

The Siemens logo is displayed in a white rectangular box in the upper left corner. The word "SIEMENS" is written in a bold, teal, sans-serif font. The background of the entire page is a low-angle photograph of a modern glass skyscraper reaching towards a blue sky with scattered white clouds. The glass facade of the building reflects the sky and clouds, creating a sense of height and modernity.

**SIEMENS**

# Active Workspace 5.2

## Active Workspace Installation

AW001 - 5.2

# Contents

## Where to start

Active Workspace deployment guidemap	1-1
System requirements	1-4
Choose a documentation source	1-7
Choose the correct Active Workspace deployment process	1-8

## Installing Teamcenter microservices

Microservices and the microservice framework	2-1
Microservices and Docker on Linux hosts	2-3
Microservices and Kubernetes on Linux hosts	2-6
Install microservices using TEM	2-7
Install microservices using TEM- Docker on Linux host	2-7
Install microservices using TEM- Windows host	2-10
Add microservice framework support to a Java EE web tier	2-15
Add microservice framework support to a .NET web tier	2-18
Install microservices using Deployment Center	2-20
Install microservices on a Linux host via Deployment Center	2-20
Install microservices on a Windows host via Deployment Center	2-23
Add microservices and microservice nodes for Docker on a Linux host	2-26
Add microservices and microservice nodes on a Windows host	2-26
Starting microservices in Docker	2-27
Join a node to a running Docker swarm	2-28
Starting microservices in Kubernetes	2-29
Manage microservice instances on Windows hosts	2-30
Finding microservice logs on Windows hosts	2-31
Redirect microservice logs on Windows hosts	2-32
Configure HTTPS for microservices	2-33
Configure service dispatcher as an HTTPS server	2-33
PKI key requirements for microservices	2-41
High availability for microservices	2-42
Troubleshooting microservices	2-46

## Installing Active Workspace

Active Workspace components installation overview	3-1
Active Workspace Server Extensions	3-2
Server Extensions overview	3-2
Install Server Extensions features	3-4
Install database triggers manually	3-5
Active Workspace client components	3-8
Active Workspace client overview	3-8
Install Active Workspace microservices	3-9
Install Active Workspace Gateway	3-10

Install the Active Workspace client	3-17
Sign in using the Active Workspace interface	3-20
<b>Indexing</b>	3-21
Prepare to install indexing	3-21
Configure Classification Search (optional)	3-21
<b>Install Dispatcher</b>	3-22
<b>Visualization Server</b>	3-23
Visualization Server overview	3-23
Should I use client-side or server-side rendering?	3-24
Should I use MMV?	3-25
Visualization Server Manager	3-26
Visualization Server Pool Assigner	3-36
Visualization Data Server (optional)	3-41
<b>Install the Active Workspace Launcher</b>	3-49
<b>Install classification libraries</b>	3-50
<b>Verify Active Workspace installation</b>	3-52

## Updating Active Workspace and microservices

<b>Patch Active Workspace 4.3 or later</b>	4-1
<b>Patch Active Workspace 4.2 or earlier</b>	4-2
<b>Complete the Active Workspace update</b>	4-4
Remerge Solr and Teamcenter schemas and update the index	4-4
Additional postupdate steps	4-5
<b>Publishing changes to the Active Workspace file repository</b>	4-7

## Active Workspace features reference

<b>Active Workspace Server Extensions features</b>	A-1
<b>Active Workspace Client features</b>	A-21
<b>Active Workspace microservices features</b>	A-32

## Load balancing Active Workspace components B-1

## Required RPM package managers C-1

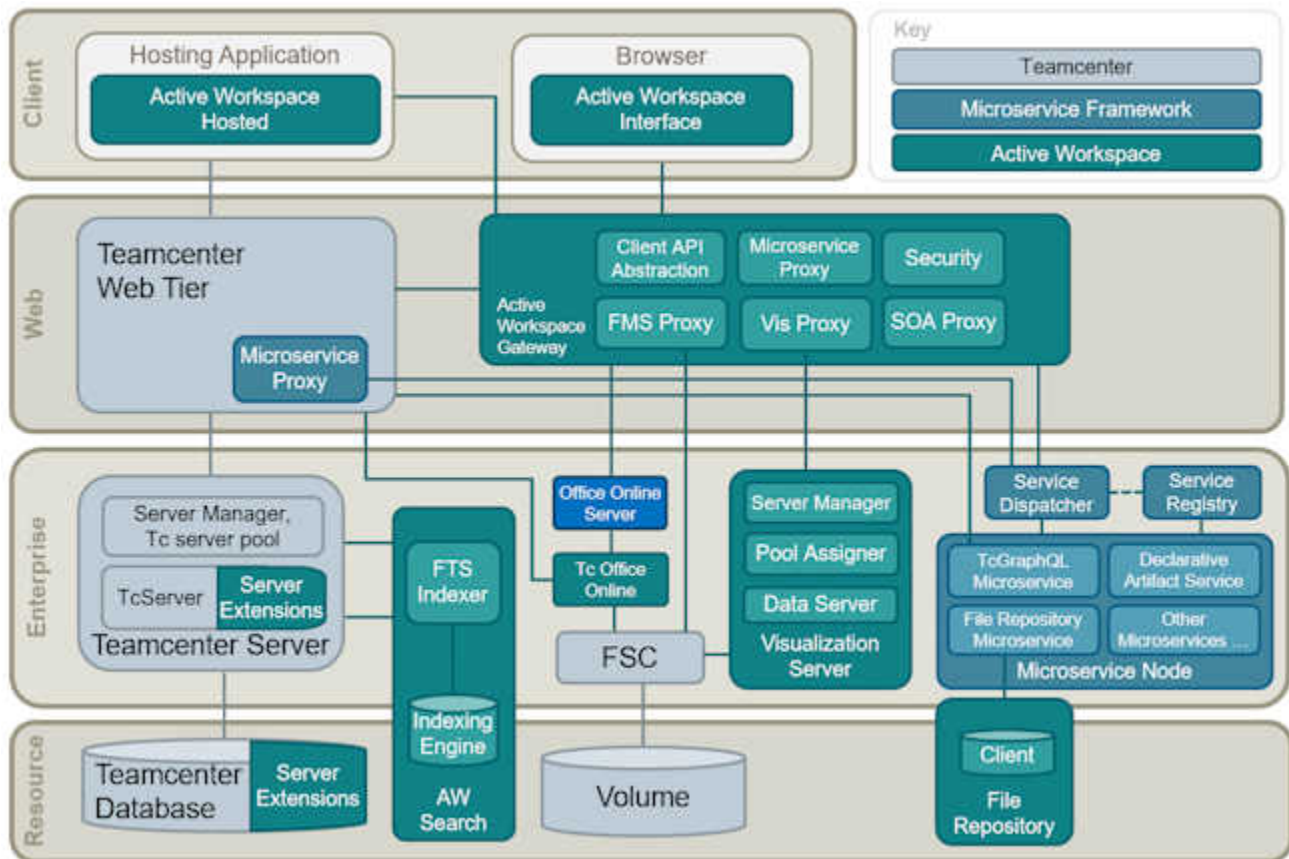




# 1. Where to start

## Active Workspace deployment guidemap

Active Workspace components leverage Teamcenter server and client functionality, adding enhanced indexing and visualization. The Active Workspace architecture and microservice framework integrate with the Teamcenter four-tier architecture as shown.<sup>1</sup>



An Active Workspace deployment begins with a Teamcenter environment to which you add the microservices framework and then Active Workspace components. You can deploy all components on a single machine or on multiple machines according to your business needs.

### Get Active Workspace deployment help

These resources provide essential guidance for planning, deploying, and administering your Active Workspace environment.

- **Active Workspace documentation**

<sup>1</sup> This illustration also shows the Microsoft Office Online Server with the Teamcenter Office Online integration.

**Choose an online help source** from which to access the complete library of Active Workspace documentation.

- **Deployment Reference Architecture**

This guide provides architecture examples and recommended approaches for deploying Teamcenter and Active Workspace.

The Deployment Reference Architecture is available from the Teamcenter downloads page.

### Download the required software

Active Workspace 5.2 supports multiple versions of Teamcenter. From the Teamcenter downloads page on Support Center, download the Active Workspace software kit that corresponds to your major version of Teamcenter and your platform type:

**Tcversion.0.0\_ActiveWorkspace5.2.0\_platform.zip**

From the Teamcenter downloads page, select **Additional Downloads**→**Microservice Framework**, and then download the microservice framework 5.2 software kit for your platform type. This kit is required for Active Workspace installation.

**TcMicroserviceFramework5.2.0\_platform.zip**

If a later software patch kit is available for Active Workspace or microservice framework, for example, version 5.2.1, you can download that kit instead.

Expand the Active Workspace and microservice framework software kits to separate directories that are accessible to your Active Workspace host. If you use Deployment Center, add these kits to your software repository.

### Choose the right Active Workspace installation process

Active Workspace architecture changed in Active Workspace 4.3, replacing the Active Workspace WAR file and .NET client with a system of microservices and an Active Workspace Gateway. The process you follow to install or update to Active Workspace 5.2 depends on your current version of Active Workspace.

Make sure your designated machines for each component meet **system requirements**. Then, **choose the right deployment process** for your Active Workspace environment.

### Learn about Active Workspace microservices

Core components of Active Workspace require the microservice framework and **Teamcenter microservices**. The file repository and its related components provide secure file storage for Active Workspace. Other microservices support certain Active Workspace components and applications.

- **File Repository**

The File Repository is a centralized storage location for Active Workspace client content that is managed by the File Repository Microservice and accessed through the Active Workspace Gateway. The File Repository is used primarily for temporary file storage and will be made available to other microservices as a place to upload, download, and keep files. The File Repository provides microservices an alternative to File Management System (FMS) or the Teamcenter server for temporary file storage needs.

- **Active Workspace Gateway**

The Active Workspace Gateway functions as a web server for the Active Workspace browser interface. It uses the File Repository to store its static and dynamic assets. The Gateway also communicates with microservices, the Teamcenter server, and the Volume server, as well as the File Repository. The Active Workspace Gateway is implemented in Node.js and routes HTTP requests for static or dynamic content to the appropriate services handling those requests. The Active Workspace Gateway and microservices should be **deployed in a load-balanced environment** to manage network resource demands.

- **Specialized microservices**

Some Active Workspace applications require certain **associated microservices**.

## Plan your distribution of Active Workspace components

Active Workspace components extend Teamcenter functionality. You can deploy all components on a single machine or on multiple machines according to your business needs. Plan your distribution of Active Workspace components to maximize availability and performance. **Load balance** components for failover support.

- **Server extensions**

Active Workspace **Server Extensions** add Active Workspace support to the Teamcenter server. You install server extension features on each Teamcenter server where **Teamcenter Foundation** is installed. This includes corporate servers, volume servers, and server manager hosts. Installing these features adds binaries to *TC\_ROOT* and data model changes to the database and *TC\_DATA*.

- **Index and search**

The Indexer and the Indexing Engine provide global search capabilities for the Active Workspace client.

- **Indexing Engine**

Installs the Solr enterprise search platform. The search engine stores indexed Teamcenter data for global search in Active Workspace.

- **Indexer**

Installs a four-tier SOA client that exports Teamcenter data for merging into Solr. The Indexer (called TcFTSIndexer) manages overall indexing processes. The TcFTSIndexer manages initial indexing for object data. You can then schedule synchronization to run periodically for subsequent updates to object data or structure data indexes.

There are two modes for installing the **Indexer**, standalone for object and structure data and Dispatcher for asynchronous job processing and translator conversions.

- **Visualization Server**

The Visualization Server provides dynamic 3D visualization to the Active Workspace client. The Visualization Server is optional. If you plan to use Visualization with Active Workspace, **install Visualization Server**.

The Visualization Server components are:

- **Visualization Server Manager**

The Visualization Server Manager streams visualization data to the Active Workspace client. It manages a pool of rendering processes, starting and stopping processes as needed.

- **Visualization Server Pool Assigner**

The Visualization Server Pool Assigner routes visualization processes to an available Visualization Server Manager.

- **Visualization Data Server** (optional)

The Visualization Data Server improves Visualization performance by caching visualization data close to the Visualization Server Manager. It caches product structure, computes Massive Model Visualization (MMV) spatial hierarchies, and prepopulates JT files in the File Client Cache (FCC) for faster visualization streaming to the Active Workspace client. You can use the Visualization Server Manager without this component.

A single Visualization Data Server can support one or more Visualization Server Managers. A Visualization Server Manager must be installed on the same host as the Visualization Data Server. The Visualization Data Server is required for using MMV feature in Active Workspace. Additionally, you need to index structure data for the product configurations that you want to view using MMV.

## System requirements

Versions of system software and hardware certified for Teamcenter and Active Workspace are described in the Hardware and Software Certifications knowledge base article on Support Center.

Active Workspace software components require the following system software and hardware:

- **Active Workspace Server Extensions**

- 64-bit operating system and 64-bit JDK on the corporate server host.
- Teamcenter web tier application (Java EE or .NET) deployed on the web tier host.

- **Active Workspace Client**

- A minimum of 1 GB of free physical RAM (unused by other applications) to avoid paging. Siemens Digital Industries Software recommends that the machine used have a minimum of 2 GB.



- On Linux hosts, Active Workspace Gateway requires Node.js.  
For certified versions of Node.js, see the Hardware and Software Certifications knowledge base article on Support Center.
- On Linux hosts, **microservice framework requires Docker**.  
Microservice framework is required by Active Workspace Gateway. If you install microservice nodes on a Linux host, make sure Docker is installed on the host.  
For certified versions of Docker, see the Hardware and Software Certifications knowledge base article on Support Center.
- 64-bit Java development kit (JDK) installed and the **JAVA\_HOME** system environment variable set.

- **Indexing Engine**

- 64-bit operating system.
- 64-bit Java development kit (JDK) or Java runtime environment (JRE).

- **Visualization Server**

- **Visualization Server Manager**

- The Visualization Server Manager requires the following software and hardware:

- ◇ **Operating system**

Supported Microsoft Windows server or Linux server platform. For supported versions, see the Hardware and Software Certifications knowledge base article on Support Center.

- ◇ **Graphics cards**

Server class hardware using NVIDIA GRID K1, K2, Tesla M60, P40, T4, RTX 6000, or RTX 8000 graphics cards as certified by NVIDIA.

- ◇ **GPU**

Windows Server 2012 R2 supports a maximum of 8 GPUs.

For information about NVIDIA server hardware compatible with the GRID graphics cards, see [www.nvidia.com](http://www.nvidia.com).

Supported hardware is suitable for a production environment. Other hardware may work with Active Workspace, but Siemens Digital Industries Software accepts problem reports only for issues that are reproducible on a supported configuration.

For example, when booting Windows Server 2012 R2 with a legacy VGA device and two or more NVIDIA GRID K1 cards, one of the NVIDIA GPUs is unavailable for use.

- **Visualization Data Server (optional)**

You can install the Visualization Data Server (VDS) component to improve visualization performance. The Visualization Data Server uses the structure indexing infrastructure of Active Workspace to keep cached product structure up-to-date.

The Visualization Data Server is required for Massive Model Visualization (MMV). Structure indexing must be set up for MMV.

- Supported Microsoft Windows server platform or supported Linux server platform. For supported versions, see the Hardware and Software Certifications knowledge base article on Support Center.
- The machine hosting the Visualization Data Server should have a minimum of 16 GB of RAM but may require more. The amount of RAM needed depends on the number of structures to be indexed and their size.  
A rough rule of thumb is to count the number of lines in the unconfigured structure to be indexed and allow at least 2000 bytes per line. For example, if there are 1 million lines in the unconfigured product index,  $1 \text{ million} * 2000 = 2 \text{ GB}$  of RAM.  
If you are not sure of the size of the structures, Siemens Digital Industries Software recommends that you allow approximately 4 GB of RAM for each structure you are planning to cache in the Visualization Data Server. For example, if 4 structures are to be indexed, 16 GB of RAM is required.
- Siemens Digital Industries Software recommends that you install the Visualization Data Server on a machine with multiple processors.  
The Visualization Data Server is a multithreaded server program and is therefore resource intensive; multiple processors are utilized if they are available. Standard server class machine hardware is sufficient.
- There are no graphics card requirements.
- You must deploy the Visualization Data Server on a high-speed LAN near the Visualization Server Manager.
- Siemens Digital Industries Software recommends that you deploy the Visualization Data Server near or on a machine hosting an FSC cache or FSC volume. If you deploy the Visualization Data Server on a remote machine (on a WAN) from the FSC volume, you should deploy an FSC cache on a LAN near or on the Visualization Data Server host machine.
- You must deploy an FMS client cache (FCC) component on the machine hosting the Visualization Data server.  
For maximum performance, the Visualization Data server should be installed on the same machine as the Visualization Server Manager, and should use the same cache.

## Choose a documentation source

Active Workspace documentation is available from two sources:

- **Support Center**

Siemens Digital Industries Software's comprehensive support portal, which provides documentation for all Siemens software products and versions.

The *Siemens Secure Documentation Proxy* provides secure, easy access to documentation without a need to log on, through a personalized API key specific to your site.

- **Siemens Documentation Server**

Locally-installed server that can host documentation for all your Siemens Digital Industries Software products. No Internet access is required. You can configure the server for single-machine or network-wide access.

The Siemens Secure Documentation Proxy and the Siemens Documentation Server are available on Support Center under **Products**→**Siemens Documentation Installer**→**Downloads**.

Choose how you want to access documentation, then download and install the required software:

Installing Siemens Secure Documentation Proxy	Installing Siemens Documentation Server
Install the Siemens Secure Documentation Proxy to access documentation on Support Center.	<ol style="list-style-type: none"> <li>1. Install the Siemens Documentation Server.</li> <li>2. Install documentation for your software products. Each <i>publications kit</i> contains documentation content and an installation wizard that automatically installs documentation onto your Documentation Server. Publications kits (<b>docs-product-name-and-version-locale.zip</b>) are available from the <b>Downloads</b> pages for each product on Support Center.</li> </ol>

Complete installation instructions for the Secure Documentation Proxy and the Siemens Documentation Server are available on Support Center.

For a step-by-step orientation to Support Center, see Siemens Software [Support Center videos](#) on YouTube.

### Enable Active Workspace access to documentation

If you use Active Workspace documentation on Support Center, no further configuration is necessary. The Active Workspace **Help** button links to Support Center by default.<sup>1</sup>

If you install the Siemens Documentation Server on your local network, configure the **Help** button to link to Active Workspace documentation on your local server:

1. In Active Workspace, open Command Builder.
2. Find the **showHelp** action and set its **Navigate To** property to the URL to Active Workspace documentation on your local documentation server:

**`http://host:port/en-US/product/282219420/doc/PL20201019171517939.xid1899337/html/xid1899339`**

Replace *host* and *port* with the host name and port of your local documentation server.

3. Commit your UI Builder changes to your module to update the **Help** button link for your users.

## Choose the correct Active Workspace deployment process

The process you follow to install or update Active Workspace depends on your existing Active Workspace environment.

### Active Workspace 4.3 or later

If you are running Active Workspace 4.3 or later, your environment includes the microservice framework. To update to Active Workspace 5.2, you update the microservice framework and Active Workspace:

1. Update your Teamcenter environment (patch or upgrade) to a version supported for Active Workspace 5.2.

Make sure you update every server in your environment.

2. **Patch microservice framework and Active Workspace.**

Make sure you patch every server in your environment.

On Linux microservice nodes, to ensure new Docker images are used, **redploy them** using the updated **.yml** files.

3. **Update the Active Workspace environment.**
4. **Publish changes to the file repository.**

---

<sup>1</sup> The default URL to Active Workspace documentation on Support Center is <http://docs.sw.siemens.com/en-US/product/282219420/doc/PL20201019171517939.xid1899337/html/xid1899339>.

## Active Workspace 4.2 or earlier

If you are running Active Workspace 4.2 or earlier, you must install microservice framework before you patch Active Workspace to Active Workspace 5.2:

1. Update your Teamcenter environment (patch or upgrade) to a version supported for Active Workspace 5.2.

Make sure you update every server in your environment.

2. **Install and configure the microservice framework.** This includes:

- Installing the microservice framework.
- Installing the microservice proxy (microservice support for the web tier).
- Starting the microservice node and any additional configuration for your platform type.

3. **Patch Active Workspace.**

Make sure you patch every server in your environment.

4. **Update the Active Workspace environment.**

5. **Publish changes to the file repository.**

## Teamcenter environment without Active Workspace

If you are adding Active Workspace to an existing Teamcenter environment, you must install the microservice framework before you install Active Workspace:

1. Update your Teamcenter environment (patch or upgrade) to a version supported for Active Workspace 5.2.

Make sure you update every server in your environment.

For supported combinations of Teamcenter or Teamcenter Rapid Start with Active Workspace, see the Hardware and Software Certifications knowledge base article on Support Center.

2. **Install and configure the microservice framework.** This includes:

- Installing the microservice framework.
- Installing the microservice proxy (microservice support for the web tier).
- Starting the microservice node and any additional configuration for your platform type.



### 3. **Install Active Workspace.** This includes:

- Active Workspace server extensions
- Active Workspace client
- Indexing
- Visualization Server
- Other components, according to your business needs

## New environment

If you have no existing Teamcenter environment, you install Teamcenter, then add the microservice framework, then add Active Workspace:

### 1. Install a Teamcenter environment as described in the Teamcenter server installation guides for Windows or Linux.

Make sure you install a Teamcenter version supported for Active Workspace 5.2. For supported combinations of Teamcenter with Active Workspace, see the Hardware and Software Certifications knowledge base article on Support Center.

### 2. **Install and configure the microservice framework.** This includes:

- Installing the microservice framework.
- Installing the microservice proxy (microservice support for the web tier).
- Starting the microservice node and any additional configuration for your platform type.

### 3. **Install Active Workspace.** This includes:

- Active Workspace server extensions
- Active Workspace client
- Indexing
- Visualization Server
- Other components, according to your business needs

## 2. Installing Teamcenter microservices

### Microservices and the microservice framework

The microservice framework is the foundation for running microservices seamlessly across diverse platforms. All microservices require the framework. Various solutions and applications include microservices as part of their deployment. For example,

- The Active Workspace client requires DARSI, TcGQL, and File Repo microservices.
- The Classification and Requirements Manager applications each have their own required microservices.
- The Product Configurator application can optionally employ its application-specific microservice to achieve better performance.

To install the microservice framework and the microservices that run on it, you must configure and deploy a master microservice node. If the server hardware has sufficient capacity, you can choose to deploy microservice nodes on the same hardware as the Teamcenter pool managers.

Once a master microservice node is configured, *worker* microservice nodes (for non-Kubernetes deployments) can be added and configured in order to increase capacity and provide failover.

All master and worker microservice nodes in a Teamcenter environment must be hosted on servers of a single operating system type. The following table provides guidance in choosing the operating system type for servers that will host microservice nodes. It compares characteristics of microservice nodes hosted on Linux and Windows.

	Linux 64-bit	Windows 64-bit
Third party software required	<p>Kubernetes or Docker must be installed on the Linux host before you can install the microservice framework.</p> <p>For deployment in Kubernetes, a container registry is required.</p> <div><p>Note:</p><p>Currently, support for microservices in a Kubernetes environment is considered pre-release. Certification of deployment into Kubernetes is planned for a later release.</p></div>	None
Management of microservice framework and	Kubernetes or the Docker swarm manager starts, stops, restarts, and scales all Teamcenter microservices running as containers as needed to best utilize resources.	On Windows, each microservice framework node includes a Teamcenter process

	Linux 64-bit	Windows 64-bit
application microservices		manager to handle the microservices on that node.

## Means of installation

The following table describes the basic differences in the approach to installing the microservice framework and microservices, depending on the environment in which they are installed.

Host environment	Installation tool support		Configured Nodes	
	Deployment Center	Teamcenter Environment Manager (TEM)	Master	Worker
Docker on Linux	Yes	Yes	Required	Optional
Kubernetes on Linux	Yes	No	Required	Not used.  Scaling of microservice containers for capacity and failover is configurable in Kubernetes.
Windows	Yes	Yes	Required	Optional

## Microservice framework constituents

The microservice framework has the following primary constituents:

<b>Service Registry</b>	Maintains a list of running microservice instances across all nodes.
<b>Service Dispatcher</b>	Receives microservice requests from a Teamcenter client, queries the service registry to find an instance of the requested microservice, and then routes the request to an instance of the microservice.
<b>Microproxy</b>	Forwards web tier application requests to the service dispatcher.
<b>Process manager (Windows hosts)</b>	Manages microservices on the node (Windows hosts).  You can use a web-based utility to <b>dynamically adjust the quantity of microservice instances managed by the process manager.</b>
<b>Microservice Parameter Service (MPS)</b>	Manages logging levels for microservices.
<b>File Repo</b>	Manages files for Active Workspace and the microservice parameter service.

## Microservices and Docker on Linux hosts

If you configure microservice nodes or the Active Workspace Gateway for Docker on Linux hosts, you must install Docker on your Linux hosts before installing the framework. This can be done either through TEM or by running the Deployment Center deployment scripts for the hosts.

The following sections provide more information about microservices deployments into Docker:

- **Install Docker**
- **Configure IPv4 forwarding on the Docker host machine**
- **Working with Docker containers**
- **Docker troubleshooting**

### Install Docker

For certified versions of Linux and Docker software, refer to the Hardware and Software Certifications knowledge base article on Support Center.

Deploy the Docker swarm based on the installation instructions at <https://docs.docker.com>. Deployment requires the following steps at a minimum:

1. Install Docker.
2. Configure Docker to restart on system boot.

### Configure IPv4 forwarding on the Docker host machine

Docker requires that IP forwarding be enabled on the host machine for successful communication between Docker containers and the host machine. Docker installation alters the Linux iptables to allow forwarding of packets between the host and bridge networks when such forwarding is enabled. See Docker documentation for information on how to partially restrict forwarding (based on IP addresses) for tighter security.

IP forwarding is controlled by Linux kernel parameters such as **net.ipv4.ip\_forward** and **net.ipv4.conf.all.forwarding**, depending on the version and flavor of Linux. The current setting can be checked with the command **sysctl net.ipv4.ip\_forward** (sudo or root access is required). A value of 0 indicates that forwarding is disabled, while 1 indicates that forwarding is enabled.

1. To quickly enable IP forwarding for testing, issue the command:

```
sysctl -w net.ipv4.ip_forward=1
```

2. To restart docker, issue the command:

```
systemctl restart docker
```

- To preserve this setting across a machine reboot, edit the file `/etc/sysctl.conf` and set **net.ipv4.ip\_forward** to **1**.

## Working with Docker containers

Use common Docker commands to control the environment and monitor the status. Additionally, many open source tools, such as Portainer, are available to help manage the Docker swarm.

### Commonly used Docker commands

To do this	Run this command
List the Docker container stacks.	<b>docker stack ls</b>
List the services currently running.	<b>docker service ls</b>
Display the last five lines that were output by a particular service.	<b>docker service logs -f --no-task-ids --tail 5 <i>service_id</i></b>  <i>A <i>service_id</i> has the form <i>stackname_servicename</i>.</i>
List the nodes in a swarm.	<b>docker node ls</b>
List the images registered in the container registry on a node.	<b>docker image ls</b>

### Managing containers with Portainer

Portainer (<https://portainer.io/>) is an open source product that provides a web-based UI to easily manage Docker swarms, services, and containers. You can use Portainer to do the following:

- View Docker container log files.
- View the Docker applications (stacks) that have been started.
- View the status and location of running services.
- Manage the nodes in a swarm and temporarily adjust scaling of services across the swarm.

