

SIEMENS TEAMCENTER UPGRADE

Authored by Saket Mohorir



Table of Contents

01

Introduction

01

Overview of PLM
Upgrade

02

Key Considerations
While Planning
Teamcenter Upgrade

02

High-level
Procedures

04

Teamcenter
Upgrade Project
Execution

05

Teamcenter Upgrade
Methodology

06

Upgrading More
than Two Versions

07

Essential
Considerations in
Teamcenter Upgrade
Testing

08

Conclusion

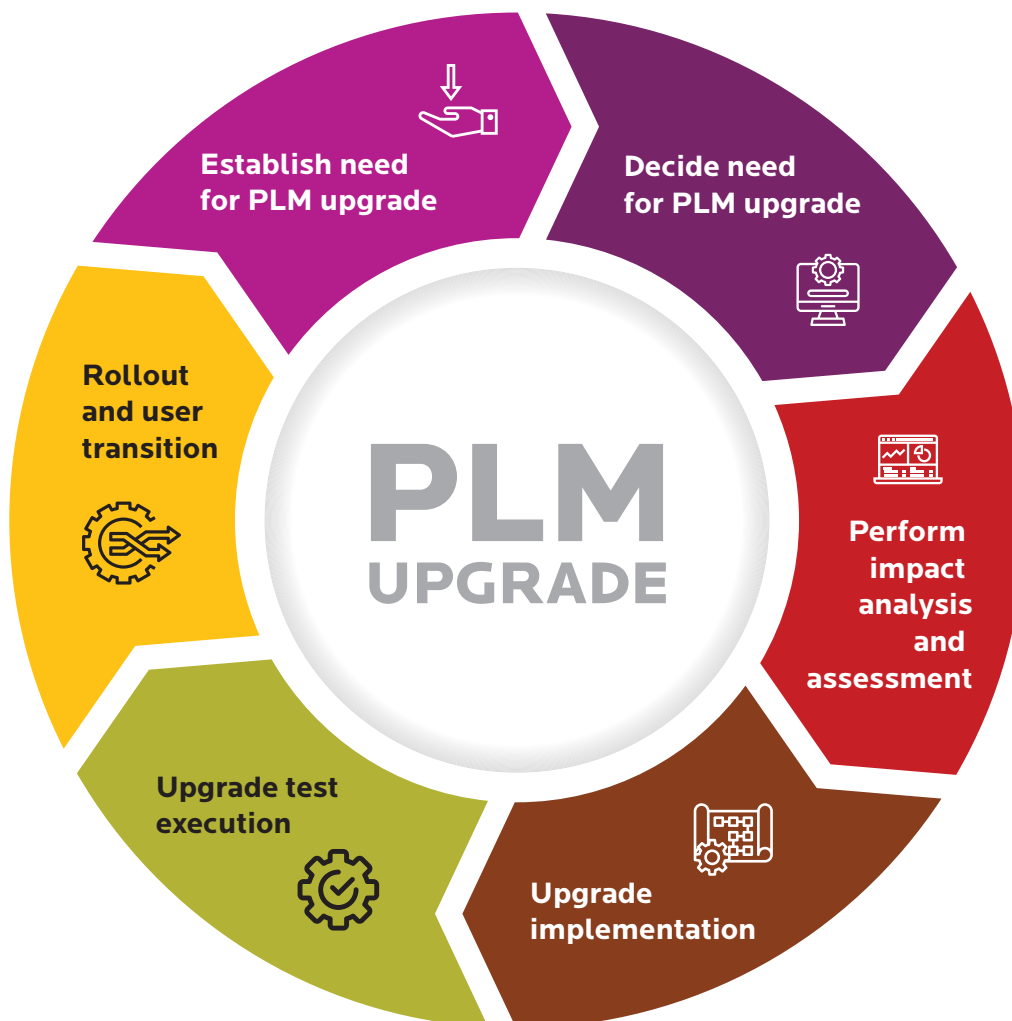
Introduction

There is an old saying, “The only thing constant in life is change,” and this applies to everything including the Siemens Teamcenter PLM. There are changes all over the world that are affecting the way organizations collaborate and develop products.

