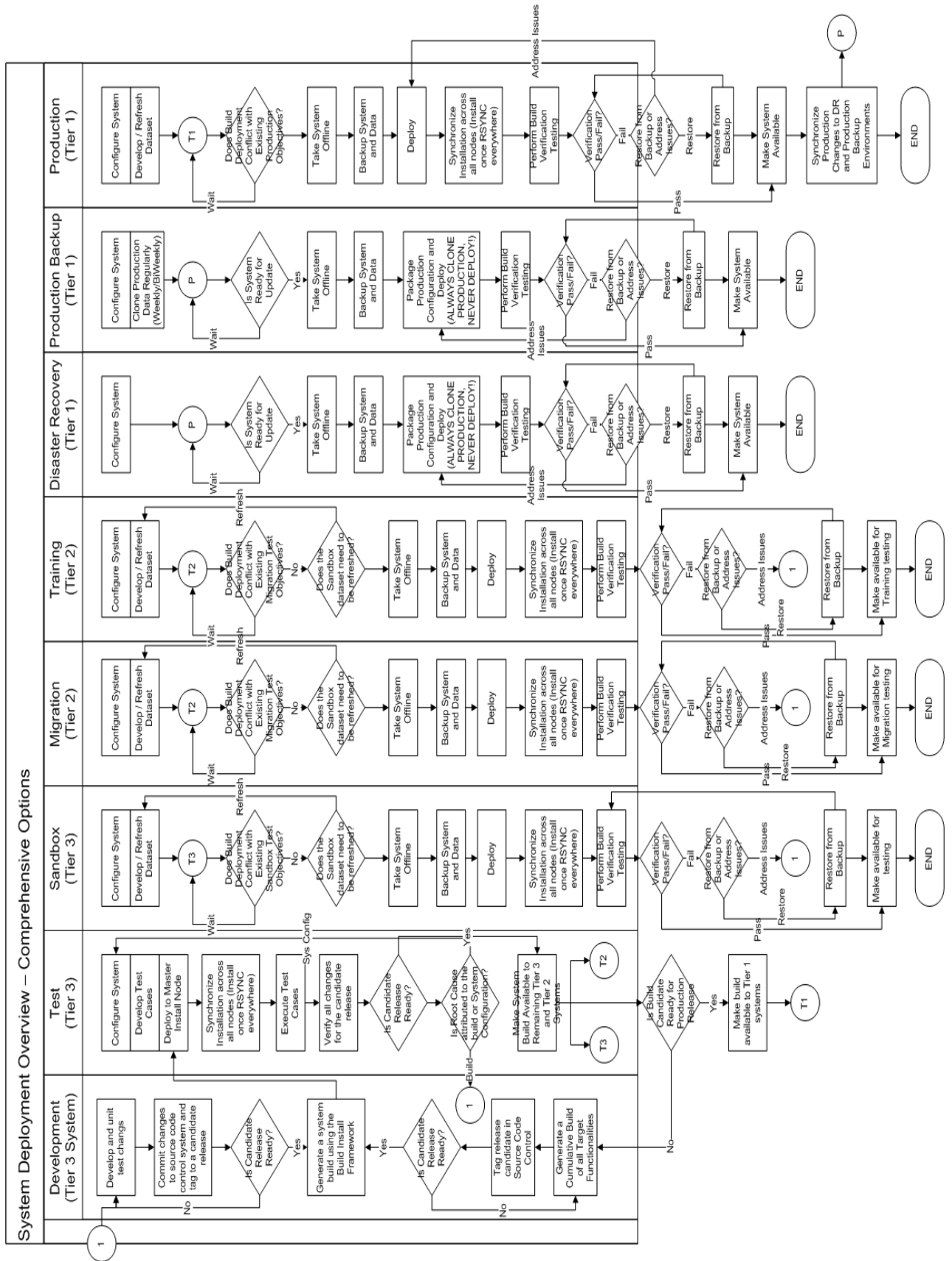
<http://www.ptc.com/appserver/cs/doc/refdoc.jsp>

To access the Reference Documents site, you must have a valid PTC user name and password. For active maintenance customers, use your Service Contract Number (SCN). If you do not have an SCN, contact the PTC Maintenance Department using the instructions found in your PTC Customer Service Guide. For prospective customers, work with a PTC sales representative to get the information you need.

After you have logged on, you can search the site by selecting a Windchill product, release, and document type.

Deployment Diagrams





©2011 Parametric Technology Corporation (PTC). The information contained herein is provided for informational use and is subject to change without notice. The only warranties for PTC products and services are set forth in the express warranty statements accompanying such products and services and nothing herein should be construed as constituting an additional warranty. PTC shall not be liable for technical or editorial errors or omissions contained herein. For important Copyright, Trademark, Patent, Licensing and Data Collection Information see the About Box, or Copyright Notice, of your PTC software.