## Docker troubleshooting

If you receive the following error when running any Docker command:

```
Cannot connect to the Docker daemon
```

then do the following.

- To check whether **dockerd** is running, run

```
ps -eaf | grep dockerd
```



2. Perform remedial steps depending on the result from Step 1.

If dockerd is	Then do this
Not running	Restart Docker, and configure <b>dockerd</b> to restart on the next boot:  <pre>sudo systemctl start docker sudo systemctl enable docker</pre>
Running	The user is likely not a member of the Docker Linux group. Add the user to the group. Ignore any error output from <b>groupadd</b> .  <pre>sudo groupadd docker sudo usermod -aG docker \$USER</pre>

### What do I do when I receive the error **Cannot connect to the Docker daemon**?

For more debugging information, refer to Docker information on configuring the Docker daemon.

### What do I do if the Docker command does not behave as expected?

If the Docker command does not behave as expected, add the **-debug** option, run the command again, and review the log for issues.

Example:

You run the command **docker deploy -f mystack.yml mystack** and it does not behave as expected.

To enable logging, insert **-debug** after **docker**:

**docker -debug deploy -f mystack.yml mystack.**

### How do I view logs from the Docker daemon?

To view logs from the Docker daemon, open a new shell and enter the following:

```
sudo journalctl -fu docker.service
```

This tails the log files and keeps outputting new log commands until the command prompt is closed or you enter **Ctrl-C**.

### Where can I get help with more complicated environments?

For help with more complicated environments and networking when microservice nodes are on Linux hosts, see the Docker documentation at <https://docs.docker.com/engine/swarm/swarm-mode/>.

## Microservices and Kubernetes on Linux hosts

### Note:

Currently, support for microservices in a Kubernetes environment is considered pre-release. Certification of deployment into Kubernetes is planned for a later release.

The microservice framework and microservices can be deployed into a Kubernetes environment on Linux hosts. If deployed into Kubernetes, only a single microservice node must be configured in Deployment Center. Once the node is deployed into the Kubernetes environment, the scaling of microservices for capacity and failover can be configured in Kubernetes.

For certified versions of Linux and Kubernetes software, refer to the Hardware and Software Certifications knowledge base article on Support Center.

### Kubernetes information needed for configuring the microservice node in Deployment Center

Kubernetes-specific information needed for configuring the microservice node in Deployment Center is as follows:

<b>Container Registry</b>	This is the registry URL to store Docker images.
<b>Namespace</b>	This is the unique name that identifies the group of Teamcenter resources interacting with each other in a Kubernetes cluster.
<b>Additional Service Dispatcher URLs</b>	These are the URLs for additional endpoints to access the service dispatcher from outside the cluster.

### Configure IPv4 forwarding on the Kubernetes host machine

IP forwarding must be enabled on the host machine for successful communication between containers and the host machine. The Linux iptables must be configured to allow the forwarding of packets between the host and bridge networks when such forwarding is enabled. See Kubernetes software documentation for information on how to partially restrict forwarding (based on IP addresses) for tighter security.

IP forwarding is controlled by Linux kernel parameters such as **net.ipv4.ip\_forward** and **net.ipv4.conf.all.forwarding**, depending on the version and flavor of Linux. The current setting can be checked with the command **sysctl net.ipv4.ip\_forward** (sudo or root access is required). A value of 0 indicates that forwarding is disabled, while 1 indicates that forwarding is enabled.

1. To quickly enable IP forwarding for testing, issue the command:

```
sysctl -w net.ipv4.ip_forward=1
```

2. To restart docker, issue the command:

**systemctl restart docker**

- To preserve this setting across a machine reboot, edit the file `/etc/sysctl.conf` and set **net.ipv4.ip\_forward** to 1.

## Install microservices using TEM

### Install microservices using TEM- Docker on Linux host

Before you install microservices:

- Review **Microservices and the microservice framework**.
- Install a certified version of Docker on the host.  
For certified versions of Linux and Docker software, see the Hardware and Software Certifications knowledge base article on Support Center.
- Download the microservice framework software kit (**TcMicroserviceFramework5.2.0\_platform.zip**) from the Siemens Digital Industries Software download site. Expand the kit to a local directory.

### Add microservice features to TEM

Add microservice features to Teamcenter Environment Manager (TEM), then **install Teamcenter microservices**.

For this task	Perform these steps
Installing a new Teamcenter environment	<ol style="list-style-type: none"> <li>Launch TEM (<b>tem.sh</b>) from the Teamcenter software kit.</li> <li>Proceed to the <b>Install/Upgrade Options</b> panel and click <b>Install</b>.</li> <li>Proceed to the <b>Media Locations</b> panel. In the <b>Update Location</b> box, enter the path to the microservice framework software kit. If you are installing a minor release of Teamcenter, enter the path to the major release software kit in the <b>Original Media Location</b> box.</li> <li>In the <b>Configuration</b> panel, enter an ID and description for the Teamcenter configuration.</li> <li>Proceed to the <b>Features</b> panel, and then <b>install Teamcenter microservices</b>.</li> </ol>
Updating an existing Teamcenter environment	<ol style="list-style-type: none"> <li>Stop all Teamcenter services except FSC services.</li> <li>Launch TEM from its installed location (<b>TC_ROOT/install</b>).</li> <li>In the <b>Maintenance</b> panel, select <b>Updates Manager</b>.</li> </ol>

For this task	Perform these steps
	<ol style="list-style-type: none"> <li>4. In the <b>Apply Updates</b> panel, enter required paths: <ul style="list-style-type: none"> <li><b>Update kit location</b> Enter the path to the expanded microservice framework software kit.</li> <li><b>Backup directory</b> Enter a path in which to create backups of files replaced during the update.</li> <li><b>Original Media Location</b> Enter the path to the software kit for the installed Teamcenter major version.</li> </ul> </li> <li>5. Proceed to the <b>Confirmation</b> panel and click <b>Start</b> to add microservices features to TEM. When the update is complete, close TEM.</li> <li>6. Launch TEM as in step 2.</li> <li>7. In the <b>Configuration Maintenance</b> panel, select <b>Perform maintenance on an existing configuration</b>.</li> <li>8. In the <b>Old Configuration</b> panel, select the configuration to update.</li> <li>9. In the <b>Feature Maintenance</b> panel, select <b>Add/Remove Features</b>.</li> <li>10. Proceed to the <b>Features</b> panel, and then <b>install Teamcenter microservices</b>.</li> </ol>

## Install Teamcenter microservices

1. In the **Features** panel in TEM, under **Microservices**, select **Microservice Framework** and the microservices you want to install.

If you install multiple microservice nodes, make sure you install *all* the same microservices on every microservice node that you will join to the Docker swarm.

2. Proceed to the **Microservice Framework** panel and specify values.

- a. Choose the node type to install, either **Master Node** or **Worker Node**.

A *master* microservice node is required in an environment.

- b. If you are installing a *master* microservice node, then enter a **Keystore Password** to be used for generating the *.p12* files that contain keys for signing and validating authentication tokens. These tokens identify the logged in Teamcenter user.

Record and keep secure the password for potential use in case you want to open and edit the keys.

- c. If you are installing a *worker* microservice node, skip to step 3.

If you are installing a *master* microservice node, in **Dispatcher Port** type the port for the service dispatcher that will run on this host. All service dispatchers must use the same port value.

The file `tc_microservice_framework.yml` must list all service dispatcher URLs that you want to include as endpoints. If for purposes of failover you install multiple master nodes, then edit each `tc_microservice_framework.yml` file to include all service dispatcher endpoints.

(For installation via Deployment Center, all instances are automatically included in the list.)

In Docker on Linux hosts, the microservice framework service *registry* is handled internally. If only one master node and service registry is configured in the environment, then no service registry port value or URL value is needed in `.yml` files.

When multiple service registry access points exist, edit each `.yml` file to list all service registry instances.

Example:

```
{
  "eureka1":{
    "image":"eureka_server-1.9.12_1.2.1",
    "environment":[
      "ARGS=-Deureka.serviceUrl.default=
        http://eureka1:8787/eureka/v2,
        http://eureka2:8787/eureka/v2,
        http://eureka3:8787/eureka/v2"
      "JETTY_PORT=8787"
    ]
  }
}
{
  "eureka2":{
    "image":"eureka_server-1.9.12_1.2.1",
    "environment":[
      "ARGS=-Deureka.serviceUrl.default=
        http://eureka1:8787/eureka/v2,
        http://eureka2:8787/eureka/v2,
        http://eureka3:8787/eureka/v2"
      "JETTY_PORT=8787"
    ]
  }
}
```



```

    }
    {
      "eureka3":{
        "image":"eureka_server-1.9.12_1.2.1",
        "environment":[
          "ARGS=-Deureka.serviceUrl.default=
            http://eureka1:8787/eureka/v2,
            http://eureka2:8787/eureka/v2,
            http://eureka3:8787/eureka/v2"
          "JETTY_PORT=8787"
        ]
      }
    }
  }
}

```

3. If you are installing a *master* node, proceed to the **Microservices** panel. Review the instance quantity for each service that you want to be available in the Docker swarm. Increase the instance quantity as needed to handle system load.
4. Proceed through the remaining panels and enter configuration values for your selected microservices as TEM prompts for them.
5. Review the **Confirmation** panel and click **Start** to begin installation.

The framework and microservices are installed. If you are installing a *master* node, then signer and validator keys are generated.

6. When installation completes, carefully examine any messages to determine what post-installation steps are necessary.
7. Add microservices support to the Teamcenter web tier, as applicable for **Java EE** or **.NET** web tier architecture.
8. **Start the Docker swarm, the framework container, and the services.**

## Install microservices using TEM- Windows host

Before you install microservices:

- Review **Microservices and the microservice framework**.
- Download the microservice framework software kit (**TcMicroserviceFramework5.2.0\_platform.zip**) from the Siemens Digital Industries Software download site. Expand the kit to a local directory.

## Add microservice features to TEM

Add microservice features to Teamcenter Environment Manager (TEM), then **install Teamcenter microservices**.

For this task	Perform these steps
Installing a new Teamcenter environment	<ol style="list-style-type: none"> <li>1. Launch Teamcenter Environment Manager (TEM) with administrative privileges. In the Teamcenter software kit, right-click the <b>tem.bat</b> program icon and choose <b>Run as administrator</b>.</li> <li>2. Proceed to the <b>Install/Upgrade Options</b> panel and click <b>Install</b>.</li> <li>3. Proceed to the <b>Media Locations</b> panel. In the <b>Update Location</b> box, enter the path to the microservice framework software kit. If you are installing a minor release of Teamcenter, enter the path to the major release software kit in the <b>Original Media Location</b> box.</li> <li>4. In the <b>Configuration</b> panel, enter an ID and description for the Teamcenter configuration.</li> <li>5. Proceed to the <b>Features</b> panel, and then <b>install Teamcenter microservices</b>.</li> </ol>
Updating an existing Teamcenter environment	<ol style="list-style-type: none"> <li>1. Stop all Teamcenter services except FSC services.</li> <li>2. Launch TEM from its installed location, with administrative privileges. Choose <b>Start &gt; Teamcenter version</b>, right-click <b>Environment Manager</b> and choose <b>Run as administrator</b>.</li> <li>3. In the <b>Maintenance</b> panel, select <b>Updates Manager</b>.</li> <li>4. In the <b>Apply Updates</b> panel, enter required paths: <ul style="list-style-type: none"> <li><b>Update kit location</b> Enter the path to the expanded microservice framework software kit.</li> <li><b>Backup directory</b> Enter a path in which to create backups of files replaced during the update.</li> <li><b>Original Media Location</b> Enter the path to the software kit for the installed Teamcenter major version.</li> </ul> </li> <li>5. Proceed to the <b>Confirmation</b> panel and click <b>Start</b> to add microservices features to TEM. When the update is complete, close TEM.</li> <li>6. Launch TEM as in step 2.</li> </ol>

For this task	Perform these steps
	<ol style="list-style-type: none"> <li>7. In the <b>Configuration Maintenance</b> panel, select <b>Perform maintenance on an existing configuration</b>.</li> <li>8. In the <b>Old Configuration</b> panel, select the configuration to update.</li> <li>9. In the <b>Feature Maintenance</b> panel, select <b>Add/Remove Features</b>.</li> <li>10. Proceed to the <b>Features</b> panel, and then <b>install Teamcenter microservices</b>.</li> </ol>

## Install Teamcenter microservices

1. In the **Features** panel in TEM, under **Microservices**, select **Microservice Framework** and the microservices you want to install.
2. Proceed to the **Microservice Framework** panel and specify values.

- a. Choose the node type to install, either **Master Node** or **Worker Node**.

A *master* microservice node is required in an environment.

- b. If you are installing a *master* microservice node, then enter a **Keystore Password** to be used for generating the *.p12* files that contain keys for signing and validating authentication tokens. These tokens identify the logged in Teamcenter user.

Record and keep secure the password for potential use in case you want to open and edit the keys.

- c. If you are installing a *worker* microservice node, then in **Keystore ZIP file** enter the location of the **keys.zip** file that was generated when the *master* node was installed.

The keystore ZIP file (*keys.zip*) was copied to the *TC\_ROOT/jwt\_config\_tool* directory of the *master* microservice node host. For your convenience when installing a *worker* microservice node, copy that file to a location on the local machine and in **Keystore ZIP file** point to that local copy. For security, a best practice is to delete the copy of *keys.zip* from the local machine after the *worker* node installation is complete.

- d. Enter parameter values for the service dispatcher and service registry:

**Service Dispatcher** parameters:

**Install Microservice** On a *master* microservice node, the service dispatcher is required and preselected. On a *worker* microservice node, you may optionally install the service dispatcher for failover or scalability.

<b>Dispatcher Port</b>	Type the port for the service dispatcher that will run on this host.
<b>Additional URLs</b>	Type the URLs for additional endpoints for accessing the service dispatcher. This field can be empty or can include URLs for one or more microservice nodes. Use a comma to separate multiple URLs. For example:

```
http://masternode_1:9090,http://workernode1:9090,http://workernode2:9090
```

If service dispatchers are configured on multiple nodes (Windows hosts) in an environment, the file *service\_dispatcher.json* on each node must list all instances.

If service dispatcher instances are added or changed, then edit the file on each node where a service dispatcher is installed as needed to update the file. The default location of the file is *[tc\_root]\Clients\aw\microservices\services\_config*. Similarly, some microservice .json files include an ARGV or DSP variable. In such cases, update the variable value in each .json file on each node.

(For installation via Deployment Center, all instances are automatically included in the list.)

#### Service Registry (eureka) parameters:

<b>Install Microservice</b>	On a <i>master</i> microservice node, the service registry is required and preselected. On a <i>worker</i> microservice node, you may optionally install the service registry for failover or scalability.
<b>Registry Port</b>	Type the port for the service registry that will run on this host.
<b>Additional URLs</b>	Type the URLs to all other instances of the service registry in the environment, separated by commas.

The URL pattern for a service registry is:

```
http://<host>:port/eureka/v2/
```

where <host> is the hostname, fully qualified domain name, or IP address.

When multiple microservice nodes with service registry instances are configured in an environment, every .json file for every microservice on every node must list all instances. If service registry instances are added or changed, then edit the files as needed to update the list of service registries (eureka). (For installation via Deployment Center, all instances are automatically included in the list.)

Example:

**eureka** definitions when a service registry exists on port 8787 on servers JPS008601, JPS008602, and JPS008603:

```
{
  "eureka":{
    "image":"eureka_server-1.9.12_1.2.1",
    "environment":[
      "ARGS=-Deureka.serviceUrl.default=
        http://JPS008601:8787/eureka/v2,
        http://JPS008602:8787/eureka/v2,
        http://JPS008603:8787/eureka/v2"
      "JETTY_PORT=8787"
    ]
  }
}
```

3. If you are installing a *master* node, proceed to the **Microservices** panel. Review the instance quantity for each service that you want to run on this node. Typically, Teamcenter microservices are multi-threaded, so only one instance of the microservice is needed on a node.

When the environment includes multiple microservice nodes on Windows hosts, you may want to run only a subset of microservices on a given node. In that case, for microservices that you do not want to install on the node, set the instance value to **0**.

4. Proceed through the remaining panels and enter configuration values for your selected microservices as TEM prompts for them.
5. Review the **Confirmation** panel and click **Start** to begin installation.

The framework and microservices are installed. If you are installing a *master* node, then signer and validator keys are generated.

6. When installation completes, carefully examine any messages to determine what post-installation steps are necessary.
7. Add microservices support to the Teamcenter web tier, as applicable for **Java EE** or **.NET** web tier architecture.



## Add microservice framework support to a Java EE web tier

If you used Teamcenter Environment Manager to install the microservice framework, use the following procedure to add microservice framework support to a Teamcenter Java EE architecture web tier WAR file.

This procedure assumes you installed the Web Application Manager and the *WEB\_ROOT* directory as described in the appropriate Teamcenter installation guide for Windows or Linux.

If you use scripts generated by Deployment Center to install the microservice framework, you do not need to manually perform this procedure.

1. Install a microservice framework *master* node.

Locate the keystore ZIP file (*keys.zip*) in the *TC\_ROOT/jwt\_config\_tool* directory of the *master* microservice node host.

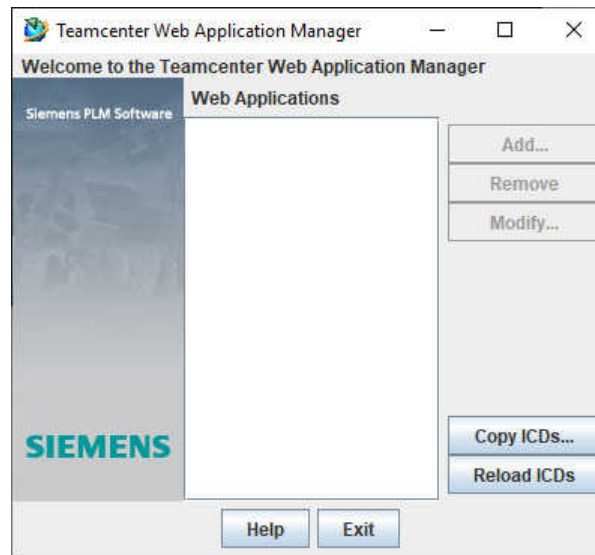
2. Copy the keystore ZIP file (*keys.zip*) to a location on your local machine and extract the ZIP file contents.

The keystore ZIP file contains a directory named **signer\_config**.

**Caution:**

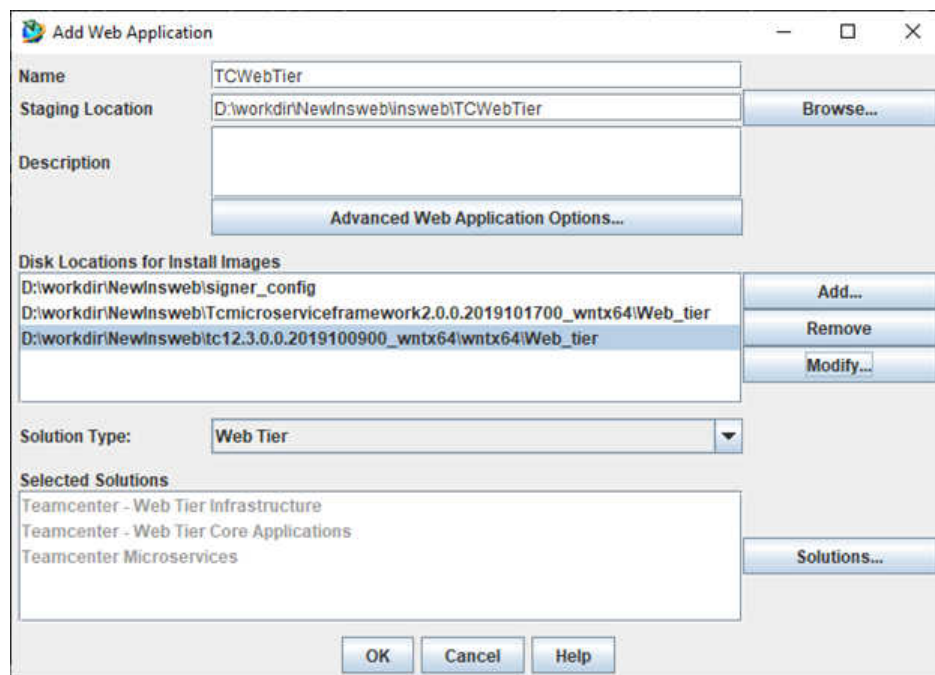
Safeguard the keystore files. A best practice is to remove them from any temporary locations once you complete this procedure.

3. Locate the following software kits:
  - Teamcenter software kit for your installed version of Teamcenter
  - Microservices Framework
4. Change to the *WEB\_ROOT* directory and launch the Web Application Manager using the appropriate command:
  - Windows systems: **insweb.bat**
  - Linux systems: **insweb.sh**



5. Copy ICD files from the software kits. This populates the list of solutions available to install.
  - a. Click **Copy ICDs**.
  - b. In the **Copy ICD Files** dialog box, click **Browse** and browse to the following location:  
*Teamcenter-kit\Web\_tier\icd*
  - c. Click **OK** to copy ICD files from the kit.
  - d. Repeat steps a through c, specifying the path to the Microservices Framework ICD files:  
*Microservices-Framework-kit\Web\_tier\icd*
6. Click **Add** to begin creating a new web application.
7. In the **Add Web Application** dialog box, create the web application:
  - a. Type a **Name** and **Staging Location** for the web application.
  - b. Enter software locations:
    - A. Click **Add**.
    - B. In the **Add Disk Location** dialog box, enter software locations:
      - i. Click **Browse**, browse to the **signer\_config** directory from the keystore ZIP file you extracted in step 2, and then click **Apply**.

- ii. Click **Browse**, browse to the location of the Microservice Framework kit, and then click **Apply**.
  - iii. Click **Browse**, browse to the location of the Teamcenter software kit, and then click **OK**.
- c. Select solutions:
  - A. Click **Solutions**.
  - B. In the **Select Solutions** dialog box, select the following web tier solutions:
    - **Teamcenter - Web Tier Infrastructure**
    - **Teamcenter - Web Tier Core Applications**
    - **Teamcenter Microservices**
8. Verify your selections, and then click **OK** to continue creating the web application.



9. In the **Modify Required Context Parameters** dialog box, ensure the following context parameters have correct values:

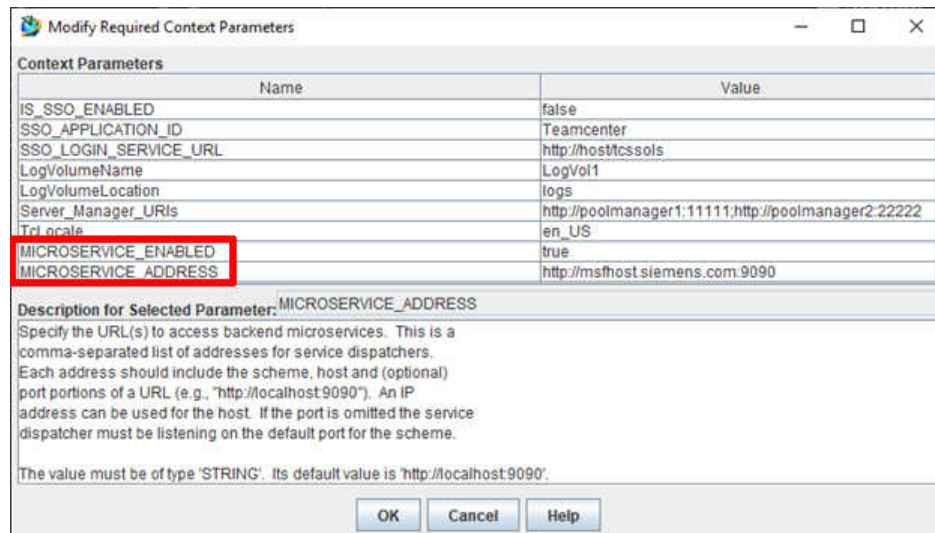
**MICROSERVICE\_ENABLED**

Set to **true**.

**MICROSERVICE\_ADDRESS**

Specifies a comma-separated list of service dispatcher URLs of the form:

**http://host:port**



- Click **OK** to begin building the WAR file.

When the web application generation is complete, close the Web Application Manager.

- Locate the WAR file (**tc.war**) in the **deployment** directory under the staging location you specified.
- Deploy the web application on a supported application server, as described in *Web Application Deployment* in the Teamcenter help.

## Add microservice framework support to a .NET web tier

In the typical case that the Teamcenter .NET architecture web tier machine does not also have a microservice node installed on it, and installation is through TEM, the following procedure adds microservice framework support (the microservice proxy).

The procedure is performed automatically if TEM is used to install both a master microservice node and the .NET web tier on the same host, or if installation is through Deployment Center.

If you use a distributed environment in which your server managers are on hosts separate from your microservice nodes, make sure that on the local machine that runs IIS, the **IIS\_IUSRS** ID has full control permissions on the **net\_webtier\_configuration** directory. **IIS\_IUSRS** is a local user account created by IIS.

- Install a *master* microservice node on some machine in the Teamcenter environment.

The keystore file *signer\_keystore.p12* is installed to the *TC\_ROOT\signer\_config* directory of the microservice node host.

2. On the web tier machine, before starting IIS, copy the private key **signer\_keystore.p12** from the microservice node host to **C:\windows\System32\inetsrv**.
3. From the microservice framework kit, extract the file *dotnet\_microproxy-1.0.1.zip* and then extract its contents into the IIS bin folder on the web tier machine.
4. On the web tier machine, modify *ApplicationConfiguration.xml*, which you can find in the *TC\_ROOT\net\_webtier\_Configuration ID\tc\* directory.

- Set **microserviceEnabled** to **true**.

```
<!-- Enable/disable microservice requests. (Value: true/false) -->
<param name="microserviceEnabled" value="true"></param>
```

- Set **microserviceAddress** to the location of the microservice dispatcher(s).

```
<!-- Specify the URL(s) to access backend microservices. This is a
comma-separated list of addresses for service dispatchers.
Each address should include the scheme, host and (optional)
port portions of a URL (e.g., "http://localhost:9090"). An IP
address can be used for the host. If the port is omitted the
service
dispatcher must be listening on the default port for the scheme.
-->
<param name="microserviceAddress" value="http://localhost:9090"></
param>
```

- Set the microservice **keystore**.

```
<!-- Name of the keystore file that stores private key to sign
JWT. -->
<param name="microserviceKeyStore" value="signer_keystore.p12"></
param>
```

- Set the microservice keystore password. This is the password that was entered when installing the microservice framework master node.

```
<!-- Password to read private key to sign JWT. -->
<param name="microserviceKeyStorePassword" value="password"></param>
```

- Set the login-manager for microservices.

```
<!-- Micro Services -->
```

```

<injector id="micro"
handler="Framework,Teamcenter.Presentation.Framework.RequestProcess
ing.RequestInjectorImpl">
  <!--<pre-trigger
handler="Micro,Teamcenter.WebIntegrations.Micro.MicroPreTrigger"></
pre-trigger>-->
  <login-manager
handler="Micro,Teamcenter.WebIntegrations.Micro.MicroLoginManager"
login-failure-response-chain="microLoginFailure_response_chain">
  </login-manager>
  <!-- for urls tc/micro -->
  <group-association-path id="micro"></group-association-path>
  <config-param name="DEFAULT_LINK" value="microInvoke_link"></
config-param>
  <error-handler
response-handler="Micro,Teamcenter.WebIntegrations.Micro.MicroRespo
nseHandler"></error-handler>
</injector>

```

5. Start IIS and in the **Application Pools > DefaultAppPool > Advanced Properties** page, set **Load User Profile** to **True**.