PLM systems are getting mature to add more functionalities and new business process under its umbrella, however the technical and practical complexities to handle the change remains challenge. Upgrading the Teamcenter PLM environment is always a challenge with all the integrations and new changes coming in new versions or the ecosystem. Surprises and delays can happen due to unexpected errors and findings.

Overview of PLM Upgrade

The ladder chart below shows a high-level process to describe the flow of a typical Teamcenter upgrade. It by and large applies to all the popular PLM tools currently available in the market, though there could be variations sought for specific technical aspects that need to be tailored for a particular PLM product.



Key Considerations While Planning Teamcenter Upgrade

Typically, PLM upgrades are under-estimated due to inaccuracies and limited availability of data points, critical for planning. This leads to surprises during upgrade execution resulting in time and budget overruns. A careful evaluation of your environment is needed to plan and design your upgrade execution.

Upgrade Path

Multiple options varying in degree of functional changes adopted during the upgrade are available. These range from shortest and least disruptive like-to-like upgrade to transformational upgrade involving large scale changes to an existing solution. Most organizations conduct like-to-like upgrades and adopt innovations in an incremental manner subsequently through smaller, more focused projects.

Upgrade Scope

Mostly, application upgrades also involve tech-stack upgrade, infrastructure refresh, or even re-platforming and upgrade of interfacing third party systems as well. These activities, along with current database size, the volume of retrofitted customizations also determine the downtime needed during the final production upgrade.

Code Retrofit

With every new major release of Teamcenter/ Active workspace there is an architectural or functional changes which can potentially impact the customizations deployed. The break-fix approach of fixing only what is broken is reactive and a more scientific approach of impact assessment to drive project planning is recommended.

Testing & User Adoption

Upgrade projects involve rigorous test iterations to ensure that nothing is broken in the upgraded environment. For a like to like upgrades, the majority of this testing is conducted by the IT team. There are additional considerations around delta training to business users as well.

High-level Procedures

Most enterprises struggle to make the right decision about when to upgrade and what level of upgrade is necessary. The biggest challenges identified in an upgrade project are timing, balancing the execution of the upgrade project with other concurrent projects, and lack of knowledge regarding release specific functional enhancements. Also, validate the versions of all integrated and dependent software like CAD, ERP, Database, MS-Office, etc. to see that they remain supported for next 2-3 years on new version.

The certification matrix can be downloaded from Siemens website here

<https://www.plm.automation.siemens.com/global/en/support/certifications.html>

PRE-UPGRADE

- Assess key characteristics of the client Teamcenter environment- Hardware, software, disk space, key system dependencies, etc.
- Recommend supporting system upgrades or enhancements as necessary to support productivity improvements with the new software version
- Design a Teamcenter production clone environment to act as the TC12 installation and test bed for the software upgrade
- Document any system changes or fixed variables required for the target production environment
- Establish expected performance benchmarks and load processing capabilities for the upgraded software
- Develop and validate an Upgrade project plan, procedures, and checklist compatible with your needs
- Set up a Test system for use as a Test source target system in support of the upgrade process



CORPORATE SERVER UPGRADE

- Construct and validate the clone system according to the findings and agreements set forth in the pre-planning phase
- Install, configure, and validate the new software version and/or modules
- Configure the test-target system to process test-source data
- Run the Upgrade Manager and Teamcenter utilities using the test system and database to evaluate configuration goodness of fit
- Validate pre-upgrade process on test source and target system
- Adjust test system as necessary
- Prepare and configure the production system and environment
- Perform the upgrade according to the plans developed and tested
- Run the upgrade manager using the production system
- Validate and tune the upgraded software