## Install microservices using Deployment Center

### Install microservices on a Linux host via Deployment Center

**Note:**

Currently, support for microservices in a Kubernetes environment is considered pre-release. Certification of deployment into Kubernetes is planned for a later release.

1. Download a compatible Teamcenter microservice framework kit and place it in the Deployment Center software repository.
2. In Deployment Center, open or create an environment.
3. On the **Software** task, add **Microservice Framework**.
4. On the **Applications** task, add the applications that you want to install in the environment.
5. On the **Components** task, specify values for the **Microservice Node** options.

For worker nodes deploying into Docker on a Linux host, only a small subset of the following options appear.

For this option	Do this
<b>Installation Path</b>	Enter the path to the Teamcenter installation root folder on the microservice node host machine.
<b>Machine Name</b>	Enter the fully qualified domain name of the microservice node host machine. This machine name is used to construct the service dispatcher URL
<b>OS</b>	Choose <b>lnx64</b> (Linux).
<b>Microservice Node Type</b>	<p>Choose one of two node types:</p> <p><b>Master</b></p> <p>The master (manager) microservice node in the Teamcenter environment. Exactly one master-type microservice node is required in an environment. A master node must be configured before worker nodes are configured.</p> <p><b>Worker</b></p> <p>A worker microservice node in the Teamcenter environment.</p> <p>You can add worker-type Microservice Node components as needed.</p> <p>When deploying into Kubernetes, do not configure worker nodes in Deployment Center. Instead, use the <b>Additional Service Dispatcher URLs</b> option. This option is made available when <b>Container Manager</b> is set to <b>Kubernetes</b>.</p>
<b>Container Manager</b>	<p>Choose one of two container manager types, <b>Docker Swarm</b> or <b>Kubernetes</b>. For Kubernetes, specify:</p> <ul style="list-style-type: none"> <li>• <b>Container Registry</b> (the registry URL to store Docker images)</li> </ul> <div style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <p>Example:</p> <pre>registry.example.com.</pre> </div> <ul style="list-style-type: none"> <li>• <b>Namespace</b> (the unique name that identifies the group of Teamcenter resources interacting with each other in a Kubernetes cluster)</li> </ul>
<b>Keystore Password and Confirm Password</b>	<p>Enter a password to be used for generating the .p12 files that contain keys for signing and validating authentication tokens. The tokens identify the logged in user.</p> <p>Record and keep secure the password for potential use should you want to open and edit the keys.</p>
<b>Protocol</b>	<p>Choose the protocol to use for moving data between the Teamcenter web tier and the service dispatcher.</p> <p>If you choose <b>https</b>, then you need to perform additional steps after running deployment scripts to <b>Configure service dispatcher as an HTTPS server</b>.</p>



For this option	Do this
<b>Port</b>	Enter the port number for communication with the service dispatcher.  For Kubernetes, the valid port range is 30000 to 32767.
<b>Additional Service Dispatcher URLs</b>	In the case of deployment on Kubernetes, enter the URLs for additional endpoints to access the service dispatcher from outside the cluster. This field can be empty or can include URLs, including port values, for one or more Kubernetes worker nodes. Port and protocol values in the additional URLs should be the same as those specified in <b>Protocol</b> and <b>Port</b> . Use commas to separate multiple URLs.  <div style="border: 1px solid black; padding: 10px; margin-top: 10px;"> <p>Example:</p> <pre>http://workernode1:30090,http://workernode2:30090</pre> </div>
<b>Teamcenter Microservice URL and Service Registry URL</b>	As applicable, Deployment Center supplies these values as you complete the environment configuration.

- Enter microservice parameter values as required. The parameters shown vary, depending on which applications are selected for the environment.

Example:

Active Workspace uses a file repository microservice. To configure that service for deployment on a Linux host, parameter values **Deploying User UID**, **Deploying User GID**, and **File Repository Storage Location** values are required. Values entered for the master microservice node must be valid on all worker nodes.

For Kubernetes, the user cannot be **root**.

- In the **Services** list, review the quantity of instances for each service.

To increase capacity, increase the number of instances.

- Save the component settings.
- Complete configuration of the environment and generate deployment scripts.
- Depending on the container manager, ensure that Docker or Kubernetes is appropriately installed on the microservice node host before you run its deployment script. As applicable, refer to the following:
  - Microservices and Docker on Linux hosts**
  - Microservices and Kubernetes on Linux hosts**

11. If you are deploying to Kubernetes, login to the container registry before starting actual deployment.

```
docker login -u "<user>" -p "<password>" <container registry URL>
```

12. Run the deployment scripts.

Run the microservice node scripts before you run the web tier deployment script.

13. If your environment uses the .NET architecture, then on the web tier server machine where IIS is running, in the Application Pools > DefaultAppPool > Advanced Properties, set **Load User Profile** to **True**.
14. If you chose the **https** protocol for moving data between the Teamcenter web tier and the service dispatcher, then **Configure service dispatcher as an HTTPS server**.
15. Depending on the container manager, follow the appropriate instructions to complete the installation and start the microservices:

**Docker**      **Starting microservices in Docker.**  
**Swarm**

**Kubernetes**      **Starting microservices in Kubernetes.**

## Install microservices on a Windows host via Deployment Center

1. Download a Teamcenter microservice framework kit and place it in a Deployment Center software repository.
2. In Deployment Center, open or create an environment.
3. On the **Software** task, add **Microservice Framework**.
4. On the **Applications** task, add the applications that you want to install in the environment.
5. On the **Components** task, specify values for the **Microservice Node** options.

For this option	Do this
<b>Installation Path</b>	Enter the path to the Teamcenter installation root folder on the microservice node host machine.
<b>Machine Name</b>	Enter the fully qualified domain name of the microservice node host machine. This machine name is used to construct the Service Dispatcher URL.
<b>OS</b>	Choose <b>wntx64</b> (Windows).

For this option	Do this						
	Check <b>Install Teamcenter Process Manager as a Windows service</b> to automatically start services when the server reboots.						
<b>Microservice Node Type</b>	<p>Choose one of two node types:</p> <p><b>Master</b></p> <p>The master microservice node in the Teamcenter environment. Exactly one master microservice node is required in an environment. A master node must be configured before worker nodes are configured.</p> <p><b>Worker</b></p> <p>A worker microservice node in the Teamcenter environment.</p> <p>You can add worker Microservice Node components as needed.</p>						
<b>Keystore Password and Confirm Password</b>	<p>Enter a password to be used for generating the .p12 files that contain keys for signing and validating authentication tokens. The tokens identify the logged in user.</p> <p>Record and keep secure the password for potential use should you want to open and edit the keys.</p>						
<b>Protocol</b>	<p>Choose the protocol to use for moving data between the Teamcenter web tier and the Service Dispatcher. The default protocol is <b>http</b>. If the Teamcenter architecture type is <b>Java EE</b>, then you have the option of choosing <b>https</b>.</p> <table border="1"> <thead> <tr> <th>For this web tier architecture</th><th>Do this</th></tr> </thead> <tbody> <tr> <td>.NET</td><td>Choose <b>http</b>.</td></tr> <tr> <td>Java EE</td><td>           Choose either <b>http</b> or <b>https</b>.             If you choose <b>https</b>, then you need to perform some additional steps later to <b>Configure service dispatcher as an HTTPS server</b>.         </td></tr> </tbody> </table>	For this web tier architecture	Do this	.NET	Choose <b>http</b> .	Java EE	Choose either <b>http</b> or <b>https</b> .  If you choose <b>https</b> , then you need to perform some additional steps later to <b>Configure service dispatcher as an HTTPS server</b> .
For this web tier architecture	Do this						
.NET	Choose <b>http</b> .						
Java EE	Choose either <b>http</b> or <b>https</b> .  If you choose <b>https</b> , then you need to perform some additional steps later to <b>Configure service dispatcher as an HTTPS server</b> .						
<b>Port</b>	<p>As applicable, enter the port number for communication with the Service Dispatcher and the Service Registry.</p> <p>Both the Service Dispatcher and the Service Registry are required on the master node.</p>						
<b>Teamcenter Microservice URL and Service Registry URL</b>	As applicable, Deployment Center supplies these values as you complete the environment configuration.						

- Enter microservice parameter values as required. The parameters shown vary, depending on which applications are selected for the environment.

**Example:**

Active Workspace uses a file repository microservice. To configure that service for deployment on a Windows host, a parameter value for **File Repository Storage Location** is required.

7. In the **Services** list, review the quantity of instances for each service.

Typically, Teamcenter microservices are multi-threaded, so only one instance of the microservice is needed on a server.

When the environment includes multiple microservice nodes, you may want to run only a subset of microservices on a given node. In that case, for microservices that you do not want to install on the node, set the instance value to zero.

8. Save the component settings.
9. Complete configuration of the environment and generate deployment scripts.
10. Run the deployment scripts.

Run the microservice node scripts before you run the web tier deployment script.

11. If your environment uses the .NET architecture, then on the web tier server machine where IIS is running, in the Application Pools > DefaultAppPool > Advanced Properties, set **Load User Profile** to **True**.
12. If you chose the **https** protocol for moving data between the Teamcenter web tier and the Service Dispatcher, then **Configure service dispatcher as an HTTPS server**.
13. Start the framework and services.

On a host running the Windows operating system, microservice processes are started by the Teamcenter Process Manager. The Teamcenter Process Manager either can be started as a Windows service, or you must run a startup file to start the Teamcenter Process Manager.

**Windows service** If you selected the **Install the Teamcenter Process Manager as a Windows service** option, then the Teamcenter Process Manager starts automatically with system startup.

The Teamcenter Process Manager appears in the Windows service list as **Teamcenter Process Manager** if POOL\_ID is not defined, else **Teamcenter Process Manager <POOL\_ID>**.

**Startup file** Run the Teamcenter Process Manager startup file:

```
[installation directory] \process_manager\start_manager.bat
```

## Add microservices and microservice nodes for Docker on a Linux host

To increase capacity of heavily used microservices deployed to Docker on Linux hosts, you can add microservice nodes and microservice instances via Deployment Center.

### Add microservice instances

1. In Deployment Center, on the **Components** task for your environment, open the **Microservice Node** for the master node.
2. In the list of microservices, change values for the instances as desired.
3. Complete your environment configuration and follow the Deployment Center instructions for deploying the generated zip files onto the target machine(s).
4. **Start the Docker swarm, the framework container, and the services.**

### Add a worker node

1. In Deployment Center, on the **Components** task for your environment, add a new **Microservice Node**.
2. Set the new node **Microservice Node Type** to **worker**.

The displayed node parameters needing specification reduce to the **Installation Path** and **Machine Name**.

3. Complete your environment configuration and follow the Deployment Center instructions for deploying the generated zip files onto the target machine(s).
4. **Start the Docker swarm, the framework container, and the services.**

## Add microservices and microservice nodes on a Windows host

To increase capacity of heavily used microservices deployed on Windows hosts, you can add microservice nodes and microservice instances via Deployment Center.

1. In Deployment Center, on the **Components** task for your environment, select an existing microservice node component or add a new **Microservice Node** component.
2. Configure the node, including the microservices you want to run on the node, as described in **Install microservices on a Windows host via Deployment Center**.

3. Complete your environment configuration and follow the Deployment Center instructions for deploying the generated zip files onto the target machine(s).
4. **Start the framework and services.**

## Starting microservices in Docker

If the microservice framework is deployed for Docker on a host running a Linux operating system, use the following procedure to start Docker and then start microservices.

### Start Docker

To start Docker on a microservice framework node, run the following command:

```
docker swarm init
```

The output of the command is similar to the following:

```
Swarm initialized: current node (lccilqci5tpvy6xmsjlu8gap3) is now a manager.
```

To add a worker to this swarm, run the following command:

```
docker swarm join --token SWMTKN-1-26h1be2gk2kozzecvgkw93smho5ueb7azn8uw1j2079
isc8b25-dfc8r1f6qhh50ev250tb4st9r 192.168.0.8:237
```

#### Tip:

If this is the master node and you intend to later join other nodes to this swarm as workers, save the output command string for later use.

Once you have started Docker on a node, you can **join the node to a running swarm**.

### Start the microservice stack (master node)

During installation of microservice nodes, one master node must be configured. Microservice `.yaml` files are copied to this master node. These files define the microservice container parameters and are used to start the microservice containers on the master node. Once the stack of containers is started on the master node, the Docker swarm manages the stack across all microservice nodes, automatically starting containers as needed on other nodes that join the swarm. The same containers must exist on all worker and manager nodes.

1. Change to the Docker `<installation-path>/container` directory.
2. Run the following command to start a stack for the microservice framework service **tc\_microservice\_framework.yml**:

```
docker stack deploy -c tc_microservice_framework.yml <myStackName>
```

- Using the same command pattern and the same stack name, start all other *.yml* files in the directory.

## Join a node to a running Docker swarm

Once the master node is running, you can join additional nodes to the swarm as either *workers* or *managers*. Any number of nodes can be added as workers. If the swarm includes multiple manager nodes, then the Docker swarm manager nodes vote to determine which node is the controlling node. To ensure a decisive vote, the swarm must have an odd number of manager nodes.

- Start Docker** on the node.
- Depending on whether you want to join the node as a worker or as a manager, use the corresponding procedure from the following table.

For this join mode	Do this
<i>worker</i>	Run the Docker command that you saved from the output when the swarm was started on the master node.

```
docker swarm join --token
SWMTKN-1-26hlbe2gk2k0zzecvgkw93smho5ueb7azn8uw1j2079isc8b25-dfc8r1f6qhh50ev250
tb4st9r 192.168.0.8:237
```

If a saved join token is not available, then on the master node, run the following command to request a token:

```
docker swarm join-token
```

*manager* a. Ensure that in Teamcenter Web Application Manager (**insweb**), you configure the Teamcenter *.war* file to include the node host's URL in the **Context Parameters** value list for **MICROSERVICE\_ADDRESS**.

- On the master node, run the following command to request a manager token:

```
$ docker swarm join-token manager
```

The output of the command is similar to the following:

To add a manager to this swarm, run the following command:

```
docker swarm join --token
SWMTKN-1-26hlbe2gk2k0zzecvgkw93smho5ueb7azn8uw1j2079isc8b25-ct7cb2rwewvmff
mi69c7gtlzn 192.168.0.8:2377
```



For this join mode	Do this
--------------------	---------

- c. Copy the command output and paste it to a command line on the machine you want to join to the swarm.

```
docker swarm join --token
SWMTKN-1-26hlbe2gk2kozzecvgkw93smho5ueb7azn8uw1j2079isc8b25-ct7cb2rwewvmff
mi69c7gtlzn 192.168.0.8:2377
```

The output of the command is similar to the following:

```
This node joined a swarm as a manager
```

## Starting microservices in Kubernetes

If you deploy the microservice framework and microservices into Kubernetes on a host running a Linux operating system, configure and start microservices as follows.

1. Establish the namespace.

- a. Create a custom namespace.

```
kubectl create namespace <custom-namespace>
```

- b. Check your namespace.

```
kubectl get namespace
```

- c. Change context to the namespace.

```
kubectl config set-context --current --namespace=<namespace name>
```

2. Create secrets and configmaps.

- a. Traverse to the Kubernetes *scripts* directory.

```
cd $TCROOT/container/kubernetes/setup/scripts
```

- b. Run all scripts in the *scripts* directory.

**IMPORTANT:** Address any errors before proceeding to the next step.

3. Deploy the microservice framework and microservices in a Kubernetes cluster.

- a. Traverse to the Kubernetes setup directory.

```
cd $TCROOT/container/kubernetes/setup
```

- b. Run all the setup files (network policies, volumes, persistent volume claims, and persistent volume).

```
kubectl create -f .
```

- c. Traverse to the Kubernetes deployment directory.

```
cd $TCROOT/container/kubernetes/deployment
```

- d. Create all the deployments (deploying microservices).

```
kubectl create -f .
```

## Validate the microservice framework and microservices in a Kubernetes cluster

Validation step	Example command
Check the list of pods.	<code>kubectl get pods -n &lt;namespace&gt;</code>
Check the list of services along with the exposed port.	<code>kubectl get svc -n &lt;namespace&gt;</code>
Check logs of the pods.	<code>kubectl logs &lt;pod-name&gt;</code>
Check environment variables of the pod.	<code>kubectl exec &lt;pod-name&gt; env</code>
Test microservice (in a web browser).	<code>&lt;IP address&gt;:&lt;service dispatcher port&gt;/mps/health/checkhealth</code>

## Manage microservice instances on Windows hosts

If a microservice framework node is running on a Windows host, then you can temporarily change the number of microservice instances on the node.

This method is simple and quick, but temporary. To add new microservices to a node, or to configure the number of instances that start when the **Teamcenter Process Manager** service is restarted, you must run TEM or Deployment Center.

1. On the Windows microservice framework node host, browse to **localhost:8082**.

The default logon credentials are "manager" and "manager". The logon credentials can be changed once logged in to the management interface.

If port 8082 does not work, then check the port setting in the *serverPool.properties* file. The file is stored in the *TC\_ROOT\process\_manager\* directory.

- On the **Agent View** page, under **Administer services**, click the link to the microservice page that you want to manage.

For example, click **id=hello\_service**.

The administration page for the service opens.

- To change the number of microservice instances, at the bottom of the page in the box next to **Change\_Replicas**, enter a new value.
- To apply the new value, click **Change\_Replicas**.

**Note:**

If you use the .NET web tier in a distributed environment in which your server managers are on a separate machines from your microservice nodes, make sure that the **IIS\_IUSRS** ID on the local IIS host has full control on the **net\_webtier\_configuration** directory. **IIS\_IUSRS** is a local user account created by IIS.

## Finding microservice logs on Windows hosts

If a microservice framework node is running on a Windows host, then by default, logs of microservice instances on the node are written to the %TEMP% location. An administrator can **redirect log files**.

To find the files, search for [%TEMP%]\\*-msf.log

Component name	Log file name	Example
Service Dispatcher	service_dispatcher<Instance#>@<PID>-msf.log	service_dispatcher1@2184-msf.log
Service Registry	eureka<Instance#>@<PID>-msf.log	eureka1@2184-msf.log
Microservices	<microservice name><instance#>@<PID>-msf.log	file-repo1@2184-msf.log
Microservice Manager	%TC_ROOT%\microservice_manager\mgr.output	
Active Workspace Gateway	gateway<Instance#>@<PID>-msf.log	gateway1@2184-msf.log

## Redirect microservice logs on Windows hosts

If a microservice framework node is running on a Windows host, you can specify the location for the logs of microservice instances on the node. Logs are written to the location specified by a %TEMP% path variable.

1. Depending on how you run Process Manager, set the value of %TEMP% as desired.

If you run Process Manager as	Do this to set the %TEMP% value
a Windows process	<p>Edit the <i>start_manager.bat</i> file.</p> <p>The file is in the %TC_ROOT%\microservice_manager folder.</p>
a Windows service	<p>Set either a global value or a variable specific to the Process Manager:</p> <ul style="list-style-type: none"> <li>• For a global value, set the system variable %TEMP% path.</li> <li>• For a variable specific to the Process Manager: <ol style="list-style-type: none"> <li>a. At a command prompt, stop and delete the service. <pre>sc stop "Microservice Manager PoolA" sc delete "Microservice Manager PoolA"</pre> </li> <li>b. In an editor, open the file %TC_ROOT%\microservice_manager\install_manager.bat and edit the <b>EXTRA_ENV</b> variable to set the location of <b>TEMP</b> as desired. <div data-bbox="693 1304 823 1339" data-label="Section-Header"> <p><b>Example:</b></p> </div> <pre>rem EXTRA_ENV specifies additional environment variables needed to run as Windows service. set EXTRA_ENV=-env MICRO_ROOT=%MICRO_ROOT% TEMP=C:\mylogs</pre> </li> </ol> </li> </ul>

2. Restart the Process Manager.

To run the Process Manager as	Do this
a Windows process	Run <i>start_manager.bat</i> .
a Windows service	Run <i>install_manager.bat</i> .

3. If you have implemented log aggregation, do the following:
  - a. Stop the **Teamcenter Log Forwarder** Windows service.
  - b. Edit the file `%TC_ROOT%/tc_logging_forwarder/config/sources/msf.conf` to reflect the new path value.

**Example:**

```
#####
# Source Input - MSF Log Files (Eureka, Dispatcher, Microservices)
#####
<source>
  @type tail
  path C:/mylogs/*@*-msf.log
```

- c. Restart the **Teamcenter Log Forwarder** Windows service.

## Configure HTTPS for microservices

### Configure service dispatcher as an HTTPS server

1. To support HTTPS communication from microservices to the service dispatcher, obtain **PKI certificates and keys** suitable for your installed microservices.
2. Use Java keytool to create a Java trust store in .pk12 format and place the private certificate in that store.

The trust store holds the private (server-side) key for SSL. It can contain certificate keys that are either CA-signed or self-signed.

While both .pk12 and .jks trust store file formats are accepted, .pk12 format has greater microservice compatibility and is recommended.

3. Using the Teamcenter installer, ensure the following:
  - The machine name where the service dispatcher is being deployed matches the name in the signed certificate.  
For example, if the certificate is for **sd.example.com**, then the service dispatcher host machine name must be **sd.example.com**.
  - **https** is the selected protocol for service dispatcher.

4. Place the trust store in the proper directory according to the node host operation system.

OS	Trust store location
Linux - Docker	In the <code>TC_ROOT\container\secrets</code> directory of your master microservice node.
Windows	In the secrets folder at <code>TC_ROOT\microservices\secrets</code> of each microservice node where a Service Dispatcher is running.

5. Depending on the microservice node host OS, edit the `service_dispatcher.json` file (Windows), or `TC_ROOT/container/tc_microservice_framework.yml` file (Linux - Docker), or `TC_ROOT/container/kubernetes/deployment/tc_microservice_framework.yml` file (Linux - Kubernetes) to extend the ARGV value for the service dispatcher to provide values for the following:

For this argument	Do this
protocol	Use <b>https</b> .
keystore	Specify the location of the trust store ( <b>pkcs12</b> or <b>jks</b> ) holding the private server-side key (certificate).
kspassword	Specify the password for the trust store.
keystoreType (if not jks)	Specify <b>PKCS12</b> if not using <b>JKS</b> .

#### **service\_dispatcher.json (Windows host)**

Add security properties to the file `installation_path\microservices\services_config\service_dispatcher.json`.

Example:

Original file content, including the ARGS setting:

```
{
  "service_dispatcher":{
    "image":"service_dispatcher-1.2.0",
    "environment":[
      "ARGS=-Dport=9090 -Deureka.serviceUrl.default=http://msnode1:8080/eureka/v2"
    ]
  }
}
```

Content after adding security properties:

```
{
  "service_dispatcher":{
    "image":"service_dispatcher-1.2.0",
    "environment":[
      "ARGS=-Dport=9090 -Dprotocol=https -Dkeystore=path_to_trust_store_file
      -Dkspassword=trust_store_password -DkeystoreType=pkcs12
      -Deureka.serviceUrl.default=
      http://msnode1:8080/eureka/v2"
    ]
  }
}
```

### tc\_microservice\_framework.yml (Docker on Linux host)

In the Service Dispatcher (Eureka) Docker container, add security properties to the file *installation-path/container/tc\_microservice\_framework.yml*. You must also add the keystore file as a secret.

Example:

File content before adding security properties and keystore file:

```
service_dispatcher:
  image: siemens/teamcenter/service_dispatcher:1.2.0
  deploy:
    replicas: 1
  environment:
    - ARGS=-Dport=9090 -Deureka.serviceUrl.default=http://eureka:8080/eureka/v2/
  ports:
    - 9090
  depends_on:
    - eureka
```

File after adding security properties:

```
service_dispatcher:
  image: siemens/teamcenter/service_dispatcher:1.2.0
  deploy:
    replicas: 1
```

```

environment:
  - ARGS=-Dport=9090 -Dprotocol=https -Dkeystore=my_trust_store_file
    -Dkspassword=trust_store_password -DkeystoreType=pkcs12
-Deureka.serviceUrl.default=
  http://eureka:8080/eureka/v2/
ports:
  - 9090
depends_on:
  - eureka
secrets:
  - <keystore_file>

secrets:
  <keystore_file>:
    file: ./secrets/my_trust_store_file

```

6. Update microservice configuration files as appropriate for the operating system. In the following examples, **service\_dispatcher**, **host1**, and **host2** are placeholders for what is signed in the certificate, which is typically the fully qualified domain name.

For this OS	Do this
Microsoft Windows	<p>a. Update <i>services_config/*.json</i> files to point to the service dispatcher URL using HTTPS protocol. Not all json files will have references to the dispatcher.</p>
	<p>Example:</p> <p>DSP=http://host1:9090,http://host2:9090</p> <p>becomes</p> <p>DSP=https://host1.domain.com:9090,https://host2.domain.com:9090</p>
	<p>b. To get the microservices to pick up the new changes, restart the <b>Teamcenter Process Manager</b> process.</p>
Docker on Linux	<p>a. Update <i>*.yaml</i> files to point to the service dispatcher URL using HTTPS protocol. Note that not all .yaml files will have references to the dispatcher.</p>

Example:

ENDPOINT\_SERVICE\_DISPATCHER=http://host1:9090,http://host2:9090

becomes

ENDPOINT\_SERVICE\_DISPATCHER=https://host1.domain.com:9090,https://host2.domain.com:9090



For this  
OS

Do this

- b. To deploy the changes to the YAML file, redeploy the YAML files to the stack.

7. If the framework nodes are hosted on Windows operating system, add the certificate to the list of certificates the Microservices trust. The procedure for this varies based on the language in which the microservice is written.

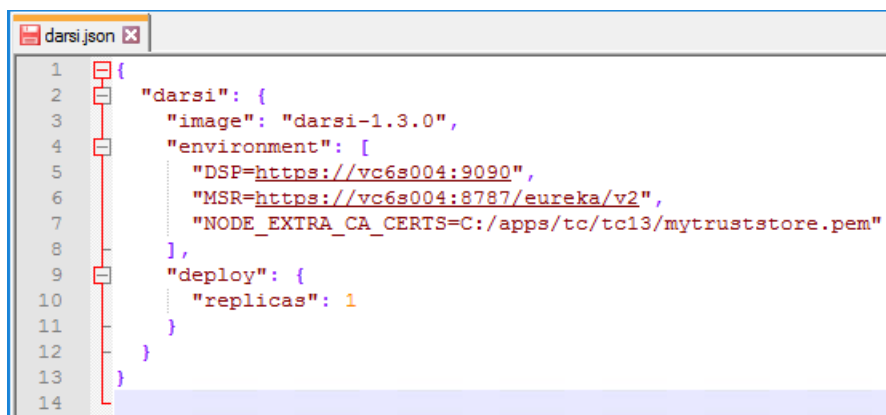
### Javascript/Typescript (NodeJS) based Microservices

For Javascript/Typescript based microservices that are using NodeJS, an environment variable must be set which points to the location of the signed certificate. The certificate must be in PEM format, and must not have been generated using DSA encryption.

Please see NodeJS documentation for additional information about this variable.

Edit the file `%TC_ROOT%\microservices\services_config<microservice>.json`.

- a. Add the variable into the **environment** section.  
Example:



- b. Restart the process manager.

### Java based Microservices

For Java based microservices, the JVM must have the CA added to its list of trusted certificates.

- a. Ensure that the following two arguments are passed to the JVM:

```
-Djavax.net.ssl.trustStorePassword=
```

```
-Djavax.net.ssl.trustStore=<path to truststore file in .jks
format>
```

The method for doing this for Java based microservices depends on their implementation. Generally, the **.json** file will have an **ARGS** variable which may be appended to. In lieu of

that, the microservice's **start\_service.bat** file may be edited to add these parameters to the JVM.

Some microservices support editing the **%TC\_ROOT%\microservices\services\_config\<microservice>.json** file to alter the JVM arguments.

Others require that you modify the corresponding **%TC\_ROOT%\microservices\<microservice>\start\_service.bat** script to add the JVM arguments.

- b. Restart the process manager.

### C# based Microservices

- a. For C# based microservices, if the trust store file is in .jks format, convert the .jks file to .pk12.

For example, to list entries in a keystore file named **keystore2.jks**, run the command:

```
keytool -list -keystore ./keystore2.jks
```

To convert a keystore file named **keystore2.jks** to a .pk12 file using the key **mykey** and the password **testKeyStorepw**, run the command:

```
keytool -importkeystore -srckeystore [./keystore2.jks] -
destkeystore ./keystore2.pk12 -srcstoretype JKS -deststoretype
PKCS12 -srcstorepass testKeyStorepw -deststorepass testKeyStorepw
-srcalias mykey -destalias mykey -srckeypass testKeyStorepw -
destkeypass testKeyStorepw -noprompt
```

- b. Double-click the .pk12 file to install it as a trusted certificate.

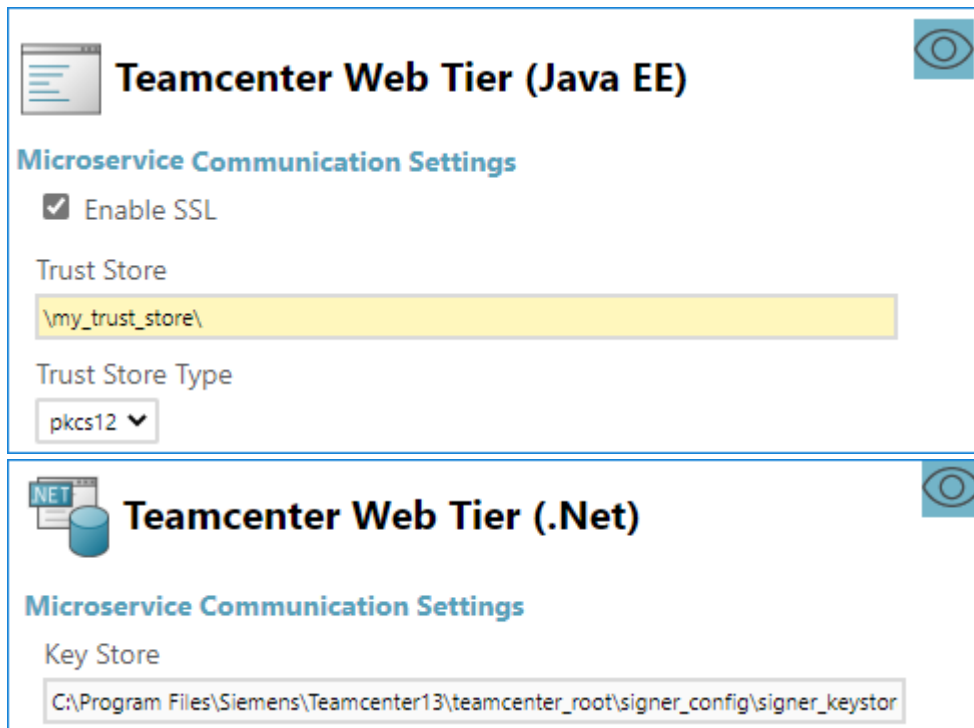
## Configure the web tier for HTTPS with the service dispatcher

1. Configure the Teamcenter web tier as appropriate for the web tier type.

For this web tier type	Do this
Microsoft .NET	<ol style="list-style-type: none"> <li>a. In <i>ApplicationConfiguration.xml</i>, modify the parameter <b>microserviceAddress</b>. <div style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <p>Example:</p> <pre>&lt;param name="microserviceAddress" value="http:// service_dispatcher:9090"/&gt;</pre> <p>becomes</p> <pre>&lt;param name="microserviceAddress" value="https:// service_dispatcher.domain.com:9090"/&gt;</pre> </div> </li> </ol>

For this web tier type	Do this
	b. Restart the web server.
Java EE	<p>a. In the Teamcenter Web Application Manager (insweb), modify the Context Parameters:</p> <p style="margin-left: 40px;">MICROSERVICE_ADDRESS</p> <p style="margin-left: 40px;">MICROSERVICE_TRUSTSTORE</p> <p style="margin-left: 40px;">MICROSERVICE_TRUSTSTORE_TYPE</p> <p style="margin-left: 40px;">MICROSERVICE_HOST_VERIFY_MODE</p> <div style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <p>Example:</p> <pre>&lt;param-name&gt;MICROSERVICE_ADDRESS&lt;/param-name&gt; &lt;param-value&gt;http://service_dispatcher:9090&lt;/param-value&gt;</pre> <p>becomes</p> <pre>&lt;param-name&gt;MICROSERVICE_ADDRESS&lt;/param-name&gt; &lt;param-value&gt;https:// service_dispatcher.domain.com:9090&lt;/param-value&gt;</pre> </div> <p>b. Redeploy the tc.war.</p>

2. Public keys for CA-issued certificates are already available in the web tier. If you are using self-signed certificates, then do the following:
  - a. Put the certificate public key in a trust store for the web tier.
  - b. If the trust store is not already located as configured, then put the trust store in the location that was configured in the Teamcenter installer web tier **Microservice Communications Settings**.



The image shows two configuration panels for Teamcenter Web Tiers. The top panel is for 'Teamcenter Web Tier (Java EE)' and the bottom panel is for 'Teamcenter Web Tier (.Net)'. Both panels have a 'Microservice Communication Settings' section.

**Teamcenter Web Tier (Java EE)**

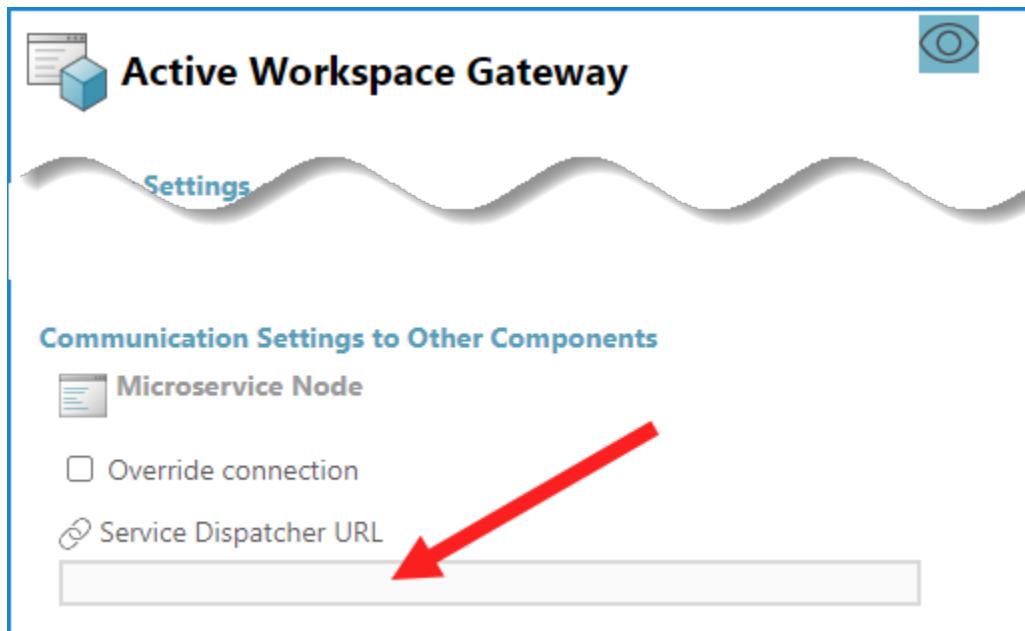
- ☒ Enable SSL
- Trust Store:
- Trust Store Type:

**Teamcenter Web Tier (.Net)**

- Key Store:

### Configure Active Workspace Gateway for HTTPS with the service dispatcher

To configure Active Workspace Gateway for HTTPS with the service dispatcher, in the component configuration panel for **Active Workspace Gateway**, ensure that the **Service Dispatcher URL** uses **https** prefix.



The image shows the 'Active Workspace Gateway' configuration panel. It has a 'Settings' tab and a 'Communication Settings to Other Components' section. Under 'Microservice Node', there is an 'Override connection' checkbox and a 'Service Dispatcher URL' field. A red arrow points to the 'Service Dispatcher URL' field.

**Active Workspace Gateway**

**Settings**

**Communication Settings to Other Components**

**Microservice Node**

- ☐ Override connection
-

If the Certificate Authority for the service dispatcher certificate is known to the Active Workspace Gateway, no further action is required.

If the service dispatcher certificate is a self-signed certificate or is from a certificate authority not known to the Active Workspace Gateway host operating system, then on the Active Workspace Gateway host set the **NODE\_EXTRA\_CA\_CERTS** environment variable ([https://nodejs.org/api/cli.html#cli\\_node\\_extra\\_ca\\_certs\\_file](https://nodejs.org/api/cli.html#cli_node_extra_ca_certs_file)) to point to the required certificate.

## PKI key requirements for microservices

If you want to use HTTPS protocol for sending data in the microservice framework, a certificate and matched pair of encryption keys is required:

### public key

A cryptographic key that can be obtained and used by anyone to encrypt messages intended for a particular recipient, such that the encrypted messages can be deciphered only by using a second key (the private key) that is known only to the recipient.

A client uses the public key when establishing a secure connection to a server. When the client initiates a request to the server, the server responds with its certificate, which contains the public key, and which is signed either by a certificate authority (CA), or is a self-signed certificate generated by some tool, rather than purchased from a CA.

In order for the client to trust the certificate sent by the server, and consequently the server, the certificate must be signed by an authority (signer) that the client trusts. CA signatures are generally trusted. Signatures in a self-signed certificate must be from a signer included in a client's trust store in order for the client to trust the certificate.

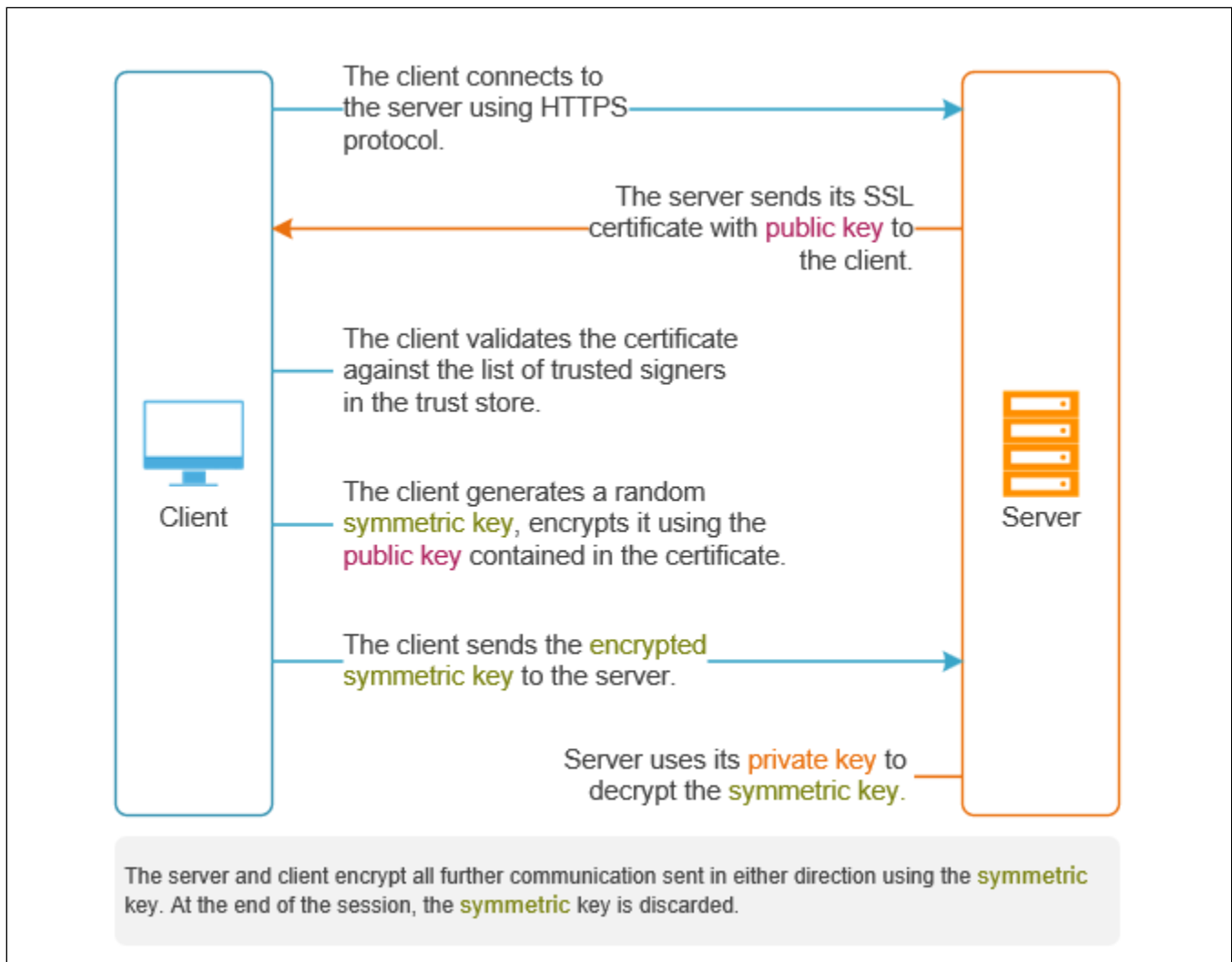
### private key

A cryptographic key that is used by a server to decrypt the data it receives from the client, which encrypted the data using the corresponding public key. A private key is also known as a secret key. The private key should be kept secret and shared only with the key's generator, thereby making it secure.

Standard authentication is unidirectional. For example, a request from a microservice to the service dispatcher can eventually result in encryption of data traveling to the service dispatcher. If authentication of communication initiated by the service dispatcher to a microservice is desired, the microservice must be configured to serve its own certificate, and the service dispatcher must include the certificate signer in its trust store.

A single certificate and its public/private keys can be used by all microservices. A server certificate signed by a certificate authority (CA) can be purchased from a CA, and is recommended. Alternatively, cryptographic tools such as OpenSSL can be used to create a self-signed certificate and its keys.

Note that microservices developed in Typescript language are incompatible with certificates generated using DSA encryption.



Sequence for one-way SSL handshake

## High availability for microservices

In a distributed Teamcenter production environment, ensure high availability by configuring redundant microservice node servers and service instances.

### Capacity

With the many variables affecting a Teamcenter environment, no simple formula exists that can prescribe the precise combination of microservice nodes and microservice instances. As with all server-side deployments, monitor the consumption of CPU and memory on each microservice node. If you observe resource contention, you can increase resources for microservice execution by deploying additional microservice nodes and services running on additional hardware.

## Failover

### Windows

Achieving failover capability on Windows requires that a service registry, a service dispatcher, and instances of all microservices must each be running on at least two nodes. By default, an instance of the service registry and service dispatcher run on the master node; additional instances can be running on any worker nodes. When installing microservice nodes through TEM, be sure to list all instances of the service registry and the service dispatcher.

### Docker

Achieving failover capability with Docker on Linux requires that an odd number of nodes be joined to the swarm as managers, typically three or five. This helps the Docker swarm effectively manage the swarm by majority vote. Any number of nodes can be joined to the swarm as workers.

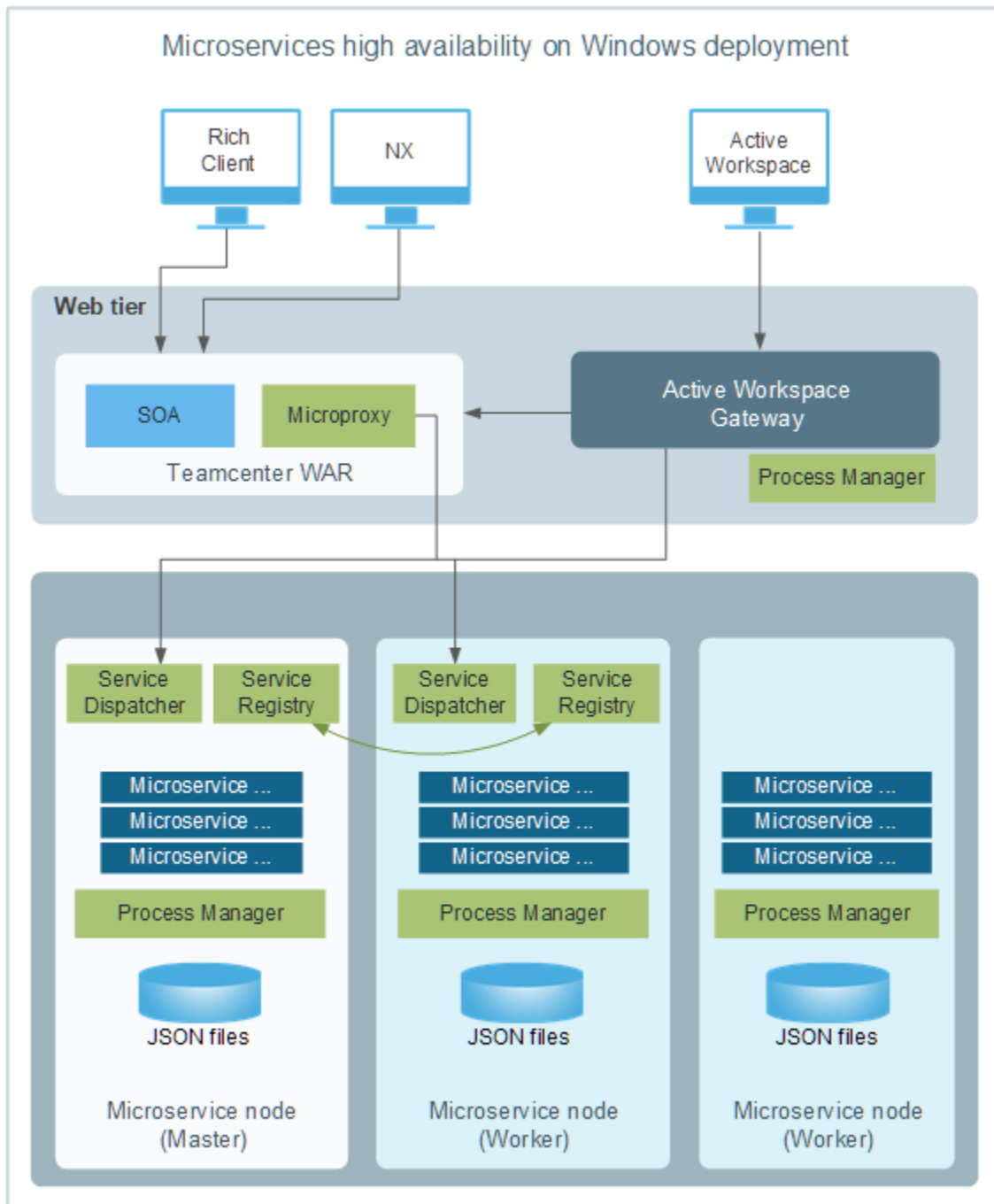
### Kubernetes

<b>Control Plane</b>	Follow the vendor documentation. If using a cloud provider, the provider typically provides a Control Plane with failover.
<b>Worker nodes</b>	To avoid a single point of failure, in on-site deployments implement at least two Kubernetes worker nodes. Ensure that these nodes are allocated on different physical hardware. Allocate at least two replicas of every component to avoid a single point of failure. For worker nodes in cloud deployments, to avoid location-specific outages, ensure that the nodes are spread across different failure zones (such as AWS Availability Zones).

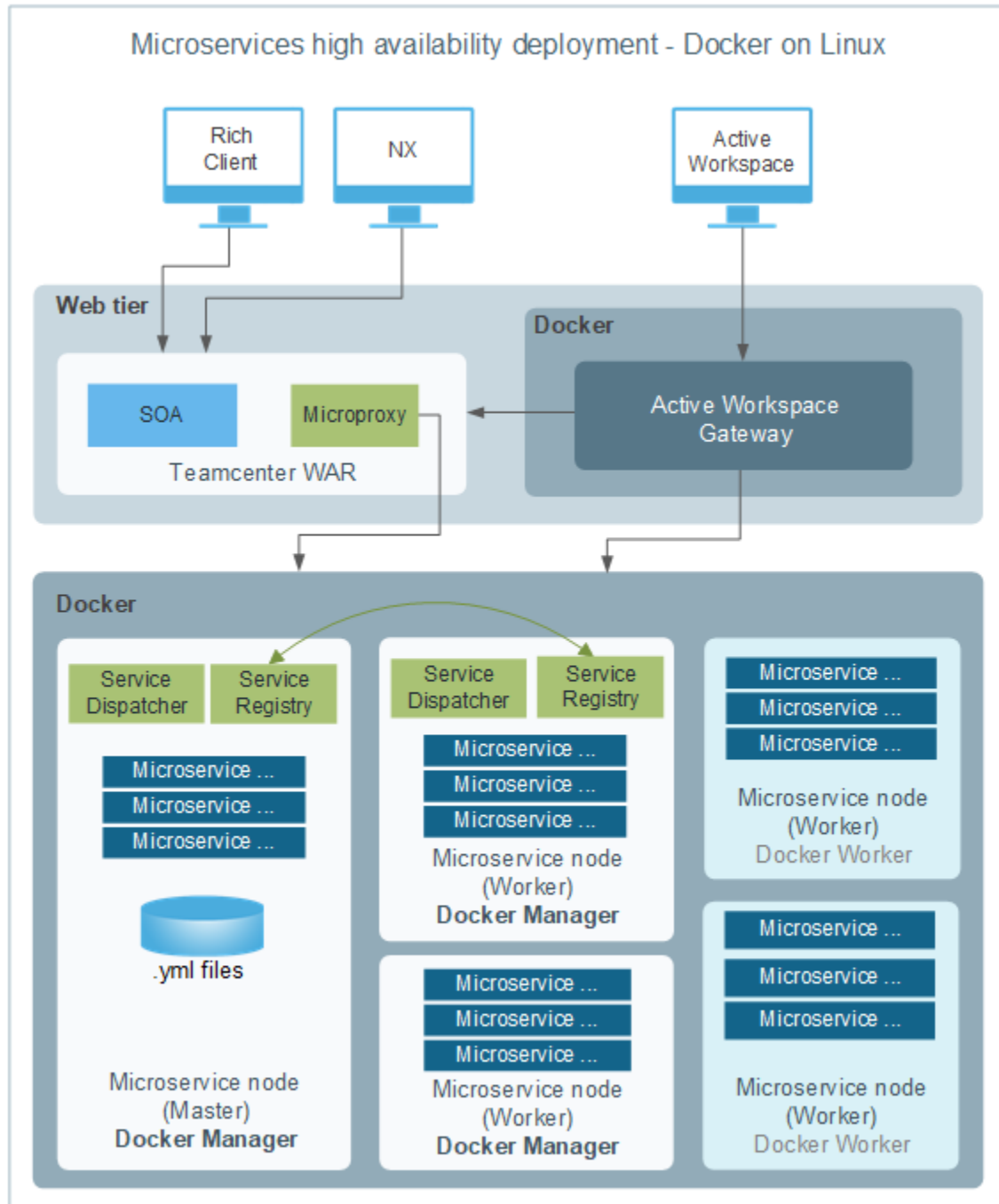
If possible, test for worker node failure conditions and validate that client requests are handled via service load balancing. Ensure desired scale once the worker nodes are recovered.

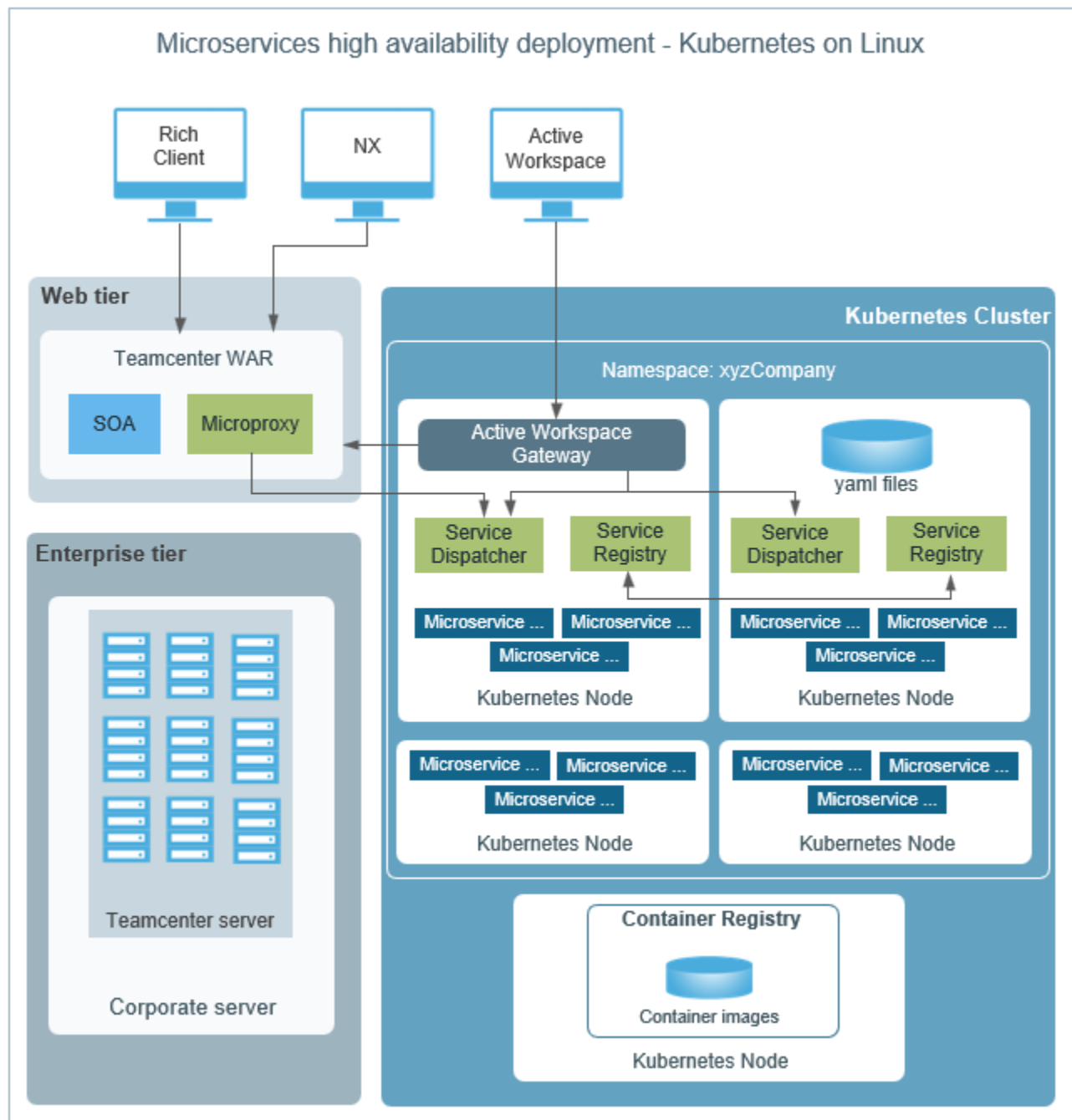
For backup options, consult the vendor documentation.