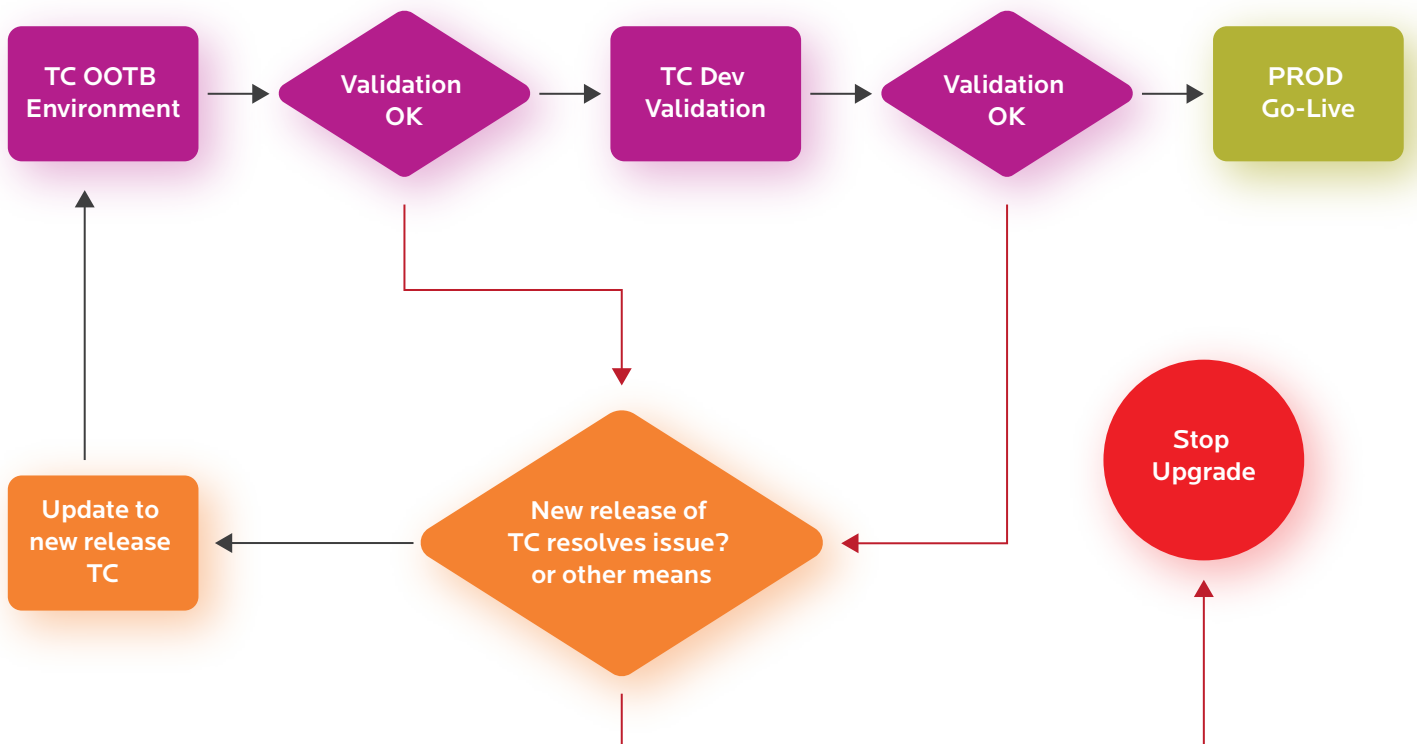
POST-UPGRADE

- To ensure production software functionality and performance, we will provide system and environment trouble-ticket and helpdesk support as necessary for a period of five days beyond the actual go live date for the upgraded software in the production environment
- Trouble ticket and helpdesk support required beyond the initial five day period will be charged on a per-ticket basis at an hourly rate or we can work out a contract that covers a specific period of time for you

Teamcenter Upgrade Project Execution

The execution of Teamcenter upgrade project is little different from any other software development projects as it impacts the entire ecosystem within the organization. Teamcenter upgrade projects, depending on the scope and strategy, could bring about significant impact on the underlying data. The upgrade project has to ensure that code / configuration merging is done properly, to ensure that none of the current functionalities of the deployed system, or from the new release gets inadvertently overridden. Another challenge is that during the length of upgrade project, production fixes or some other enhancements requests could run in parallel. If any organization opts for a comprehensive upgrade, then configuration and build management becomes most critical, since the team would need to develop rules and guidelines on how different teams will manage configurations, and resolve the conflicts.