## Example microservice deployment topologies for high availability









## Troubleshooting microservices

Problem/error message	Possible cause	Solution
404 error for a microservice request with the Service Dispatcher logging a	In a deployment with a load balancer configured, due to the addition of large cookies by the load balancer, some requests	Create a CUSTOM_REQUEST_BUFFER_SIZE environment variable and set its value higher than the default microservice service dispatcher request buffer size of 8192 (8 KB), and then restart the service dispatcher.

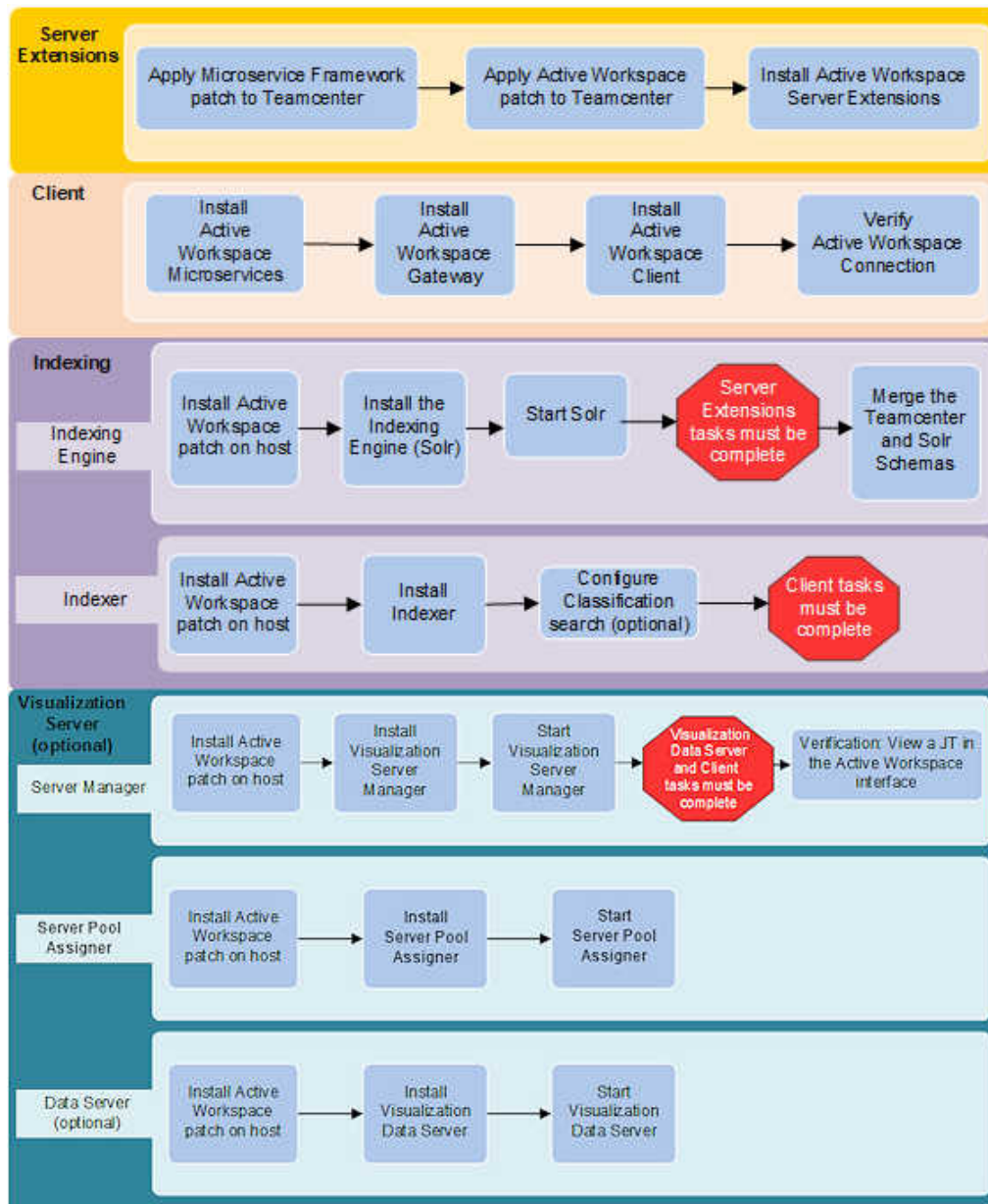
Problem/error message	Possible cause	Solution
message that the HTTP header is too large.	exceed the limit for the header size.	



# 3. Installing Active Workspace

## Active Workspace components installation overview

The primary Active Workspace components are Server Extensions, Client, Indexing, and Visualization Server. Each of these require their own sequences of installation tasks. In the installation paths shown, octagons indicate verification tasks where you must ensure all previous steps are complete before you continue.

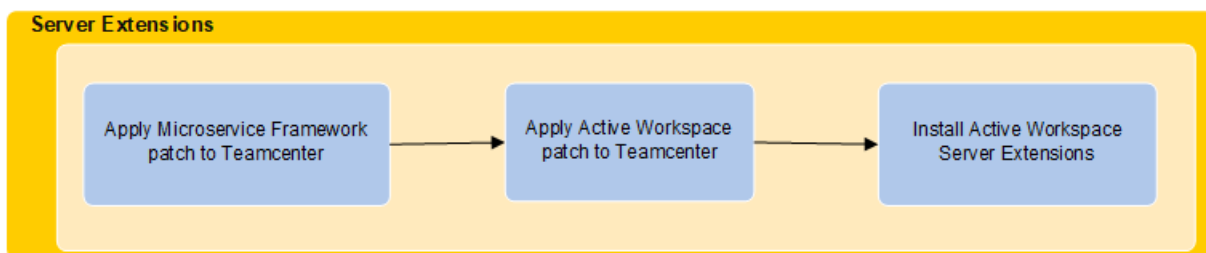


Whether you use TEM or Deployment Center to install Active Workspace components, you may need to enter or verify the following values from your Teamcenter environment:

Teamcenter four-tier environment configuration	
Parameter	Your value
Teamcenter machine name and credentials	
Teamcenter installation directory ( <i>TC_ROOT</i> )	
Teamcenter data directory ( <i>TC_DATA</i> )	
Teamcenter four-tier URL ( <b>http://host:port/tc</b> )	
Teamcenter FMS machine name and bootstrap URLs ( <b>http://host:4544</b> )	
Licensing server machine name and port (28000 by default)	
Database system machine and credentials	
Operating system credentials for component servers	
Location of security certificates and keys	
Microservice node information if already installed	

## Active Workspace Server Extensions

### Server Extensions overview



Active Workspace Server Extensions add binaries to *TC\_ROOT* and add data model changes to the database and *TC\_DATA*. These additions enable Active Workspace functionality. Server Extensions must be installed on each Teamcenter server that includes Teamcenter Foundation, such as corporate servers, volume servers, and server manager hosts.

Make sure your Server Extensions host has a supported 64-bit operating system and a 64-bit JDK.

Make sure your Teamcenter environment has the following:

- Supported version of Teamcenter  
Active Workspace 5.2 supports the latest versions of Teamcenter or Teamcenter Rapid Start, and some earlier versions. If you do not use the latest version of Teamcenter or Teamcenter Rapid Start, see the Hardware and Software Certifications knowledge base article on Support Center.
- Corporate server
- Server manager
- Teamcenter web tier (Java EE or .NET)

Installing Server Extensions requires the following values from your Teamcenter environment. Be prepared to enter these values into the installation tool (TEM or Deployment Center) during Active Workspace installation.

Values required before installing Server Extensions	
Parameter	Your value
Teamcenter administrative user password	
Database system user login/password	
Teamcenter home directory ( <i>TC_ROOT</i> )	
Teamcenter data directory ( <i>TC_DATA</i> )	
Indexing Engine user name and password	
Search engine URL (Example: <b>http://host:8983/solr</b> )	

## Install Server Extensions features

Install Server Extensions on every host that has Teamcenter Foundation installed.

1. Stop all Teamcenter services except FSC services.
2. Launch the installed Teamcenter Environment Manager (TEM) for the corporate server.
3. In the **Maintenance** panel, select **Configuration Manager**.
4. In the **Configuration Maintenance** panel, select **Perform maintenance on an existing configuration**.
5. In the **Old Configuration** panel, select the corporate server configuration.
6. In the **Feature Maintenance** panel, select **Add/Remove Features**.
7. In the **Features** panel, select Active Workspace Server Extensions features:
  - a. Expand **Base Install**→**Active Workspace**→**Server Extensions**.
  - b. Select the **Active Workspace** Server Extensions feature. (This feature is required.)
  - c. Select any **additional Server Extensions features** you want to include in your Active Workspace deployment.

Some features may display additional panels in TEM that are not described in this procedure. See the help buttons in those panels for information.

Note:

- To search for a feature by name, type the name or a partial name in the search box, and then click the search icon.
- Some Server Extensions features depend on other Server Extensions features.

8. In the **Teamcenter Administrative User** panel, type the user's password.
9. In the **Indexing Engine User** panel, type the Solr administrator's user name and password.

The Solr administrator's user name and password are defined when you install the Indexing Engine (Solr).

If Indexing Engine is not installed yet, record the Solr administrator user name and password and make sure to use these credentials when you install Indexing Engine. These credentials *must* match



for Indexing Engine, the Indexer, Server Extensions, and the Active Content Structure Translator (if used).

10. In the **Active Workspace Server Extensions Settings** panel, type the Solr search engine URL. The format is:

**`http://host:port/solr`**

*host* is the machine designated to run Solr. This is the machine on which Indexing Engine is to be installed.

*port* is the port value used by Solr. The default is **8983**.

If your environment uses multiple Solr search engines for failover, select **Use additional search engine URLs**, and then type the URLs in the **Search Engine URL List**.

11. In the **Database User** panel, type the password for the Teamcenter database user.
12. In the **Database Template Summary** panel, review the list of templates to be applied to the Teamcenter database. The list varies depending on the features selected for installation.
13. In the **Confirmation** panel, click **Start** to begin the installation.

The **TcFTSIndexer** process requires database triggers that enable database access for the Indexer to detect changes to the database when performing run-time (synchronous) indexing. If TEM cannot install these indexing triggers automatically, it informs you to **install the database triggers manually**.

14. When installation is complete, close TEM.

## Install database triggers manually

The **TcFTSIndexer** process requires database triggers that enable database access for the Indexer to detect additions, modifications, and deletions to the database when performing run-time (synchronous) indexing.

In most cases, TEM installs these indexing triggers automatically. However, if you do not provide the necessary values (logon values for the database user and the system user) to install them during your TEM session, TEM displays a **TODO** message about enabling database access to Teamcenter. If this happens, install the database triggers manually using the appropriate procedure for your database:

### Install database triggers in Oracle

1. To grant the **create trigger** privilege to the Oracle user that owns the Teamcenter database, perform the following steps:

- a. Open a command prompt.

- b. Type:

**sqlplus system/password**

- c. Type:

**grant Create trigger to Tc-Oracle-user identified by password;**

- d. Type:

**exit**

2. Create the trigger:

- a. In the command prompt, type:

**sqlplus Tc-Oracle-user/password**

- b. Type:

**@Teamcenter-installation-media\tc\install\sitecons\sitecons\_install\_triggers\_oracle.sql**

#### Install database triggers in Microsoft SQL Server

1. Open Microsoft SQL Server Management Studio.
2. Complete the **Connect to Server** dialog box:
  - a. In the **Server name** box, select the host on which Microsoft SQL Server is installed.
  - b. In the **Authentication** box, select **SQL Server Authentication**.
  - c. In the **Login** box, type the database administrative user name.
  - d. In the **Password** box, type the database administrative user password.
  - e. Click **Connect**.
3. In the **Object Explorer** panel of the **Microsoft SQL Server Management Studio** dialog box, expand the **Databases** tree and select the Teamcenter database name, for example, **tc**.
4. From the menu bar, choose **File→Open→File**.
5. In the **Open File** dialog box, navigate to the software kit for the Teamcenter major release.

In the *Tc-software-path\tc\install\sitecons* directory, select **sitecons\_install\_triggers\_mssql.sql**.

Microsoft SQL Server Management Studio opens the selected file.

6. Click **Query→Execute**.

The query installs the database triggers.

7. Verify that the query completed with no errors.
8. Close the Microsoft SQL Server Management Studio.

## Installing database triggers from the command line

If Microsoft SQL Server Management Studio is not installed on your host, you can install the database triggers from a command line. Type the following command in a Windows command prompt:

```
sqlcmd -H host -d database -U user -P password -i path
\sitecons_install_tables_and_triggers_mssql.sql
```

Replace:

- *host* with the database server host name.
- *database* with the Teamcenter database name.
- *user* with the database user name.
- *password* with the database user password.
- *path* with the path to the **sitecons\_install\_triggers\_mssql.sql** file.

For example:

```
sqlcmd -H myhost -d TcDB -U dbUser -P dbPassword -i
C:\software\tc\install\sitecons\sitecons_install_triggers_mssql.sql
```

To verify the triggers installed successfully, log into Microsoft SQL Server and type the following commands in an SQL prompt:

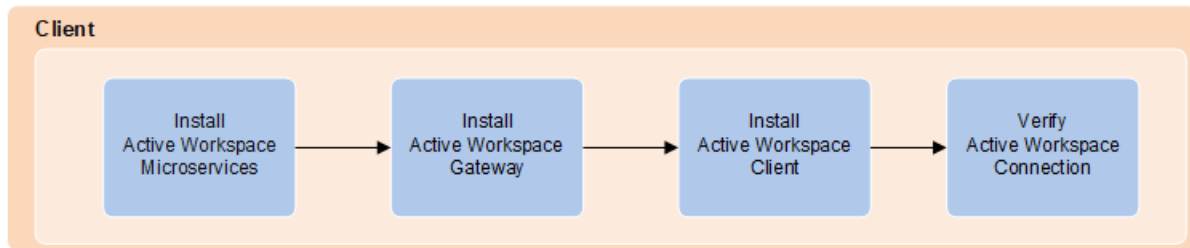
```
1> Select name,is_disabled from sys.triggers2> Go
```

If the installation succeeds, Microsoft SQL Server displays a table similar to the following showing that the database triggers are not disabled:

name	is_disabled
fast_sync_add_trigger	0
fast_sync_delete_trigger	0
(2 rows affected)	

## Active Workspace client components

### Active Workspace client overview



Active Workspace client functionality is delivered through *Active Workspace Gateway*, a Node.js implementation that routes all requests for static content such as HTML, CSS, Javascript, JSON, and other types, and dynamic content such as API routing to back-end services and microservices for SOA, FMS, Visualization, GraphQL, and others.

Active Workspace Gateway employs Teamcenter microservices for some request types. For example, Active Workspace client content is stored in an Active Workspace file repository and managed by the File Repository microservice.

Deploying the Active Workspace client requires installing the following applications using TEM or Deployment Center. You must deploy the applications in the order listed. Review each of the procedures before you begin your installation.

#### Active Workspace microservices

#### Active Workspace Gateway

#### Active Workspace client

Make sure your Active Workspace Gateway host has a supported 64-bit operating system with a minimum of 2 GB of free memory.

Be prepared to enter the following values from your Teamcenter environment during installation of Active Workspace client components:

Values required before installing the Active Workspace Gateway	
Parameter	Your value
Teamcenter web tier URL	
FSC Bootstrap URLs ( <b>http://host:4544</b> )	

For information about configuring security for this client, see Security configuration tasks in the Active Workspace help collection.

## Install Active Workspace microservices

The **microservice framework** must be installed before you begin these steps. You can add Active Workspace microservices to an existing microservice node or install the microservices and Microservice Framework at the same time.

1. Launch your preferred installation tool (TEM or Deployment Center) to begin installing Active Workspace microservices.
2. Choose the software.

**TEM** Add the Active Workspace 5.2 software kit and the Microservice Framework software kit to the **Media Locations** panel.

**Deployment Center** Add the Active Workspace 5.2 software kit and the Microservice Framework software kit to **Software Repositories**. Whether you are installing or updating an environment, ensure all applicable software kits are present.

Check the **Selected Software** list, including the base and minor release versions of Teamcenter software. You may install microservices as part of an installation or an upgrade.

In **Options**, choose the **Distributed** environment type and the architecture.

3. In TEM, add the following microservices in the **Features** panel:

- **File Repository Service**
- **Teamcenter GraphQL Service** (optional)
- **Declarative Artifact Service** (optional)

In Deployment Center, **Selected Applications** automatically includes these microservices. You can view them in the **Microservice Node** component configuration.

4. In TEM, configure the microservices with the required values.

In Deployment Center, configure the **Microservice Properties** on a **Microservice Node** component.

- **File Repository Microservice**

Value	Action
<b>File Repository Storage Location</b>	Type a location for the file repository to be used by the Active Workspace Gateway. The path must exist on the machine that hosts the microservice node. For example: <b>c:\tc\file_repository</b>  The file repository stores Active Workspace content.
<b>User ID</b>	(Linux only) Type the user ID of the user installing the File Repository Microservice.
<b>Group ID</b>	(Linux only) Type the Group ID of the user installing the File Repository Microservice.

- **GraphQL Microservice**

Value	Description
<b>Teamcenter Web Tier URL</b>	In TEM, type the path to the deployed Teamcenter web tier, for example:  <b>http://host:port/tc</b>  In Deployment Center, the <b>Microservice Node</b> locates the URL from the <b>Teamcenter Web Tier</b> component automatically.

**Note:**

On Linux systems, microservices Worker Nodes must contain the same installed microservices as the Master Node.

5. In TEM, start the microservices after you complete the microservices installation, using the appropriate method for your operating system.

In Deployment Center, continue with the **Active Workspace Gateway installation**.

## Install Active Workspace Gateway

Active Workspace Gateway requires the keystore ZIP file (**keys.zip**) from the microservice master node. Before you install Active Workspace Gateway, copy the **keys.zip** file from the **jwt\_config\_tool** directory under **TC\_ROOT** on the microservice master node host to a directory on the Active Workspace Gateway host.

You can install Active Workspace Gateway in a new or an existing Teamcenter environment. Use the same installation tool you use to **install Active Workspace microservices**:

## Install the Active Workspace Gateway using TEM

## Install the Active Workspace Gateway using Deployment Center

Depending on your network configuration, you may choose to **restrict access to the JMX interface** that Active Workspace Gateway uses.

### Install the Active Workspace Gateway using TEM

1. If you are installing Active Workspace Gateway in a *new* configuration (on a machine with *no* existing Teamcenter environment), skip to step **2**.

If you are adding Active Workspace Gateway to an *existing* configuration:

- a. **Add Active Workspace and Microservice Framework** to the Teamcenter host.
  - b. Launch the installed TEM in maintenance mode.
  - c. Skip to step **7**.
2. Make sure you have access to the following software kits:
    - Teamcenter major release
    - Teamcenter minor release
    - Active Workspace 5.2
    - Microservices Framework
  3. Launch TEM from the Teamcenter minor release kit.
 

On Windows hosts, right click the **tem.bat** program icon and choose **Run as administrator**.
  4. Proceed to the **Install/Upgrade Options** panel and then click **Install**.
  5. In the **Media Locations** panel, enter paths to the software kits.
    - In the **Original Media Location** box, enter the path to the software kit for the major version of Teamcenter.
    - In the **Update Location** box, enter the paths to Active Workspace 5.2 and Microservices Framework software kits.

The path to the Teamcenter minor version software kit is already included in the list of update locations.
  6. In the **Configuration** panel, enter an **ID** and a **Description** for the configuration.

7. In the **Features** panel, under **Base Install**→**Active Workspace**→**Client**, select **Active Workspace Gateway**.

If you are installing a new configuration, enter an installation location in the **Installation Directory** box.

8. In the **Gateway Settings** panel, enter the required values:

Value	Description
<b>Teamcenter Web Tier URL</b>	Enter the URL for the Teamcenter web application, in the following form:  <b><code>http://host:port/tc</code></b>
<b>Visualization Assigner URL</b>	Enter the URL to the Visualization Server Assigner, in the following form:  <b><code>http://host:port</code></b>
<b>Easy Plan URL</b>	Enter the URL to the Easy Plan application, if installed. The URL should be of the following form:  <b><code>http://host:port/application-name</code></b>
<b>NGP URL</b>	Enter the URL to the Next Generation Planning (NGP) application, if installed. The URL should be of the following form:  <b><code>http://host:port/application-name</code></b>
<b>Service Dispatcher URLs</b>	Enter the URL to the service dispatcher, based on the service dispatcher <i>host</i> and <i>port</i> you used when you installed the microservice node:  <b><code>http://host:port</code></b>  The default <i>port</i> is <b>9090</b> .  If you installed multiple nodes, you may enter multiple URLs as a comma-separated list. Use fully-qualified domain names or IP addresses in URLs.
<b>Gateway Service Port</b>	Enter the port used by the Active Workspace Gateway. The default value is <b>3000</b> .  The URL to the Active Workspace client interface is based on this port.

Under **FMS Settings**, enter the following settings for accessing FMS volumes:



Value	Description
<b>Use as Bootstrap URLs</b>	<p>The Active Workspace client uses FMS to download and upload files. You define the FSC servers that are used by selecting either <b>Use as Bootstrap URLs</b> or <b>Use Assigned FSC URLs</b>.</p> <p>To use bootstrap URLs, select this option and fill in the <b>Bootstrap URLs</b> and <b>Bootstrap Client IP</b> boxes.</p> <p>On Linux hosts, if you select <b>Use as Bootstrap URLs</b>, you need to ensure the client map is configured correctly.</p>
<b>Bootstrap URLs</b>	<p>Enter a comma-separated list of URLs to one or more existing FMS server caches (FSCs).</p> <p>URLs must be of the form:</p> <p style="text-align: center;"><b><code>http://host:port</code></b></p> <p>By default, the IP address from the HTTP connection of the requestor is used unless a <b>Bootstrap Client IP</b> value is provided. (The client/requestor is the host on which Active Workspace Gateway is deployed.)</p>
<b>Bootstrap Client IP</b>	<p>Enter the FMS bootstrap client IP address to be used for the assignment.</p> <p>On Linux hosts, enter the internal IP address of the Active Workspace Gateway machine.</p>
<b>Use Assigned FSC URLs</b>	<p>Specifies whether you want to assign FSC servers. Select this only if you want explicit control of the FSCs used.</p> <p>To use assigned FSCs, select this option and fill in the <b>Assigned FSC URLs</b> box.</p>
<b>Assigned FSC URLs</b>	<p>Enter a comma-separated list of one or more assigned FSC URL values.</p> <p>The URL values entered are directly used for file operations. This allows you to declare the FSC servers that should be used.</p>

For information about other values in the **Gateway Settings** panel, click the help button .

- In the **Gateway Security Settings** panel, enter the required values:

Value	Description
<b>Keystore Zip File</b>	<p>Enter the location of the keystore zip file (<b>keys.zip</b>) generated when you installed the microservice master node.</p> <p>The keystore file is generated in the <b>jwt_config_tool</b> directory under <b>TC_ROOT</b> on the microservice node host. For security, copy the <b>keys.zip</b> file to a directory on the Active Workspace Gateway host and specify that location here.</p>

Under **Enable TcSS Support**, enter values for configuring Security Services.

<b>Enable TcSS Support</b>	Select this check box if you want to enable Security Services support in Active Workspace.
<b>TcSS Application ID</b>	Type the Security Services application ID.
<b>TcSS Login URL</b>	<p>Type the logon URL for the Security Services application.</p> <p>For details about configuring Security Services, see <i>Security Services Installation/Customization</i> in Teamcenter documentation.</p> <p>When you configure Active Workspace for Security Services, be sure to only install the language packs for the Security Services that</p>

Value	Description
	Active Workspace supports.  If Active Workspace is deployed on a different URL, you must configure Security Services with multiple application IDs.
Under <b>Security Key Settings</b> , enter security key values for Teamcenter.	
<b>Security Certificate</b>	Enter the path to the security certificate.
<b>Security Key</b>	Enter the path to the security key file.

10. In the **Confirmation** panel, click **Start** to begin the installation.
11. When the installation is complete, close TEM.
12. On Linux hosts, start the Docker swarm, microservices node, and the Gateway service.

The Gateway service should *not* join the swarm.

## Install the Active Workspace Gateway using Deployment Center

You can install Active Workspace Gateway into a new or existing environment:

1. Make sure you have the following software in your repository.
  - Teamcenter major release
  - Teamcenter minor release
  - Active Workspace 5.2
  - Microservices Framework

Choose your new or existing environment, and create or update the **Selected Software** list.

2. In **Selected Applications**, required **Active Workspace** applications are automatically listed. You may add other applications you want to deploy that do not use **Active Workspace Gateway**.
3. In the **Selected Components** list, choose **Active Workspace Gateway**.

Enter the machine name and operating system. The installation path to Teamcenter may be specified automatically if it was entered in another component.

- Expand the configuration sections to show all parameters, and enter the required values.

Value	Description
<b>Port</b>	Enter the port for Active Workspace Gateway. The default value is <b>3000</b> . The URL to the Active Workspace client interface will use this port.
<b>TLS certificate file path</b>	If you use HTTPS protocol, enter the path to the certificate file.
<b>TLS key file path</b>	Enter the path to the keystore ZIP file ( <b>keys.zip</b> ) on the Active Workspace Gateway host.  The file must be copied from the microservice framework master node to the Active Workspace Gateway host.
<b>Use as Bootstrap URLs</b>	The Active Workspace client uses FMS to download and upload files. You define the FSC servers that are used by selecting either <b>Use as Bootstrap URLs</b> or <b>Use Assigned FSC URLs</b> .  On Linux hosts, you must select <b>Use as Bootstrap URLs</b> to ensure the client map is configured correctly.
<b>FSC Connection URLs</b>	Specifies a comma-separated list of URLs to one or more existing FMS server caches (FSCs).  The URL must be of the form:  <b>http://host:port</b>  By default, the IP address from the HTTP connection of the requestor is used unless a <b>Bootstrap Client IP</b> value is provided. (The client/requestor is the host on which Active Workspace Gateway is deployed.)
<b>Bootstrap Client IP</b>	Specifies the FMS bootstrap client IP address to be used for the assignment.  On Linux hosts, enter the internal IP address of the Active Workspace Gateway machine.
<b>Use Assigned FSC URLs</b>	Specifies whether you want to assign FSC servers. Select this only if you want explicit control of the FSCs used.
<b>Assigned FSC URLs</b>	Specifies a comma-separated list of one or more assigned FSC URL values.  The URL values entered are directly used for file operations. This allows you to declare the FSC servers that should be used.

Service Dispatcher URLs are obtained from the Microservice Node.

You may choose whether to communicate with Teamcenter through the Teamcenter web tier or through a load balancer. Specify your settings in the **Teamcenter Server Connection Settings** section.

5. You may specify the configuration for other components now or later. Proceed to **installing the Active Workspace client** for instructions.
6. (If Active Workspace Gateway is deployed on a Linux host) Start the Docker swarm, microservices node, and the Gateway service.

The Gateway service should *not* join the swarm.

## Restricting access to the JMX interface

The Active Workspace Gateway's microservice manager contains a Java Management Extensions (JMX) interface that allows administrators to dynamically change microservice manager configuration values such as instance count for the processes it controls. This could potentially allow non-administrator users to view microservice manager controls. Changing settings of the JMX interface requires a username and password, but non-administrator users may be able to see or connect to the web tier machine on which the Active Workspace Gateway is running.

If you want to prevent access to the JMX interface, locate the following line in the **serverPool.properties** file:

```
JMX_HTTP_ADAPTOR_PORT=8082
```

To disable access to the JMX interface, set this property to a value of **0**, or comment out the line.

## Install the Active Workspace client

Before you install the Active Workspace client using TEM, you must complete the following:

- *Install Active Workspace microservices*
- *Install Active Workspace Gateway*

If you are using Deployment Center, you can install the Active Workspace client concurrently with Active Workspace Gateway.

Install the Active Workspace client configuration using the following software kits:

- Teamcenter major release
- Teamcenter minor release
- Active Workspace 5.2

- Microservices Framework

Note:

Teamcenter software kits contain both Teamcenter and Teamcenter Rapid Start.

## Install the Active Workspace client using TEM



1. Launch TEM from the Teamcenter minor release kit.
2. Proceed to the **Install/Upgrade Options** panel and then click **Install**.
3. In the **Media Locations** panel, enter paths to the software kits.
  - In the **Original Media Location** box, enter the path to the software kit for the major version of Teamcenter or Rapid Start.
  - In the **Update Location** box, enter the paths to Active Workspace 5.2 and Microservices Framework software kits.

The path to the Teamcenter or Rapid Start minor version software kit is already included in the list of update locations.

4. In the **Configuration** panel, enter an **ID** and a **Description** for the configuration.
5. In the **Features** panel, under **Base Install**→**Active Workspace**→**Client**, select **Active Workspace Client**.

Select any **additional client features** you want to include in your Active Workspace environment. (See [Active Workspace Client features](#).)

Note:

- Some features add additional panels to the installation process that are not described in this procedure. For information about any TEM panel, click the help button .
- To search for a feature by name, type the name or a partial name in the search box, and then click the search button .
- Some features depend on other features. Prerequisite features are usually listed earlier than the features that depend on them.

6. In the **Active Workspace Client Settings** panel, enter the required values:

Value	Description
<b>Publish to Gateway</b>	<p>Select this check box to enable automatic publishing of Active Workspace content to the Gateway.</p> <div style="border: 1px solid black; padding: 10px; margin-top: 10px;"> <p>Note:</p> <p>The Active Workspace Gateway must be installed <i>and</i> running before content can be published.</p> </div>
<b>Gateway URL</b>	<p>Type the URL to Active Workspace Gateway:</p> <p style="text-align: center;"><b>http://host:port</b></p> <p>Replace <i>host</i> with the host on which you installed the Gateway. Replace <i>port</i> with the <b>port you specified when you installed the Gateway</b>.</p>

For information about other values in the **Active Workspace Client Settings** panel, click the help button .

7. In the **Confirmation** panel, click **Start** to begin the installation.
8. When the installation is complete, close TEM.

## Install the Active Workspace client using Deployment Center

1. Selecting the Active Workspace software automatically includes its basic applications in the **Selected Applications** list. The associated components required to deploy Active Workspace are listed in the **Selected Components**.

If you haven't already, you can select additional applications you want to include in your Active Workspace environment.

2. In the **Selected Components** list, choose **Active Workspace Client**.

Enter the machine name and operating system. The installation path to Teamcenter may be specified automatically if it was entered in another component.

3. If you want to automatically publish Active Workspace content to the Gateway, check **Publish Active Workspace Client Assets**.

**Note:**

The Active Workspace Gateway must be installed *and* running before content can be published.

4. When the remaining component configuration is complete, click **Go to Deploy** and generate your deployment scripts.

## Sign in using the Active Workspace interface

From a client machine in your environment, sign in to the Active Workspace client. This verifies the environment and the Active Workspace Gateway are running.

1. Sign in to a client machine in your environment.
2. Open a supported web browser.
3. Open the Active Workspace client URL:

**`http://gateway-host:gateway-port`**

Where *gateway-host* and *gateway-port* are the host and port of the Active Workspace Gateway.

For example:

**`http://myhost:3000`**

4. Type your Teamcenter user name and password and click **Sign in**. Verify that you can sign in without errors.

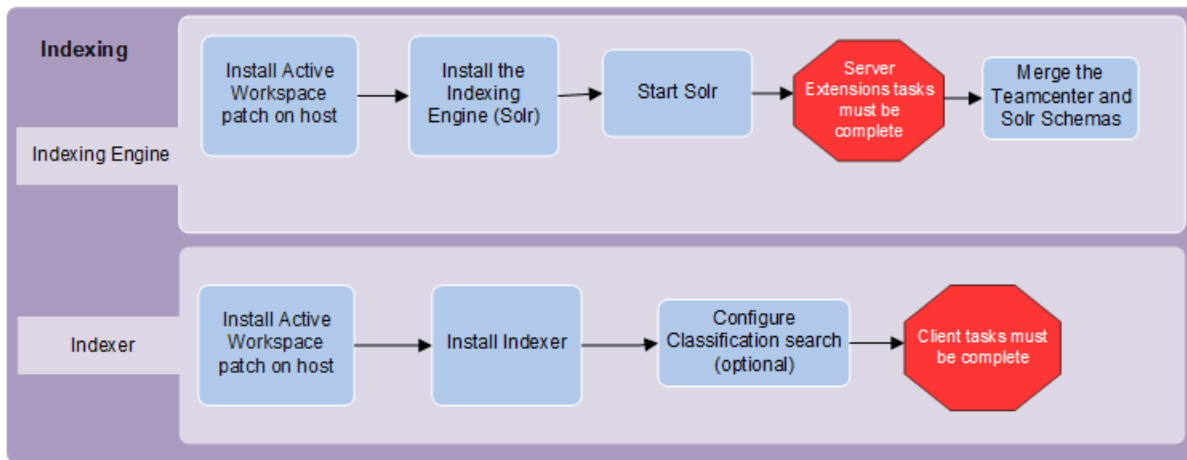
You can further verify status of the Active Workspace Gateway and microservices using the Gateway ping functionality.



# Indexing

## Prepare to install indexing

Before proceeding with installation, be sure you understand the indexing components you want to install.



To configure object data indexing, perform the following steps as described in *Indexing and Search Deployment and Configuration*:

1. Determine your indexing strategy.
2. Install Indexing Engine and the Indexer.

## Configure Classification Search (optional)

If you are new to classification and want to search for classification data in Active Workspace using classification standard taxonomy, your first step is to create a classification hierarchy.

If you already use traditional classification and want to search for classification data in Active Workspace or Teamcenter Manufacturing Access, perform the following steps:

1. Create search index views and specify facetable properties.
2. Update and merge the schema file.
3. Index or reindex your data. If your data is already indexed, it is sufficient to index only the modified classification data.

These steps are described in *Classification and Library Management*

## Install Dispatcher

The Teamcenter *Dispatcher* is an asynchronous executor and load balancer of scheduled jobs. If you use Dispatcher, install the Dispatcher server and client as described in *Installing and Configuring Dispatcher* in the Teamcenter help library. Then, install the following Dispatcher translators, which Active Workspace uses:

- **Active Content Structure Translator**

Install this translator if you use Dispatcher-based indexing for structure data. This feature must be installed in the same environment as the Dispatcher server.

- **ReqMgmtWordToHtmlTrans** (optional)

This translator converts requirements content that has been edited and saved in Microsoft Word from Teamcenter (stored as a full-text dataset), so that it can be viewed in the rich text editor in Active Workspace.

- **AsyncService** (optional)

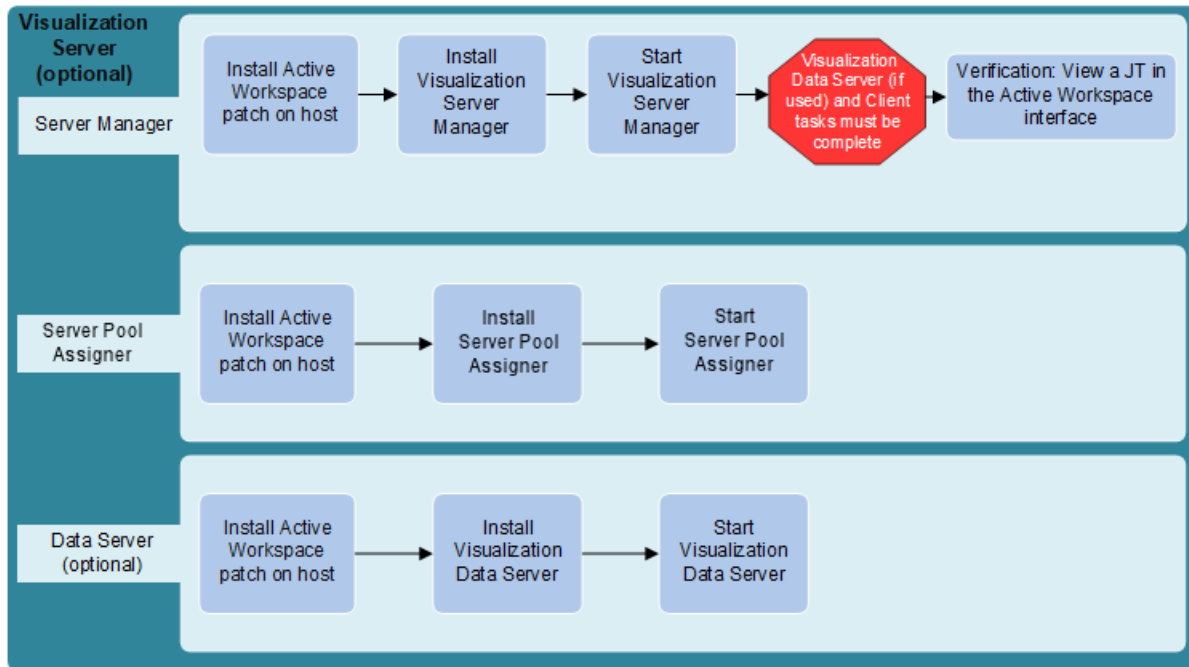
This translator provides asynchronous reporting and printing.

To set up email notifications this translator uses, set the following preferences:

- **MAIL\_OSMail\_ACTIVATED** = true
- **MAIL\_INTERNAL\_MAIL\_ACTIVATED** = true
- **MAIL\_SERVER\_CHARSET** = ISO-8859-1
- **MAIL\_SERVER\_NAME** = *mail-server-name*
- **MAIL\_SERVER\_PORT** = 25
- **MAIL\_SUBSCRIPTION\_NOTIFY\_SUB\_GROUP\_TOO** = FALSE
- **WEB\_DEFAULT\_SITE\_SERVER** = *host:port*
- **WEB\_DEFAULT\_SITE\_DEPLOYED\_APP\_NAME** = *Teamcenter-web-tier-application*

# Visualization Server

## Visualization Server overview



The Visualization Server provides dynamic 3D and 2D visualization functionality to the Active Workspace client. If you do not use the 3D viewer or the 2D part of the universal viewer in Active Workspace, do not install the Visualization Server.

The Visualization Server comprises three components:

### Visualization Server Manager

The Visualization Server Manager (VSM) starts and stops rendering processes as needed and streams visualization data to the Active Workspace client.

The Visualization Server Manager is required for any use of the 3D viewer or the 2D viewer part of the universal viewer in Active Workspace.

Siemens Digital Industries Software recommends that you install the Visualization Server Manager on a machine that does not have a Teamcenter corporate server.

### Visualization Server Pool Assigner

The Visualization Server Pool Assigner (VPA) manages Visualization Server Managers and routes users to an available VSM to open 3D documents.

Each Visualization Pool Assigner hosts two MXBeans that contain information about its current state: **Assigner** and **Assigner monitoring**. The MXBeans are located in the **Administer Assigner manager** folder.

Siemens Digital Industries Software recommends that you install the Visualization Server Manager on a machine that does not have a Teamcenter corporate server.

#### Visualization Data Server

For appropriately indexed product configurations, the Visualization Data Server (VDS) does the following to promote faster rendering and streaming to the Active Workspace client:

- Caches product structure
- Prepopulates JT files in the FCC
- Computes Massive Model Visualization (MMV) spatial hierarchies
- Provides bounding box validation  
You can use bounding box validation to suppress display of parts that fall outside a defined assembly box. This can help avoid assemblies opening zoomed out to accommodate errant parts located far outside the actual assembly. Bounding box validation can also limit a view to include only a preferred range of the assembly.  
Bounding box validation is described in *Configuration and Extensibility* in the Active Workspace help library.

A single Visualization Data Server can support one or more Visualization Server Managers.

A Visualization Server Manager is required on the same host as the Visualization Data Server. A Visualization Data Server is required for implementation of MMV, but is otherwise optional.

### Should I use client-side or server-side rendering?

#### Rendering 3D data

The **3D** viewer is displayed within the universal viewer area in the **Overview** tab for objects that have viewable attachments. The 3D viewer is also displayed in the **3D** tab to explore 3D data (JT) associated with parts and assemblies.

The render location setting applies to both viewer locations.

#### Client-side rendering

Client-side rendering uses WebGL to leverage client-side graphics capabilities via the Active Workspace client browser.

**Note:**

The Visualization Server is still required for visualizing 3D data in Active Workspace with client-side rendering. However, no graphics card is required on the Visualization Server.

**Note:**

When using Firefox, client-side rendering requires Firefox version 75 and later.

Client-side rendering provides these benefits:

- A server-side graphics card is not required.
- Rendering does not rely on the processing power of the server.
- Rendering is unaffected by network traffic and is more responsive and less latency sensitive. However, the triangles of the model must be loaded onto the client machine before it can render.

Server-side rendering may still be more suitable for larger data sizes due to its typically faster load speed and a higher data size limit. Server-side rendering is necessary for client devices that do not support Web GL.

## Set client-side rendering

To set the default rendering method for the 3D viewer and the universal viewer, specify the correct value in the Teamcenter preference **AWV0ViewerRenderOption**:

**CSR** (client-side rendering)

**SSR** (server-side rendering) This is the default value.

The user can change the rendering method on the **Viewer Options** panel.

## Should I use MMV?

Massive Model Visualization (MMV) is a visualization technology that uses Visibility Guided Rendering (VGR) to increase performance and scalability for the viewing of extremely large 3D models, such as cars, airplanes, and ships. Models of this size typically consist of a massive amount of geometry arranged in a relatively compact space with a huge amount of internal geometry hidden behind the "outer shell" of the product. It can take hours to display such models in their entirety, because every piece of geometry in the model needs to be retrieved and processed, which far exceeds the capability of today's commonly available hardware. MMV technology resolves this problem by leveraging VGR techniques to load only those parts required to render a given scene; parts that are not visible because they are occluded by other parts in the foreground are not loaded. As a result, large 3D models become visible in a fraction of the time previously required.

If a structure has more than 120,000 BOM lines, MMV rendering is required for scalability and performance. If a structure has more than 30,000 BOM lines, MMV is recommended.

Visualization of MMV data in the Active Workspace client requires an MMV license. If the necessary license is not present, the full model loads as standard JT data.

A Visualization Data Server is required for implementation of MMV, but is otherwise optional.

To use the Visualization Data Server to compute Massive Model Visualization (MMV) spatial hierarchies of structures, you must

1. Apply the MMV index structure flag to the product configurations that you want to view using MMV.
2. Use the **bomindex\_admin** utility to include the configurations in the list of structures to index.

To take advantage of the structure and JT pre-caching feature of the Visualization Data Server for improved visualization performance of structures that are not indexed for MMV:

1. Apply the VDS indexing flag for product configurations that will be viewed frequently but are not indexed for MMV.
2. Use the **bomindex\_admin** utility to include the configurations in the list of structures to index.

## Visualization Server Manager

### Visualization Server Manager prerequisites

#### Operating systems

The Visualization Server Manager (VSM) supports both large model visualization (LMV) and massive model visualization (MMV) on supported Microsoft Windows server platforms. On supported Linux server platforms, LMV is supported, but MMV is not yet supported.

On a Linux machine without a GPU or without a supported level of OpenGL, client side rendering is supported, but server side rendering is not supported and fails to load.

For supported OS versions, see the Hardware and Software Certifications knowledge base article on Support Center.

#### Server hardware and graphics cards

The following hardware is supported for the Visualization Server Manager.

- **For server side rendering:**

Server class hardware certified by NVIDIA to support NVIDIA GRID K1, K2, Tesla M60, P40, T4, RTX 6000, or RTX 8000 graphics cards. Note that any server capable of supporting Server Side Rendering also supports client side rendering.

- **For client side rendering:**

If no server side rendering is needed, any web server class hardware is sufficient to support client side rendering.

Sizing of hardware should be appropriate to support intended data sizes and usage patterns. See [VSM hardware sizing](#) for more info about hardware sizing.

Windows Server versions supported with the Visualization Server Manager support a maximum of 8 GPUs, with certain exceptions. For example, on a Windows Server 2012 R2 machine with two NVIDIA GRID K1 cards, the legacy VGA device makes the fourth GPU on one card unavailable for use.

Active Workspace versions earlier than Active Workspace 5.2 do not support virtualized server-side rendering. Active Workspace 5.2 supports virtualization for **certain hardware and software combinations**.

For information about server hardware compatible with supported NVIDIA GRID graphics cards, see [www.nvidia.com](http://www.nvidia.com).

## Virtualization

If you use only client-side rendering, the VSM can be virtualized.

For server-side rendering with Active Workspace versions earlier than Active Workspace 5.2, the VSM must be installed on physical hardware. But, with Active Workspace 5.2, the VSM can be virtualized if you follow a supported virtualization combination.

Active Workspace visualization supports virtualization for certain combinations of:

- Host OS and version
- Virtualization layer
- Guest OS and version
- NVIDIA GPU

For information about supported combinations, see the Graphics Card Certification Matrix in the Hardware and Software Certifications knowledge base article on Support Center: <https://support.sw.siemens.com>

For information about NVIDIA virtual GPU compatibility, see NVIDIA virtual GPU (vGPU) software documentation at [docs.nvidia.com](https://docs.nvidia.com).

## VSM hardware sizing

Sizing of VSM hardware should allow for typical and maximum expected usage by considering the following factors:

- Expected numbers of concurrent Active Workspace visualization users
- Expected product data sizes
- CPU, RAM, VRAM and GPU resources consumed by expected product data

In general, a high end server with:

- A maximum number of CPU cores with processing speeds of 3.0 GHz or faster
- A minimum of 64 GB of RAM
- A minimum of 256 GB of disc space

In addition, a VSM that will support Server Side Rendering requires an NVIDIA GRID graphics card. For information about server hardware compatible with supported NVIDIA GRID graphics cards, see [www.nvidia.com](http://www.nvidia.com).

For additional guidance in sizing of VSM hardware, contact your field services professionals.

## Environment information

Make sure you know the following values. These are needed during installation of the Visualization Server Manager.

**Visualization Server Pool Assigner host and port** These are defined in [Install the Visualization Server Pool Assigner](#).

**Visualization Data Server host and port (if VDS is to be installed)** These are defined in [Install the Visualization Data Server](#).

**Host and port of FCC parents** These are defined during Teamcenter installation.



## Linux machine configuration

Before you install the VSM on a Linux machine, perform the following steps:

1. Make sure the machine has the **required RPM package managers**.
2. Configure Xserver on the machine.

- **Linux machine with no GPU:**

Configure Xserver for offscreen and headless operation for use by the visualization server processes. After reboot, run the following commands:

```
- setenv DISPLAY :0
- sudo xhost +
```

- **Linux machine with GPU:**

- a. Configure Xserver for offscreen and headless operation for use by the VisServer processes.
- b. Make sure the XServer is running. One way to verify this is to see if the **X** or **Xorg** process is running by typing the following command:

```
ps -ef | grep X
```

- c. Set up the NVidia GPU on the Linux machine by typing the following command:

```
setup_xserver.sh default
```

The `default` parameter specifies to use the graphics card and bus id discovered by the script. If you do not specify this parameter, the script prompts you to confirm the card and bus id, and provides the opportunity to change these values if you want.

For example:

```
sudo ./setup_xserver.sh
```

Or:

```
sudo ./setup_xserver.sh default
```

## Install the Visualization Server Manager

You can install the Visualization Server Manager in a new or an existing Teamcenter configuration.

If you are adding the Visualization Server Manager to an existing configuration, proceed as follows:

1. **Apply the Active Workspace patch** to the host.
2. Launch the installed TEM in maintenance mode and skip to step **7** below.

Otherwise, if you are installing the Visualization Server Manager on a host with *no* existing Teamcenter environment, proceed to step **1** below.

1. Ensure you have access to the following required software kits:
  - Latest Teamcenter minor release, for example, Teamcenter 13.2
  - Corresponding Teamcenter major release, for example, Teamcenter 13
  - Active Workspace 5.2

Make sure you use the appropriate Active Workspace 5.2 software kit that corresponds to your version of **Teamcenter**.

1. Launch TEM from the Teamcenter minor release software kit.

On Windows systems, launch TEM with administrator privileges (right-click→**Run as administrator**).

2. In the **Welcome to Teamcenter** panel, select **Teamcenter**.
3. In the **Install / Upgrade Options** panel, click **Install**.
4. In the **Media Locations** panel, specify locations of Teamcenter software kits:
  - a. In the **Original Media** box, enter the location of the Teamcenter major release software kit.
  - b. In the **Update Location** box, add the location of the Active Workspace 5.2 software kit.

Note that the location of the Teamcenter minor release software kit is already included in the list. Make sure the Teamcenter location is listed *before* the Active Workspace location.

5. In the **Configuration** panel, enter values for **ID** and **Description**.
6. In the **Solutions** panel, make no selections.
7. Proceed to the **Features** panel. Select the **Visualization Server Manager** feature, under **Base Install**→**Active Workspace**→**Visualization Server**.

In the **Installation Directory** box, enter the location in which to install the Visualization Server Manager.

8. In the **File Client Cache (FCC)** panel, set the **FMS\_HOME** environment variable.
9. In the **FCC Parents** panel, define the list of FSC parents to which the Visualization Server Manager connects. You must provide protocol, host, and port of the FSC parent. To add rows to the table, click **Add**.

If you are using a Visualization Data Server, the Visualization Data Server and the Visualization Server Manager should use the same FSC.

10. In the **Configuration** section of the **Visualization Server Manager** panel, provide values for the Visualization Server Manager.

**Local Host Alias** Specifies the alias for the local Visualization Server Manager.

**Server Host** Specifies the host where the Visualization Server Manager is running. This must be the local machine name and must be resolvable by the Visualization Pool Assigner machine (the machine running the Active Workspace Gateway). Do not use **localhost** or **127.0.0.1**.

**Server Port** Specifies the port on which the Visualization Server is listening.

**Max Servers in Sub-Pool** Specifies the maximum number of Visualization server processes allowed to run in this pool (for a single-host configuration) or in this subpool (for a multihost configuration).

For example, on Windows machines, the default value is **30**. On Linux machines, the default value is **200**.

**Min Warm Servers** Specifies the minimum number of Visualization server processes in this pool that are started but not assigned.

**Note:**

If necessary to maintain the minimum number of warm servers, while not exceeding the maximum number of server processes, the server manager times out servers in use.

11. If you use a Visualization Data Server, provide the values for this server in the **Visualization Data Server Configuration** section of the **Visualization Server Manager** panel.

**Add Visualization Data Server** Select this if you are using a Visualization Data Server.

**Host** Type the name of the host on which the Visualization Data Server is installed.

**Port** Type the port value used by the Visualization Data Server.

A Visualization Data Server improves performance by caching product structure and JT parts files.

12. In the **Visualization Server Manager Settings** panel, define the settings for how the Visualization Server Manager communicates with the pool assigners.
  - **Override local node settings**  
Select to override the host name and port value. Enter the **Host** and **Vis Assigner Port** values of the local machine. In the **Gateway Connection Port** box, enter the port used by the Active Workspace Gateway.
  - **Visualization Server Pool Assigners**  
Lists the pool assigners that this Visualization Server Manager uses.
    - **Assigner Host**  
Host name of the machine where the Visualization Server Pool Assigner runs.
    - **Assigner Port**  
Port value of the pool assigner.
13. In the **Operating System User** panel, type the user's password.
14. In the **Confirmation** panel, click **Start**.
15. When the installation is complete, close TEM.

## Start the Visualization Server Manager

### Start the Visualization Server Manager on Linux

To start the Visualization Server Manager (VSM) on a Linux machine, type the following command:

```
TC_ROOT/vispoolmanager/run_servermgr.sh
```

#### Optional: Start the VSM as a Linux daemon

Alternatively, you can start the Visualization Server Manager as a daemon by running the **installservice.sh** command for each jetty server (run with admin permissions):

```
installservice.sh unique-service-name port user
```

If you do not specify parameters, the script will run in interactive mode and prompt you for the information.

For example, from the `TC_ROOT/vispoolmanager/jetty` directory, type:

```
sudo ./installservice.sh MyUniquePoolManager1 8090 MyUser
```

*Uninstalling the Linux daemon:*

To *uninstall* the VSM daemon, run the **uninstallservice.sh** command for each jetty server (run with admin permissions):

```
uninstallservice.sh service-name
```

For example, from the `TC_ROOT/vispoolmanager/jetty` directory, type:

```
sudo ./uninstallservice.sh MyUniquePoolManager1
```

If you do not know the name of the service, look in the `TC_ROOT/vispoolmanager/jetty/` or `/etc/systemd/system` directory and find the file named `service-name.service`. The *service-name* is the unique service name you provided to the **installservice.sh** command.

## Start the Visualization Server Manager on Windows

1. Make sure the **FMS\_HOME** environment variable is set as a system environment variable, not a user variable. The Visualization Server Manager (VSM) runs as a service only if **FMS\_HOME** is a system environment variable.
2. Run the following file:

```
TC_ROOT\vispoolmanager\run_visservermgr.cmd
```

### Note:

If the Visualization Server Pool Assigner is not running, the Visualization Server Manager displays console messages until it finds the VPA. To avoid this, **start the Visualization Server Pool Assigner** before you start the Visualization Server Manager.

When running **run\_visservermgr.cmd**, you can use Windows Remote Desktop Connection to sign in to the machine on which the Visualization Server Manager is installed if you have an NVIDIA card with a driver version of 340.66 or later. Other remote access products, such as VNC, also can be used.

After running **run\_visservermgr.cmd**, you can lock the machine, but you must remain logged on. If you sign out, the Visualization Server Manager is shut down.

The Visualization Server Manager requires access to the graphics card and consequently cannot run as a Windows service in server-side rendering mode. However, you can start the Visualization Server Manager as a Windows service when you use client-side rendering exclusively.

## Optional: Configure automatic logon and restart on Windows

You can configure Windows to automatically log on and restart the Visualization Server Manager in the event of a system reboot.

**Caution:**

Enabling automatic logon bypasses security. When Windows is configured to automatically log on, anyone with physical access to the machine can restart it and gain entry to the system. Use automatic logon *only* if the system is in a secure environment.

1. Open the Windows User Accounts dialog box:
  - a. Press the Windows key+R to display the **Run** dialog box.
  - b. In **Open**, type **netplwiz**, and then click **OK**.
2. In the **User Accounts** dialog box, select a user account from the list.
3. Clear the **Users must enter a user name and password to use this computer** check box.
4. Click **Apply**.

The **Automatically sign in** dialog box is displayed.

5. In the **Password** and **Confirm Password** boxes, type the user's password.
6. Click **OK**.

The specified user is automatically logged on when Windows starts.

7. Create a script or batch file to launch the Visualization Server Manager. Include the following command to lock the workstation:

```
rundll32.exe user32.dll LockWorkStation
```

8. Create a new task with Windows Task Scheduler to run the script or batch file at log on.

### Optional: Start the VSM as a Windows service

You can start the Visualization Server Manager as a Windows service only when you are exclusively using client-side rendering. Windows services cannot access the graphics card, so this is not a suitable deployment for server-side rendering.

1. To install the Visualization Server Manager as a Windows service, run the **installservice.bat** command:

```
installservice.bat "%JAVA_HOME%" "VSM-dir" pool-ID port
```

Replace *VSM-dir* with the path to the Visualization Server Manager's **jetty** directory. Replace *pool-ID* and *port* with the ID and port for the **VisPoolManager** service. The port must match the **VisPoolProxy.poolUrl** port in the **jetty.service.properties** file.