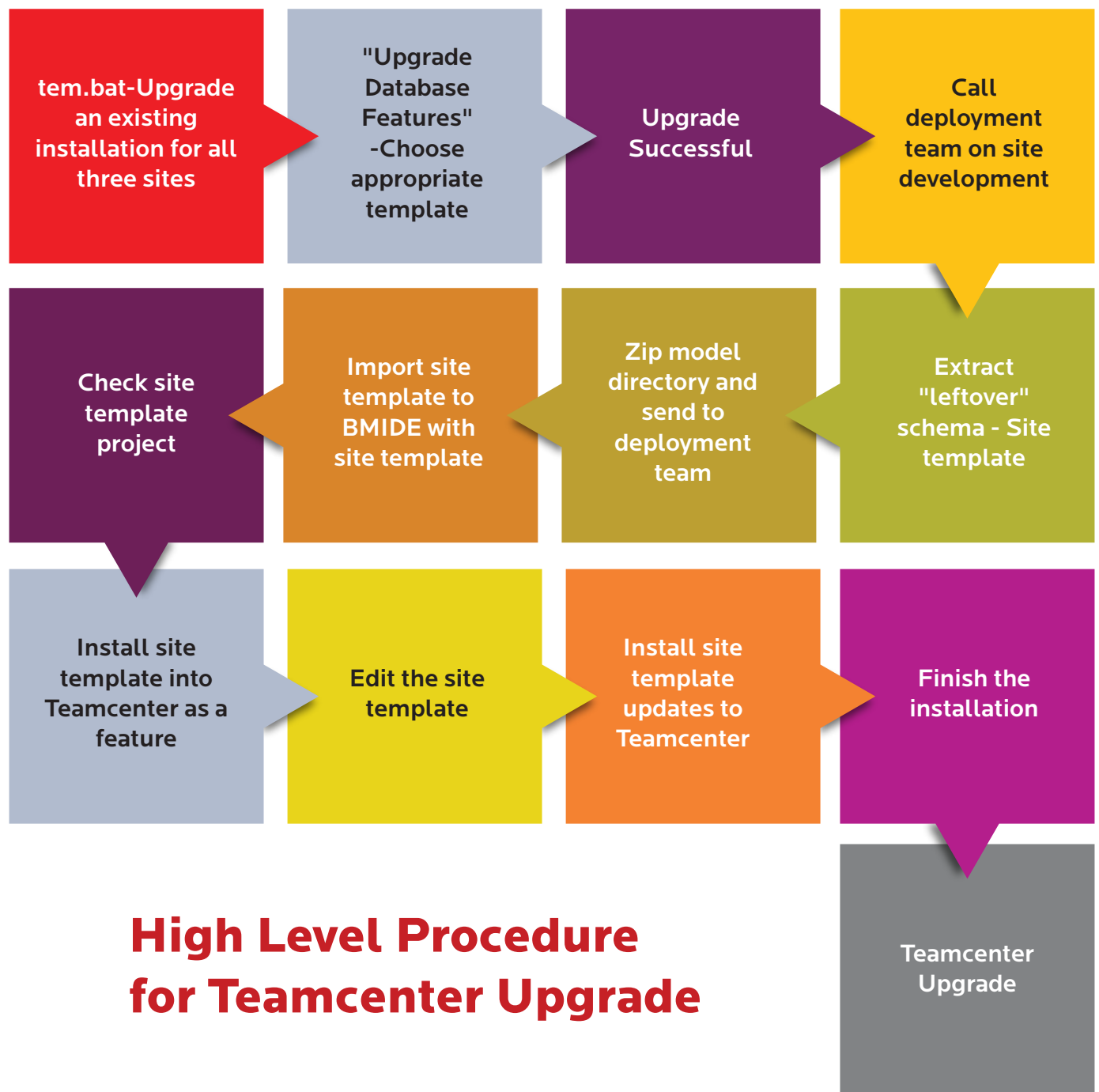
Following chart depicts the high-level flow for a typical 1 step upgrade process which can be further scalable to a multi steps procedures



High Level Flow for a TC Upgrade

Teamcenter Upgrade Methodology

Below figure depicts the procedure for a typical Teamcenter upgrade, however it may vary from instance to instance and organization to organization.