For example:

```
installservice.bat "%JAVA_HOME%" "%TC_ROOT%\vispoolmanager\jetty" vispool-A 8090
```

2. Locate the newly installed service named **Teamcenter VisServlet** *pool-ID* in the list of Windows services.
3. Right-click the service name and choose **Properties**.
4. On the **Log On** tab, enter logon credentials for the domain user account under which the service runs.

Visualization Server Manager requires an FMS client cache (FCC) to cache files. It is recommended that you use a dedicated account to run this service, not the **Local System** account.

Windows attempts to run the service automatically by default. If the service is not already running, a problem may have occurred.

If you set the service to start manually in its **Properties**, then you can click the **Run** button from the toolbar to start the service, or right-click the service in the **Services** window, and choose **Start**.

To stop the service, either click **Stop Service** on the toolbar, or right-click the service and choose **Stop**.

To uninstall the service, type `uninstallservice.bat "Teamcenter VisServlet pool-ID"`.

## Test Visualization from the Active Workspace client interface

Before you begin the following procedure, make sure the Visualization Server Manager installation *and* the Active Workspace client installation tasks are complete.

You can test the Visualization Server by logging on with the Active Workspace interface and viewing Visualization data, for example, a JT file.

1. Ensure that the following are running:
  - Visualization Server Manager
  - Visualization Pool Assigner
  - Active Workspace Gateway

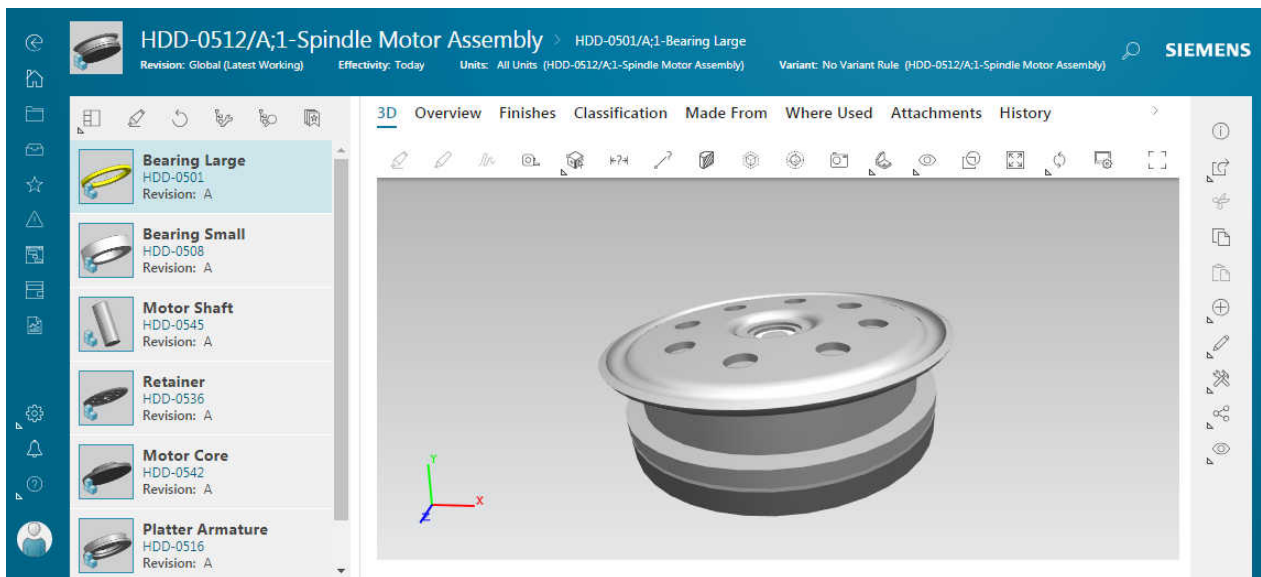
- Web application server hosting the Teamcenter web tier application
  - Teamcenter server manager
  - Teamcenter database
2. Open a supported web browser.
  3. Open Active Workspace at the following URL:

**`http://host:port`**

*host* is the machine running the Active Workspace Gateway.

*port* is the port used by the Active Workspace Gateway.

4. Sign in with a valid user name and password.
5. Search for and open an object that has an attached JT file.
6. Click the **3D** tab to display the JT file.



## Visualization Server Pool Assigner

### Visualization Server Pool Assigner prerequisites

#### Software

The Visualization Server Pool Assigner requires the following software:



- A supported Microsoft Windows Server operating system or Linux operating system on the Visualization Server Manager host.  
For supported versions, see the Hardware and Software Certifications knowledge base article on Support Center.
- The **Visualization Extension** Server Extensions feature on the corporate server and on any server that has Teamcenter Foundation installed.

## Environment information

Make sure you know the following values. These are needed during installation of the Visualization Server Manager.

Visualization Server Pool Assigner host and port

**Visualization Data Server host and port (if VDS is to be installed)** These are defined in [Install the Visualization Data Server](#).

**Host and port of FCC parents** These are defined during Teamcenter installation.

## Install the Visualization Server Pool Assigner

You can install the Visualization Server Pool Assigner in a new or an existing Teamcenter configuration.

If you are adding the Visualization Server Pool Assigner to an existing configuration, proceed as follows:

1. [Apply the Active Workspace patch](#) to the host.
2. Launch the installed TEM in maintenance mode and skip to step **7** below.

Otherwise, if you are installing the Visualization Server Pool Assigner on a host with *no* existing Teamcenter environment, proceed to step **1** below.

1. Ensure you have access to the following required software kits:
  - Latest Teamcenter minor release, for example, Teamcenter 13.2
  - Corresponding Teamcenter major release, for example, Teamcenter 13
  - Active Workspace 5.2

Make sure you use the appropriate Active Workspace 5.2 software kit that corresponds to your version of **Teamcenter**.

1. Launch TEM from the Teamcenter minor release software kit.

On Windows systems, launch TEM with administrator privileges (right-click→**Run as administrator**).

2. In the **Welcome to Teamcenter** panel, select **Teamcenter**.
3. In the **Install / Upgrade Options** panel, click **Install**.
4. In the **Media Locations** panel, specify locations of Teamcenter software kits:
  - a. In the **Original Media** box, enter the location of the Teamcenter major release software kit.
  - b. In the **Update Location** box, add the location of the Active Workspace 5.2 software kit.

Note that the location of the Teamcenter minor release software kit is already included in the list. Make sure the Teamcenter location is listed *before* the Active Workspace location.

5. In the **Configuration** panel, enter values for **ID** and **Description**.
6. In the **Solutions** panel, make no selections.
7. Proceed to the **Features** panel. Select the **Visualization Server Pool Assigner** feature, under **Base Install**→**Active Workspace**→**Visualization Server**.

In the **Installation Directory** box, enter the location in which to install the Visualization Server Manager.

8. In the **Visualization Server Pool Assigner Settings** panel, enter settings to configure the Visualization Server Pool Assigner:

Value	Description
<b>Host</b>	Specifies the host on which this Visualization Server Pool Assigner runs. This is the host on which this Visualization Server Pool Assigner is deployed. You may type the host name or IP address of the host.
<b>Vis Assigner Port</b>	Specifies the port used by the local Visualization Server Pool Assigner.
<b>Gateway Connection Port</b>	Specifies the port through which the Active Workspace Gateway connects to the Visualization Server Pool Assigner. The default value is <b>3000</b> .

Value	Description
<b>Add pool assigner</b>	Specifies whether additional Visualization Server Pool Assigners are used. Select the <b>Add pool assigner</b> check box to add pool assigners.
<b>Peer Assigners</b>	The <b>Peer Assigners</b> table lists other Visualization Server Pool Assigners known to the local Visualization Server Pool Assigners.
<b>Assigner Host</b>	Specifies the host on which a peer Visualization Server Pool Assigner is deployed. This value can be the host name or IP address.
<b>Assigner Port</b>	Specifies the port number used by the peer Visualization Server Pool Assigner.
<b>Server Side 4-tier URL</b>	Specifies an alternate four-tier URL for the viewer to connect to Teamcenter. The Visualization Server uses this in case the primary four-tier URL is blocked by a firewall, or if a more direct route is needed for performance.

9. In the **Confirmation** panel, click **Start**.
10. When the installation is complete, close TEM.

## Start the Visualization Server Pool Assigner

### Linux Systems

To start the Visualization Server Pool Assigner (VPA) on a Linux machine, type the following command:

```
TC_ROOT/visassigner/run_assigner.sh
```

Alternatively, you can start the Visualization Server Pool Assigner as a daemon by running the **installservice.sh** command for each jetty server (run with admin permissions):

```
installservice.sh unique-service-name port user
```

If you do not specify parameters, the script will run in interactive mode and prompt you for the information.

For example, from the **TC\_ROOT/visassigner/jetty** directory, type:

```
sudo ./installservice.sh MyUniqueAssigner1 7780 MyUser
```

## Windows Systems

To start the Visualization Server Pool Assigner (VPA) on a Windows machine, run the following file:

```
TC_ROOT\visassigner\run_visassigner.cmd
```

After running **run\_visassigner.cmd**, you can lock the machine, but you must remain logged on. If you sign out, the VPA is shut down.

Alternatively, you can start the Visualization Server Pool Assigner as a Windows service only when you are exclusively using client-side rendering. Windows services cannot access the graphics card, so this is not a suitable deployment for server-side rendering.

1. To install the Visualization Server Pool Assigner as a Windows service, run the **installservice.bat** command:

```
installservice.bat "%JAVA_HOME%" "VPA-dir" assigner-ID port
```

Replace *VPA-dir* with the path to the Visualization Server Pool Assigner's **jetty** directory. Replace *assigner-ID* and *port* with the ID and port used by the Active Workspace Gateway to connect to the Assigner.

For example:

```
installservice.bat "%JAVA_HOME%" "%TC_ROOT%\visassigner\jetty" VisAssigner-A 8089
```

2. Locate the newly installed service named **Teamcenter VisServlet assigner-ID** in the list of Windows services.
3. Right-click the service name and choose **Properties**.
4. On the **Log On** tab, enter logon credentials for the domain user account under which the service runs.

Windows attempts to run the service automatically by default. If the service is not already running, a problem may have occurred.

If you set the service to start manually in its **Properties**, then you can click the **Run** button from the toolbar to start the service, or right-click the service in the **Services** window, and choose **Start**.

To stop the service, either click **Stop Service** on the toolbar, or right-click the service and choose **Stop**.

To uninstall the service, type `uninstallservice.bat "Teamcenter VisServlet assigner-ID"`.

## Visualization Data Server (optional)

### Visualization Data Server prerequisites

#### Software

The Visualization Data Server requires the following software:

- Microsoft Windows Server operating system on the Visualization Data Server host.  
For supported versions, see the Hardware and Software Certifications knowledge base article on Support Center.
- The **Visualization Extension** Server Extensions feature on the corporate server and on any server that has Teamcenter Foundation installed.
- A **Visualization Server Manager** installed on the Visualization Data Server host.
- An FMS client cache (FCC) component on the Visualization Data server host.
- Structure indexing configured on the Visualization Data server host.  
The Visualization Data Server uses the structure indexing infrastructure of Active Workspace to keep cached product structure up-to-date.

#### Hardware

- Graphics card: No requirements.
- Network: You must deploy the Visualization Data Server on a high speed LAN near the Visualization Server Manager.
- Memory: The Visualization Data Server host should have a minimum of 16 GB of RAM, but may require more.

Note:

#### How to determine memory needed:

The amount of RAM needed depends on the number of structures to be indexed and their size.

A rough rule of thumb is to count the number of lines in the unconfigured structure to be indexed and allow at least 2000 bytes per line. For example, if there are 1 million lines in the unconfigured product index, then  $1 \text{ million} * 2000 = 2 \text{ GB of RAM}$ .

If you are not sure of the size of the structures, Siemens Digital Industries Software recommends that you allow approximately 4 GB of RAM for each structure you are planning to cache in the Visualization Data Server. For example, if 4 structures are to be indexed, 16 GB of RAM is recommended.

## Environment information

Installing the Visualization Data Server requires the following values from your Teamcenter environment:

Values required before beginning the Visualization Server taskflow		
Parameter	Where value is defined	Your value
FCC parents	Teamcenter installation	
Teamcenter four-tier URL	Teamcenter installation	

Values to record during the Visualization Server taskflow execution		
Parameter	Where value is defined	Your value
Visualization Data Server host and port	<b>Install the Visualization Data Server</b>	

## Recommendations

Siemens Digital Industries Software recommends that you install the Visualization Data Server on a machine with the following:

- **Multiple processors**

The Visualization Data Server is a multithreaded server program and is thus resource intensive; multiple processors are utilized if they are available. Standard server class machine hardware is sufficient.

- **FSC cache or FSC volume**

If you deploy the Visualization Data Server remote (on a WAN) from the FSC volume, you should deploy an FSC cache on a LAN near or on the Visualization Data Server host machine.

- **Visualization Server Manager**

For maximum performance, the Visualization Data server should be installed on the same machine as the Visualization Server Manager and should use the same cache.

A single Visualization Data Server can support one or more Visualization Server Managers.

## Install the Visualization Data Server

You can install the Visualization Data Server (VDS) in a new or an existing Teamcenter configuration. Siemens Digital Industries Software recommends that you install the Visualization Data Server on a machine that does *not* have a Teamcenter corporate server.

If you are adding the Visualization Data Server to an existing configuration, proceed as follows:

1. **Apply the Active Workspace patch** to the host.
2. Launch the installed TEM in maintenance mode and skip to step **8** below.

Otherwise, if you are installing the Visualization Server Manager on a host with *no* existing Teamcenter environment, proceed to step **1** below.

1. Ensure you have access to the following required software kits:
  - Latest Teamcenter minor release, for example, Teamcenter 13.2
  - Corresponding Teamcenter major release, for example, Teamcenter 13.
  - Active Workspace 5.2

Note:

- Make sure you use the appropriate Active Workspace 5.2 software kit that corresponds to your version of Teamcenter.
- Teamcenter software kits contain both Teamcenter and Teamcenter Rapid Start.

2. Launch TEM from the Teamcenter minor release software kit.
3. In the **Welcome to Teamcenter** panel, select **Teamcenter** or **Teamcenter Rapid Start** as appropriate.
4. In the **Install / Upgrade Options** panel, click **Install**.
5. In the **Media Locations** panel, specify locations of Teamcenter software kits:
  - a. In the **Original Media** box, enter the location of the Teamcenter major release software kit.
  - b. In the **Update Location** box, add the location of the Active Workspace 5.2 software kit.

Note that the location of the Teamcenter minor release software kit is already included in the list. Make sure the Teamcenter location is listed *before* the Active Workspace location.

6. In the **Configuration** panel, enter values for **ID** and **Description**.
7. In the **Solutions** panel, make no selections.
8. In the **Features** panel, select the **Visualization Data Server** feature:

**Base Install→Active Workspace→Visualization Server→Visualization Data Server**

In the **Installation Directory** box, enter the location in which to install the Visualization Data Server.

9. In the **File Client Cache (FCC)** panel, set the **FMS\_HOME** environment variable. The FCC must be installed on the same machine as the Validation Data Server.
10. In the **FCC Parents** panel, define the list of FSC parents to which the Visualization Data Server connects. You must provide protocol, host, and port of the FSC parent. To add rows to the table, click **Add**.

To improve performance, the Visualization Data Server and the Visualization Server Manager should use the same FSC.

11. In the **Teamcenter Administrative User** panel, enter the user's password.
12. In the **Visualization Data Server Configuration** panel, enter required values:

- **Server Port**

This is the port number on which the Visualization Data Server listens.

- **Teamcenter 4-tier URL**

This is the URL of the Teamcenter web tier application. The format is:

**`http://host:port/tc-web-app`**

*host* is the machine running the web application server on which the Teamcenter web application is deployed.

*port* is the port value used by the web application server.

*tc-web-app* is the name of the Teamcenter web application. The default is **tc**.

13. In the **Confirmation** panel, click **Start**.
14. When the installation is complete, click **Close**.

## MMV indexing data

If you use Massive Model Visualization (MMV), configure MMV indexing.

When structures using MMV rendering are indexed, the last valid indexed data is always retained. So, you can always see MMV indexed data; however, the data in a structure may be more recent.



When MMV data is being indexed it may use a backup system. It is recommended that the administrator retains interim files so when an error occurs, they can be analyzed to determine the issue. These two Teamcenter preferences can be used to control the output of the generated files:

- **MMV\_keep\_generated\_files**

Use this preference to preserve the generated files for further examination. You can specify when generated files are kept by using these values:

1: Keep the generated files when an error occurs.

2: Always keep the generated files.

3: Never keep the generated files.

- **MMV\_staging\_directory**

Use this preference to control the working directory to be used for the **tcxml2mmp** conversion process on the Teamcenter server. If this is not set, the default temporary directory is used as staging directory.

## Start the Visualization Data Server

To start the Visualization Data Server Manager, run:

```
TC_ROOT\VisDataServer\Program\VisDataServer.exe
```

After the Visualization Data Server is started, it automatically detects and caches product configurations that have been indexed with the MMV flag. These cached product configurations are now ready for fast visualization with the MMV technology.

A product configuration is ready for MMV visualization after it has been indexed, the Visualization Data Server has detected, downloaded and cached the structure, and prepopulated the FMS system. If you attempt to visualize a product configuration that is not yet completely indexed and cached in the Visualization Data Server, the viewer uses the regular non-MMV mode by default. Changes in the product configuration need to be re-indexed and re-read by the Visualization Data Server before they can be displayed by the viewer.

Additional configuration for the Visualization Data Server is available in the **etc/VisDataServer.properties** file. This includes detailed logging and fine tuning of other settings. If you make changes to the properties file, you need to restart the Visualization Data Server.

### Optional: Start the Visualization Data Server as a Windows service

1. Make sure the **FMS\_HOME** environment variable is set as a system environment variable, not a user variable. The VDS runs as a service only if **FMS\_HOME** is a system environment variable.
2. Inspect the **VisDataServer.properties** and make sure all file paths specified in it are full paths, not relative paths.

3. Open a Teamcenter command prompt and change to the root directory of the Visualization Data Server.
4. Install the Visualization Data Server as a Windows service by running the **VisDataServer.exe** command with the **/registerService** argument:

**VisDataServer.exe /registerService /displayName=name /startup=option**

Replace *name* with a display name for the service. Replace *option* with **automatic** or **manual**.

For example:

```
VisDataServer.exe /registerService /displayName=VisDataServer /startup=automatic
```

Optional additional arguments:

Argument	Description	Example
<b>description</b>	Specifies a description for the service.	/description="VDS for Active Workspace 5.2"
<b>config</b>	Specifies a configuration file to load for the application.	/config="VDSConfig.txt"

After the service is successfully installed, Windows displays the following message:

The application has been successfully registered as a service.

5. Configure the VDS service:
  - a. In the Windows **Services** dialog box, locate the VDS service by the name you specified in the **displayName** attribute.
  - b. Right-click the service name and choose **Properties**.
  - c. In the **Log on** tab, enable the service logon with the following options:
    - **Log on as:** Select **This account**, then enter the domain and user name (for example, **myDomain\myName**).
    - **Password:** Enter and confirm the password for the user account.

**Note:**

The VDS requires an FMS client cache (FCC) to cache files. Use a dedicated account to run this service, not the **Local System** account.

Windows attempts to run the service automatically by default. If the service is not already running when you open the Windows **Services** dialog box, the installation may have failed.

If you set the service to start manually, right-click the service name and choose **Start**. To stop the service, right-click the service name and choose **Stop**.

To uninstall the service, run the **VisDataServer.exe** utility with the **/unregisterService** argument.

## Rebuild VDS repository from scratch

As the VDS repository is updated via deltas containing incremental changes from Teamcenter that occur as design data evolves, the repository used to support MMV viewing may introduce errors. A good practice is to periodically regenerate the VDS repository from scratch. The default threshold for this scratch rebuild is every 500 delta updates, but this value is configurable by an administrator so that a full regeneration of the VDS repository can be set to occur more or less often, depending on the observed need.

To change the number of deltas that are processed before a scratch rebuild of the VDS repository, set the **MMV\_delta\_collection\_accumulation\_limit** Teamcenter preference to a value higher or lower than the default value of 500. This will change how often the VDS rebuilds its repository from scratch.

**Note** — To manage the number of delta files that are to be deleted, use the **MMP\_PERCENTAGE\_OF\_DELTA\_TO\_PURGE** preference. Its default value is 100, but based on your need, you can set it to any value from 10 to 100. Refer to the following table to understand preference value limits:

If preference value is set to	then preference value processed is
$\leq 10$	10
$10 < \text{value} < 100$	value
$\geq 100$	100

All delta files are deleted when the **MMP\_PERCENTAGE\_OF\_DELTA\_TO\_PURGE** preference value is set to 100.

**Example:**

Set the **MMV\_delta\_collection\_accumulation\_limit** preference value to 500.

In this case, since the value is set to 500, it becomes the maximum delta limit. After this many deltas are processed, a completely new mmp file is created.

To delete 100% (all) of the old delta files in the MMV dataset, set the **MMP\_PERCENTAGE\_OF\_DELTA\_TO\_PURGE** preference value to 100. If you want to delete only 10% of the old delta files in the MMV dataset, set the **MMP\_PERCENTAGE\_OF\_DELTA\_TO\_PURGE** preference value to 10.

## Visualization Data Server status log settings

Configuration for the Visualization Data Server is available in the **etc/VisDataServer.properties** file. This includes detailed logging and fine tuning of other settings. If you make changes to the properties file, you must restart the Visualization Data Server.

Log information includes the status of all products hosted by the Visualization Data Server.

```
#
# Status logger settings. The status logger can be of help showing
# the current indexing status
# and also the current and waiting task to be processed.
#
# The interval to generate the status log (see the "Interval"
# documentation
# for more info).
StatusLogger.StatusInterval=120
# This will output the name of the top level (root) node.
StatusLogger.ShowRootName = true
# Shows the timestamp of the indexed product.
StatusLogger.ShowTimestamp = true
# Shows the available revision rules of indexed product.
StatusLogger.ShowRevRule = true
# Shows the status of the Spatial JTs.
StatusLogger.ShowSpatialJt = true
# If ShowSpatialJt is true, also shows the file path of the Spatial JTs.
StatusLogger.ShowSpatialJtPath = true
# If ShowSpatialJt is true and a Spatial JT is missing, the string will
# be added
# in from of the path.
# This can be used if a specific string is needed to search for a
# missing
# file (like using the grep utility).
StatusLogger.MissingSpatialJtMessage = (missing)
# Shows all the versions of a product instead of just the latest one.
StatusLogger.ShowAllVersions= false
# Shows the active tasks being processed.
StatusLogger.ShowActiveTasks=true
# Shows any waiting tasks to be processed.
StatusLogger.ShowWaitingTasks=true
```

## Install the Active Workspace Launcher

The Active Workspace Launcher connects Active Workspace to Microsoft Office applications, opening the appropriate Office application when you open an attachment in Active Workspace.

To enable this behavior on an Active Workspace client host:

1. Install Microsoft Office.

For supported versions, see the Hardware and Software Certifications knowledge base article on Support Center.

2. Depending on your needs, install Teamcenter Client for Microsoft Office and/or Teamcenter Extensions for Microsoft Office as described in *Teamcenter Client Installation on Windows* in the Teamcenter help.
3. Install Active Workspace Launcher:
  - a. In the Teamcenter minor release software kit, open the **wntx64\additional\_applications\TcClientAppLauncher** directory and launch **setup.exe**.
  - b. Complete the installation wizard.
4. To enable the editing of requirements in Active Workspace, you must perform additional setup tasks.

If you want to host Active Workspace within Client for Office, set Active Workspace hosting preferences as described in *Configuration and Extensibility*.

Note:

Kerberos authentication is *not* supported with Client for Office.

### Install Active Workspace Launcher silently

Alternatively, you can install the Active Workspace Launcher silently, without user interaction:

1. To generate a silent installation file, type the following command in a command prompt:

```
setup.exe /r /f1"path\tclauncher.iss"
```

For example, to generate a silent file in the **c:\temp** folder, enter the following command:

```
setup.exe /r /f1"c:\temp\tclauncher.iss"
```

Do not include a space between the **f1** argument and the path that follows it. The path must be enclosed in double quotation marks (") as shown.

- To install the Active Workspace launcher silently on another system, type the following command:

```
setup.exe /s /f1"c:\tclauncher.iss"
```

## Troubleshoot the Active Workspace Launcher installation

If Microsoft Office applications fail to launch when opening an attachment, the **.awoai** file may be associated with Microsoft Word instead of the Active Workspace Launcher. To resolve this, perform one of the following tasks:

- Uninstall and reinstall the Teamcenter Active Workspace Launcher and try again.
- In the **HKEY\_CLASSES\_ROOT\.awoai\shell\Open\command** registry entry, ensure the **.awoai** file extension is correctly associated with the **TcClientApplauncher.exe** command.

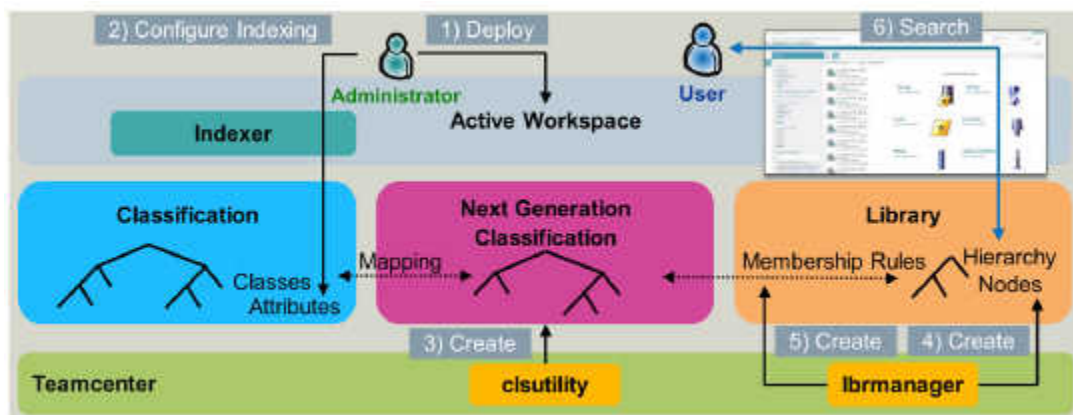
## Install classification libraries

Prerequisites:

- The Classification Library Management feature must be installed and configured on Teamcenter.
- The presentation layer must be created with the **clsutility**.
- The library data must be created with the **lbrmanager** utility.

For more information, see how to use Library Management to selectively display the classification hierarchy in *Configuration and Extensibility*.

The following displays the interaction of the various components necessary to use Classification in Active Workspace:



- Server

- To install the server components for classifying objects and searching for classified objects:  
Select **Active Workspace**→**Server Extensions**→**Reuse and Standardization**→**Classification Server**.
- To install the server components necessary for enabling visual navigation cards:  
Select **Active Workspace**→**Server Extensions**→**Reuse and Standardization**→**Next Generation Classification Server**.
- To install the server components necessary for searching within classification libraries:  
Select **Active Workspace**→**Server Extensions**→**Reuse and Standardization**→**Library Management Server**.

Functions	Classification	Server features	
		Next Generation Classification (Presentation layer)	Library Management
Index classification attributes	√	√	√
Index classification classes	√	√	√
Index classifying objects (ICOs)	√	√	
Hierarchical filtering of classes	√	√	√
Index catalog data			√
Index library elements			√
Visual navigation cards (VNC) for classes		√	√
Visual navigation cards (VNC) for library nodes			√
Compatible client feature	Classification client	Library Management client	

- Client

- To install the user interface elements for classifying objects in Active Workspace:  
Select **Active Workspace**→**Client Extensions**→**Reuse and Standardization**→**Classification Client**.
- To install the user interface components necessary for using classification libraries:  
Select **Active Workspace**→**Client Extensions**→**Reuse and Standardization**→**Library Management Client**.