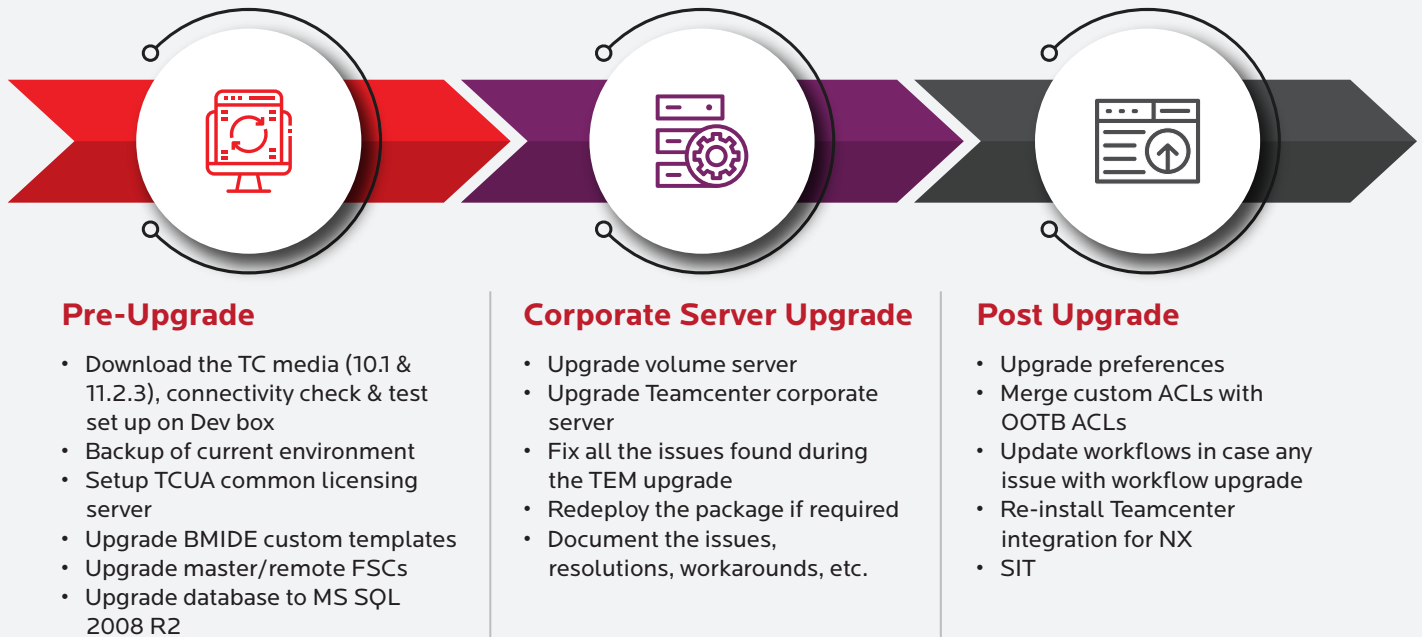


Upgrading More than Two Versions

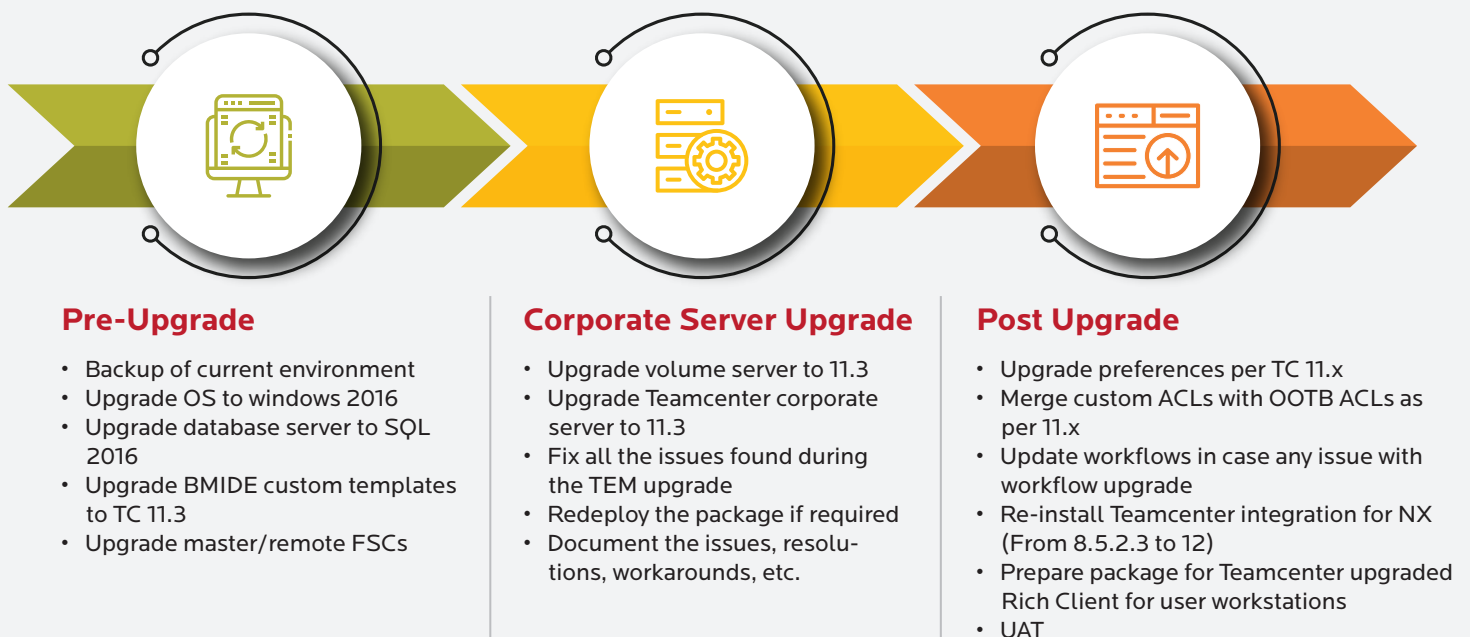
.tem upgrade does not support jump over more than two versions, but it is possible to do it internally. For example, an upgrade from TC8 to TC12 would look like upgrade from TC8 to TC10 and then TC10 to TC12. Process can be done internally without users getting to see intermediate versions. Some extra efforts will be there for two internal upgrades but better than upgrade twice with interval.

Wave 1(TC 8.3 to TC 10.1)

Compatible OS for both TC 8.3 & 10.0 is Windows Server 2008 R2

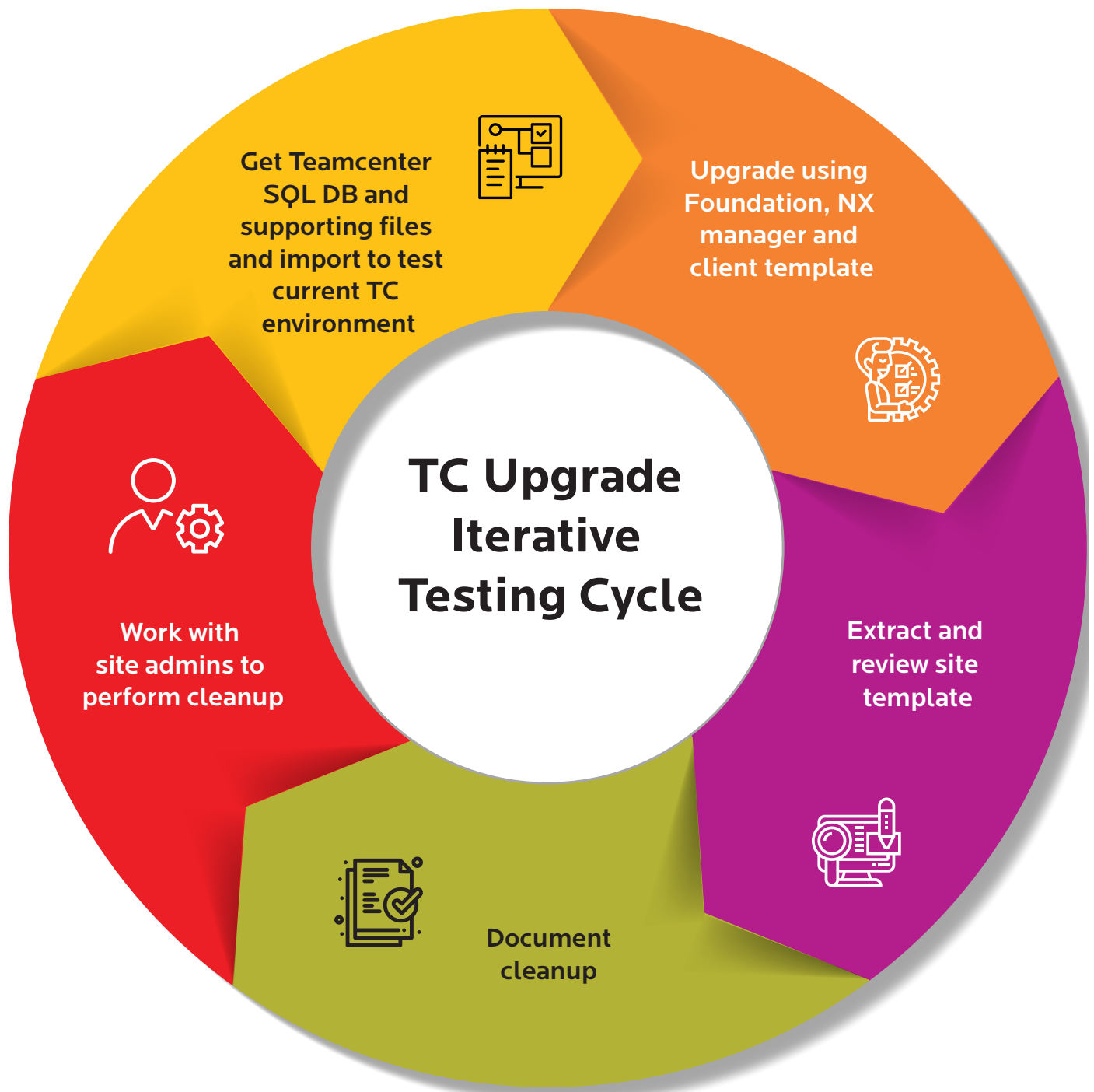


Wave 2 (TC 10.1 to TC 11.3)



Essential Considerations in Teamcenter Upgrade Testing

There are multiple changes involved like changes to code, schema and data. The impact of upgrade project can be huge from application usage and data perspective. Hence, systematic planning and execution of test plans is a very important phase of the project.



Conclusion

Establishing the need for an upgrade, followed by detailed assessment of process and technology components will ensure PLM upgrade objectives and consequences are determined, well defined process for handling data in PLM upgrade will make sure that data integrity is not compromised at any point of time in the process. Comprehensive testing methodology that includes function, regression, validation and user acceptance tests ensures PLM upgrade results in sustainable and stable rollout.



RESOURCES

plm@birlasoft.com | birlasoft.com

Enterprise to the Power of Digital™

Birlasoft combines the power of domain, enterprise and digital technologies to reimagine business processes for customers and their ecosystem. Its consultative and design thinking approach makes societies more productive by helping customers run businesses. As part of the multibillion-dollar diversified The CK Birla Group, Birlasoft with its 10,000 engineers, is committed to continuing our 158-year heritage of building sustainable communities.