Functions	Client features	
	Classification	Library Management
Classification authoring	√	
Browse classification hierarchy using visual navigation cards		√
Browse library hierarchy using visual navigation cards		√
Dedicated location for searching and browsing		√

## Verify Active Workspace installation

To verify the Active Workspace installation is complete and successful, open the Active Workspace URL in a web browser:

**`http://host:port`**

Replace *host* and *port* with the host and port of the Active Workspace Gateway.

For example:

**`http://myhost:3000`**

In the Active Workspace logon screen, enter the user name and password for the Teamcenter administrative account.

If installation is successful, the browser displays the Active Workspace client.

You can also verify the status of Active Workspace Gateway and services using the Active Workspace gateway ping:

**`http://myhost:3000/ping`**



# 4. Updating Active Workspace and microservices

## Patch Active Workspace 4.3 or later

Apply the microservice framework and Active Workspace patches on *all* Teamcenter hosts that have Teamcenter Foundation installed.

1. Make sure you have the correct **Active Workspace and microservice framework software kits**.
2. Stop all Teamcenter services except FSC services. Make sure all server manager services are stopped.
3. Launch TEM in maintenance mode for the configuration you are updating.

On Windows systems, launch TEM with administrative privileges (right-click the **tem.bat** icon→**Run as administrator**).

4. In the **Maintenance** panel, select **Updates Manager**.
5. Proceed to the **Apply Updates** panel and enter required paths:
  - In the **Update kit location** box, enter the path to the expanded microservice framework software kit.
  - In the **Backup directory** box, enter a path in which you want TEM to create backups of files replaced during patching.
  - In the **Original Media Location** box, enter the path to the software kit for the major version of Teamcenter or Teamcenter Rapid Start.
6. Proceed to the **Operating System User** panel and type the password of the operating system account used to update Teamcenter.
7. Proceed to the **Teamcenter Administrative User** panel and type the user's password.
8. Proceed to the **Confirmation** panel and click **Start** to apply the patch that adds Active Workspace to your environment.
9. When the update is complete, close TEM.
10. Repeat steps **3** through **9** to apply the Active Workspace 5.2 patch.
11. On hosts with Teamcenter Foundation installed, update the **TC\_DATA** directory:

- a. In the location of the expanded Active Workspace 5.2 software kit, browse to the **tc** directory.
- b. Expand the **data.zip** file from this location to the **TC\_DATA** folder.

If you want to add additional Active Workspace applications and components to your environment, restart TEM in maintenance mode and select additional Active Workspace **Server Extensions**, **client features**, or **microservices** in the **Features** panel.

Continue updating Active Workspace by **updating the Active Workspace environment**, and then **publishing changes to the file repository**.

## Patch Active Workspace 4.2 or earlier

Make sure you **install and configure the microservice framework** before you perform the following steps.

Apply the Active Workspace patch on *all* Teamcenter hosts that have Teamcenter Foundation installed.

1. Make sure you have the correct **Active Workspace software kit**.
2. If any server managers are running on the corporate server, shut them down before proceeding with the patch.
3. Remove your previous Active Workspace web application. Perform the appropriate step for the architecture you use:
  - Java EE: Remove the **awc.war** file and the **awc** directory under the **webapps** directory.
  - .NET: Stop the Teamcenter Active Workspace web site in Internet Information Server (IIS). Then, remove the Active Workspace web site.

**Note:**

The Active Workspace web application is not used in the **current Active Workspace architecture**.

4. Launch TEM in maintenance mode for the server you are updating.
5. In the **Maintenance** panel, select **Updates Manager**.
6. Complete the **Apply Updates** panel:
  - a. In the **Update kit location** box, enter the location of the Active Workspace 5.2 software kit.
  - b. In the **Backup directory** box, enter the location in which you want backup files to be created.

- c. In the **Original Media Location** box, enter the location of the Teamcenter major release software kit, for example, Teamcenter 13.
7. In the **Operating System User** panel, type the required password.
8. In the **Teamcenter Administrative User** panel, type the required password.
9. If the patch contains enhancements to installed features in your configuration, TEM lists the enhancements in the **Required Configuration Enhancements** panel.

Features containing data model objects may be updated, meaning database model changes may occur. Siemens Digital Industries Software recommends you back up your database before continuing.

Select the **Accept Data Model Updates** check box to continue with installation.

10. If the server you are patching contains the Indexing Engine, TEM displays the **Indexing Engine Configuration** panel.

#### SOLR schema files location

- If the Indexing Engine and the Server Extensions are installed on the same server, the **SOLR schema files location** box is prepopulated with the correct location.
- If the Indexing Engine and the Server Extensions are installed on separate servers and you have already patched the server on which the Server Extensions are installed, enter the path to the upgraded Server Extensions server that contains the Solr schema files, `TC_DATA\fts\solr_schema_files`.
- If the Indexing Engine and the Server Extensions are installed on separate servers and you have not yet patched the server on which the Server Extensions are installed, leave the **SOLR schema files location** box blank. You can patch the Indexing Engine server now, but the schemas are not automatically merged. You must manually merge them after upgrading both the Indexing Engine server and the Server Extensions server.

#### Install indexing engine as a service

Select this to install the Indexing Engine as a service. If this is not selected, you must start the Indexing Engine manually.

11. In the **Publish to Gateway** panel, deselect the **Publish to Gateway** check box.

Publishing to Active Workspace Gateway cannot be enabled until after Active Workspace microservices are installed and running.

12. In the **Database User** panel, enter the database administrator password.

13. In the **Diagnostics** panel, enter a directory for storing log files and verify that all pre-patching tests are successful.
14. In the **Confirmation** panel, click **Start**.
15. During the update, a **Status Message** popup window opens. It informs you that you must manually patch the *TC\_DATA* directory after TEM completes its updates.
16. When the update is complete, close TEM.
17. In the location of the expanded Active Workspace 5.2 software kit, browse to the **tc** directory.
18. Expand the **data.zip** file from this location to the *TC\_DATA* folder on the patched host.

If you want to add additional Active Workspace applications and components to your environment, restart TEM in maintenance mode and select additional Active Workspace **Server Extensions**, **client features**, or **microservices** in the **Features** panel.

Continue updating Active Workspace by **updating the Active Workspace environment**, and then **publishing changes to the file repository**.

## Complete the Active Workspace update

### Remerge Solr and Teamcenter schemas and update the index

After moving to the target Active Workspace and Teamcenter versions, the Teamcenter and Solr schemas are not synchronized. You can merge the schemas and then choose the method for updating the index. You must determine your indexing strategy before planning your index update.

#### Evaluate the delta of object data changes

If your indexing changes are additions, modifications, and deletions for types and properties, you can perform a delta indexing update rather than a complete reindex.

1. Merge the Teamcenter and Solr schemas.
2. Stop synchronization by the indexer if it's running.

```
runTcFTSIndexer -stop
```

3. Determine the scope of the changes between the last indexing schema and the current schema.

Run the **awindexerutil** utility using **-delta -dryrun** to get a report of the expected delta of changes. For example:

```
awindexerutil -u=adminuser -p=password -g=group -delta -dryrun
```

The differences are output to the command window as well as to a log file.

4. After you evaluate the report, determine whether you want to use the delta of changes for reindexing. If so, run **awindexerutil** to index the changes from the report:

```
awindexerutil u=adminuser -p=password -g=group -delta
```

5. Test indexer connectivity by running the indexer test flow.

```
runTcFTSIndexer -task=objdata:test
```

6. Restart the synchronization flow using the **runTcFTSIndexer** utility:

```
runTcFTSIndexer -task=objdata:sync -interval=seconds
```

## Reindex your data if needed

Your index may have a high number of changes or other kinds of changes that are outside the scope of delta changes. If that is the case, reindex your data instead:

1. Merge the Teamcenter and Solr schemas.
2. Test indexer connectivity by running the indexer test flow.

```
runTcFTSIndexer -task=objdata:test
```

3. Reindex the data.

## Additional postupdate steps

### Import table column configuration

If you have configured any custom table columns other than those available by default, perform the following after upgrading or patching:

1. Update the custom column information in the **Awb0ContentTableUiConfigCots.xml** file.
2. Run the **import\_uiconfig** utility to import the column configuration.

### Reindex classification data

If you are new to classification and want to search for classification data in Active Workspace using classification standard taxonomy, your first step is to create a classification hierarchy.

If you already use traditional classification and want to search for classification data in Active Workspace or Teamcenter Manufacturing Access, perform the following steps:

1. Create search index views and specify facetable properties.
2. Update and merge the schema file.
3. Index or reindex your data. If your data is already indexed, it is sufficient to index only the modified classification data.

These steps are described in *Classification and Library Management*

### Install workflow features

Starting in Active Workspace 3.0, workflow functionality has been modularized. For users to access **Inbox** components and workflow functionality in the Client interface, you must:

- Install the **Workflow Server** **Server Extensions feature** on the corporate server.
- Install the **Workflow** Client feature in the **Active Workspace client**.

### Install Systems Modeling and Systems Modeler features

Starting in Active Workspace 3.0, some functionality in the **Architecture** toolbar of the Client interface has been modularized. This functionality includes the **Start Authoring** in the **Architecture** toolbar. To restore this functionality after moving to Active Workspace 5.2:

- Install the **Systems Modeling** **Server Extensions** feature on the corporate server.
- Install the **Systems Modeler** **Client** feature in the Active Workspace client.

### Set up a custom template after patching

If you have a custom template that is dependent on the Active Workspace template (**aws2**) and other Active Workspace templates, you must follow a special procedure to set up this template after patching Active Workspace.

1. Add the Active Workspace template (**aws2**) any other needed Active Workspace templates to the Business Modeler IDE by selecting **Add/Update Templates for working within the Business Modeler IDE Client** in the **Feature Maintenance** panel of Teamcenter Environment Manager (TEM).
2. Run TEM and re-patch Active Workspace.

This copies the required Business Modeler IDE *.jar* files for Active Workspace into the installation.

3. In the Business Modeler IDE, import your custom template. To add the Active Workspace template (**aws2**) and the other dependent Active Workspace templates:
  - a. Right-click your custom template project and choose **Properties**.
  - b. In the left pane, choose **Teamcenter**→**BMIDE**.
  - c. In the **Dependent Templates** pane, select the **aws2** template and the other Active Workspace templates previously added using the **Add/Update Templates for working within the Business Modeler IDE Client** option in TEM.
  - d. Click **OK**.

The template is now ready to be deployed.

### Restart Teamcenter Process Manager

After completing updates to Active Workspace, restart the Teamcenter Process Manager service in the **Services** dialog of the Windows Control Panel. This allows the updates made to the Teamcenter GraphQL Service to take effect.

Some Active Workspace features such as Discussions (Active Collaboration) and Reports rely on updates to the Teamcenter GraphQL Service in order to function properly.

## Publishing changes to the Active Workspace file repository

After patching, upgrading, or adding features to the Active Workspace client, publish changes to the Active Workspace file repository.

1. Start TEM on the Active Workspace Gateway host.
2. In the **Maintenance** panel, select **Configuration Manager**.
3. In the **Configuration Maintenance** panel, select **Perform maintenance on an existing configuration**.
4. In the **Old Configuration** panel, select the configuration.
5. In the **Feature Maintenance** panel, select **Update Active Workspace client settings**.
6. In the **Active Workspace Client Settings** panel, type the URL to the Active Workspace Gateway and enable publishing of client content to the Gateway.

Value	Description
<b>Publish to Gateway</b>	Select this check box to enable automatic publishing of Active Workspace content to the Gateway.
<b>Gateway URL</b>	Type the URL to Active Workspace Gateway:  <b><code>http://host:port</code></b>  Replace <i>host</i> with the host on which you installed the Gateway. Replace <i>port</i> with the port used by the Gateway. The default port is <b>3000</b> .

7. In the **Confirmation** panel, click **Start**.
8. When the installation is complete, close TEM.



# A. Active Workspace features reference

## Active Workspace Server Extensions features

Active Workspace Server Extensions features are available in the **Features** panel in Teamcenter Environment Manager (TEM), under **Base Install**→**Active Workspace**→**Server Extensions**.

To search for a feature by name, type the name or a partial name in the search box, and then click the search button.

Feature	Description
<b>Active Workspace</b>	Adds binaries to <i>TC_ROOT</i> and data model changes to the database and <i>TC_DATA</i> . It adds the <b>Active Workspace</b> ( <i>aw3_template.xml</i> ) template to the database.
<b>Active Architect Core</b>	Installs core capabilities of Active Architect.
<b>Active Collaboration</b>	Allows users to communicate between themselves using Active Workspace.  Adds the <b>Active Collaboration</b> ( <i>ac0activecollaboration_template.xml</i> ) template to the database.  If you are installing this feature, you should also install the <b>Active Collaboration</b> feature (under <b>Client</b> ) when building the Client web application.
<b>Active Collaboration for Retail Solution</b>	Installs server support for <i>Active Collaboration</i> for Retail Footwear and Apparel users.
<b>Active Content Structure</b>	If you plan to index structure data, you must install this feature. It provides functionality and data model extensions necessary for indexing structure data. It adds the <b>Active Content Structure</b> ( <i>activeworkspacebom_template.xml</i> ) template to the database.  If you are installing this feature, you should also install the <b>Active Content</b> feature (under <b>Client</b> ) when building the Client web application.
<b>Advanced PLM Services Core</b>	Adds Advanced PLM Services capabilities to 4th Generation Design in Active Workspace.
<b>Active Workspace Assistant</b>	Installs server support for the Assistant, a guided navigation tool for Active Workspace. It predicts the next likely actions a user will perform based on the user's history, group, and role, and presents suggested actions in the <b>Assistant</b> panel. This feature requires the <b>Command Prediction Service</b> microservice and a database for the microservice to store data for the Assistant. TEM prompts you for the necessary database creation values during installation.

Feature	Description
	To enable client support for the Assistant, install the <b>Active Workspace Assistant</b> client feature.
<b>Active Workspace Document Management</b>	Adds support for document management in Active Workspace.
<b>Audit</b>	Adds Audit Manager capabilities to Active Workspace. For more information, see <i>Audit Manager</i> in the Teamcenter help collection.
<b>Authorization Active Workspace</b>	Adds Active Workspace authorization components.
<b>Digital Signatures</b>	Adds digital signatures functionality to the server. It does not add a database template.  Before selecting this feature, see the additional setup requirements for digital signatures.  If you are installing this feature, you should also install the <b>Digital Signatures</b> feature (under <b>Client</b> ) when building the Client web application.
<b>Google Online Viewer</b>	Installs server support for the Google Online Viewer, which enables Google Online features within Active Workspace, such as viewing and editing Microsoft Office documents without the need for Microsoft Office applications.
<b>In-Context Change</b>	Adds in-context change support to Active Workspace.
<b>Multisite Integration</b>	Adds multisite reporting that enables administrators to identify underlying data issues before multisite import or export.
<b>Office Online Viewer</b>	Installs viewer support for Microsoft Office Online documents.
<b>Program Planning</b>	Enables the Program Management capability in Active Workspace. This feature provides the ability to manage business investments, from planning to execution, in terms of time, reuse, volume, cost targets, and weight targets.  This feature adds the <b>Program Planning for Active Workspace (pgp0awprgplanning_template.xml)</b> template to the database.  This feature is not selectable unless the <b>Program Planning Infrastructure</b> feature (under <b>Extensions</b> ) is also selected.  If you are installing this feature, you should also install the <b>Program Planning Client</b> feature (under <b>Client</b> ) when building the Client web application.
<b>Relationship Viewer</b>	Adds the <b>Relationship Viewer (relationshipviewer_template.xml)</b> template to the database.

Feature	Description
<b>Reporting</b>	<p>If you are installing this feature, you should also install the <b>Relationship Browser</b> feature (under <b>Client</b>) when building the Client web application.</p> <p>Provides the ability to view report templates, generate reports based on selected criteria, style sheets, or both, and view them in HTML, Excel, or raw XML formats. It adds the <b>Reporting for Active Workspace</b> (<b>rb0reportingaw_template.xml</b>) template to the database.</p> <p>If you are installing this feature, you should also install the <b>Reporting</b> feature (under <b>Client</b>) when building the Client web application.</p> <p>Additionally, to allow for asynchronous report generation, install the <b>AsyncService</b> translator in the Dispatcher Server.</p>
<b>Schedule Manager</b>	<p>Enables Schedule Manager capabilities in Active Workspace.</p> <p>It adds the <b>Schedule Manager for Active Workspace</b> (<b>saw1projectmanagementaw_template.xml</b>) template to the database.</p> <p>If you are installing this feature, you must also install the <b>Schedule Manager</b> feature (under <b>Client</b>) when building the Client web application.</p>
<b>Shape Search</b>	<p>Provides the binaries and data model extensions for the shape search functionality. It adds the <b>Shape Search for Active Workspace</b> (<b>shapesearch_template.xml</b>) template to the database. To use shape search, Geolus must be installed and configured.</p> <p>If you are installing this feature, you should also install the <b>Shape Search</b> feature (under <b>Client</b>) when building the Client web application.</p>
<b>Stock Material</b>	<p>Adds stock material management to Active Workspace.</p> <p>Many parts are made from stock materials such as bar stock, tubing stock and sheet stock. This features enables you to manage stock materials in Teamcenter, performing actions like creating libraries of stock materials and assigning stock materials to parts.</p> <p>For more information, see <i>Aerospace and Defense Solution</i> in the Teamcenter help collection.</p>
<b>Subscription</b>	<p>Allows users to manage subscriptions and notifications. It adds the <b>Subscription</b> (<b>sub0subscription_template.xml</b>) template to the database.</p> <p>If you are installing this feature, you must also install the <b>Subscription</b> feature (under <b>Client</b>) when building the Client web application.</p>
<b>Work Package Management</b>	<p>Adds work package management to Active Workspace.</p> <p>A work package or package is a collection of CAD files and documentation that an outsourcing partner uses for building, testing, or maintaining a component or subassembly of a larger product.</p>

Feature	Description
	<p>Teamcenter helps to create and maintain the package as a revisable collection or a container of product information and to use in a variety of contexts.</p> <p>For more information, see <i>Aerospace and Defense Solution</i> in the Teamcenter help collection.</p>
<b>Active Workspace Requirements and Systems Engineering</b>	Adds support for Requirements and Systems Engineering to Active Workspace.
<b>Active Workspace Usage BOM</b>	Adds the BOM management overlay for Active Workspace.
<b>Concurrent Modeling</b>	Adds concurrent modeling support in Active Workspace. Concurrent modeling allows you to manage models from supported 1D modeling tools in Teamcenter, using the Teamcenter MBSE Integration Gateway MBSE framework.
<b>Contract Data Management</b>	<p>Installs Contract Data Management, which helps contractors manage the creation, review, and delivery of contracts. A <i>contract</i> is a structured procurement document that lists milestones and schedule dates.</p> <p>To enable this feature to work in Active Workspace, select the <b>Contract Data Management</b> feature under <b>Enterprise Knowledge Foundation</b>.</p>
<b>Core Services for Manufacturing on BVR</b>	Installs the manufacturing core for the BOM view revisions (BVR) data model and adds the <b>mbc0mfgbvrcore_template.xml</b> template.
<b>DPV Active Workspace</b>	Installs the database configuration for DPV in Active Workspace. This feature requires corresponding Teamcenter DPV features.
<b>EDA Server Support for Active Workspace</b>	<p>Adds support for Electronic Design Automation (EDA) to Active Workspace.</p> <p>For information about installing EDA, see the EDA documentation on Support Center.</p>
<b>Manufacturing Resource Manager (MRL)</b>	<p>Adds Manufacturing Resource Library support for Active Workspace.</p> <p>For information about using Manufacturing Resource Library, see the Teamcenter help library.</p>
<b>Material Management</b>	<p>Enables Active Workspace users to:</p> <ul style="list-style-type: none"> <li>• Associate a material revision with a vendor part or an item revision.</li> <li>• Associate a substance with a material revision.</li> <li>• View the Material/Substance Dashboard.</li> </ul>

Feature	Description
	<p>This feature adds the <b>Material Management for Active Workspace</b> (<b>mtw0materialmgmtaw_template.xml</b>) template to the database.</p> <p>This feature is not selectable unless the <b>Material Management</b> feature (under <b>Extensions</b>→<b>Enterprise Knowledge Foundation</b>) and <b>Active Content Structure</b> (under <b>Server Extensions</b>) are also selected.</p> <p>If you install this feature, you should also install the <b>Material Management</b> feature (under <b>Client</b>) when building the Client web application.</p>
<b>NX for Active Workspace</b>	<p>Enables users to access NX integration functionality from the client interface. It adds the <b>NX for Active Workspace</b> (<b>awn0nx_template.xml</b>) template to the database.</p> <p>This feature is not selectable unless the <b>NX Foundation</b> feature (under <b>Extensions</b>→<b>Teamcenter Integration for NX</b>) is also selected.</p> <p>If you are installing this feature, you should also install the <b>NX Integration</b> feature (under <b>Client</b>) when building the Client web application.</p>
<b>Part Manufacturing Active Workspace</b>	Adds Part Manufacturing support to Active Workspace.
<b>Vendor Management</b>	<p>Allows Active Workspace users to:</p> <ul style="list-style-type: none"> <li>• Associate vendors with vendor parts.</li> <li>• Associate vendor parts with commercial part revisions.</li> </ul> <p>This feature adds the <b>Vendor Management for Active Workspace</b> (<b>vm1vendormanagementaw_template.xml</b>) template to the database.</p> <p>This feature is not selectable unless the <b>Active Content Structure</b> feature and the <b>Vendor Management</b> feature (under <b>Extensions</b>→<b>Supplier Relationship Management</b>) are selected.</p> <p>If you are installing this feature, you should also install the <b>Vendor Management</b> feature (under <b>Client</b>) when building the Client web application.</p>
<b>Visualization Extension</b>	<p>This feature is required to launch the Teamcenter lifecycle visualization stand-alone application viewer from the Client interface.</p> <p>This feature enables Active Workspace 3D viewer functionality. It adds the <b>Active Workspace Visualization</b> (<b>awv0activeworkspacevis_template.xml</b>) template to the database.</p> <p>You must install this feature if you install the <b>Visualization Server Manager</b> feature (under <b>Visualization Server</b>) when building the Client web application.</p>

Feature	Description
	The <b>Visualization Extension</b> feature is not selectable unless the <b>Active Content Structure</b> feature is also selected.
<b>Workflow</b>	<p>Workflow does not add a template to the database.</p> <p>If you are installing this feature, you must also install the <b>Workflow</b> feature when building the Client web application.</p> <p>For users to access <b>Inbox</b> components and workflow functionality in the Client interface, <b>Workflow</b> must be installed on the corporate server and <b>Workflow</b> must be installed in the Client web application.</p>
<b>Easy Plan Graphics</b>	<p>Adds support for graphics capabilities used in the Work Instructions Authoring workspace.</p> <p>This feature is automatically selected when you select the Easy Plan Work Instructions Authoring option in the Features panel.</p>
<b>Electronic Work Instructions</b>	Installs the server side of Electronic Work Instructions in your environment.
<b>Finish Management</b>	<p>Installs Finish Management support for Active Workspace. A <i>finish</i> represents a finishing process on a part. It may be used to improve appearance, adhesion, corrosion resistance, tarnish resistance, chemical resistance, wear resistance, and remove burrs and so on.</p> <p>For more information, see <i>Aerospace and Defense Solution</i> in the Teamcenter help collection.</p>
<b>Product Configurator</b>	<p>Adds Product Configurator functionality for Active Workspace. Product Configurator enables you to formally introduce and manage variability across your product suite.</p> <p>For more information, see <i>Product Configurator</i> in the Teamcenter help collection.</p>
<b>Requirements Management</b>	<p>Allows users to author a requirement structure in the <b>Content</b> tab of the Client. Adds the <b>Active Workspace Requirements Management</b> (<code>arm0activeworkspacereqmgmt_template.xml</code>) template to the database.</p> <p>This feature is not available with Teamcenter Rapid Start.</p> <p>The <b>Requirements Management</b> feature is not selectable unless the <b>Active Content Structure</b> feature is also selected.</p> <p>If you are installing this feature, you must also install:</p> <ul style="list-style-type: none"> <li>• <b>Server Extensions→Workflow</b></li> <li>• <b>Extensions→Systems Engineering and Requirements Management→Requirements Management</b> feature on the corporate server.</li> </ul>

Feature	Description
<b>Systems Engineering</b>	<ul style="list-style-type: none"> <li>• <b>Requirements Management</b> feature when building the Client web application.</li> </ul> <p>Allows users to manage Systems Engineering diagrams from the Client. It adds the <b>Active Workspace Systems Engineering</b> (<code>ase0activeworkspacesyseng_template.xml</code>) template to the database.</p> <p>This feature is not available with Teamcenter Rapid Start.</p> <p>The <b>Systems Engineering</b> feature is not selectable unless the <b>Active Content Structure</b> feature is also selected.</p> <p>If you are installing this feature, you must also install:</p> <ul style="list-style-type: none"> <li>• <b>Extensions→Systems Engineering and Requirements Management→Systems Engineering</b> feature on the corporate server.</li> <li>• <b>Requirements Management</b> feature when building the Client web application.</li> </ul>
<b>Teamcenter Manufacturing Access</b>	<p>Adds support for Teamcenter Manufacturing Access to Active Workspace.</p> <p>Teamcenter Manufacturing Access provides solutions for various manufacturing planning tasks and business processes. It is designed as a cross-industry tool; use it in any environments where Process Simulate is used.</p> <p>For more information about Teamcenter Manufacturing Access, see Easy Plan help.</p>
<b>4th Generation Foundation</b>	<p>Installs 4th Generation Design (4GD) functionality for the Teamcenter server. 4GD allows users of NX CAD or Lifecycle Visualization to cooperate in real time during the design cycle of a product.</p> <p>This feature requires corresponding Teamcenter features.</p>
<b>Easy Plan - Process Planning for BTO/BTS/CTO</b>	<p>Installs the Process Planning workspace for build-to-order (BTO) or build-to-stock (BTS), or configure-to-order (CTO) in your environment.</p> <p>Adds these templates to the database:</p> <ul style="list-style-type: none"> <li>• Manufacturing core for BVR/Data Model (<code>mbc0mfgbvrcore_template.xml</code>)</li> <li>• Process Planning Change Management/Data Model (<code>epc0mfgbvrmaturity_template.xml</code>)</li> <li>• Process Planning Line Balancing/Data Model (<code>elb0linebalancing_template.xml</code>)</li> <li>• Process Planning Time Analysis/Data Model (<code>ept0timeanalysis_template.xml</code>)</li> </ul>

Feature	Description
<b>Easy Plan - Product Configurator Extension</b>	<p>Adds support for Product Configurator in Easy Plan. Product Configurator allows you to enable definitions for variant formulas on operations.</p> <p>Please see the Product Configurator (Administrator) documentation for installation details.</p> <p>To use Product Configurator variants, you must run Teamcenter 11.5 or later. By default, Easy Plan uses classic variants.</p>
<b>Embedded Software Management for Active Workspace</b>	Installs embedded software management capabilities for Active Workspace.
<b>Manufacturing 4G Core</b>	Installs the interface between Next Generation Planning (NGP) and Teamcenter, and adds the <b>mac0mfgapscore_template.xml</b> template.
<b>NX Integration for Model-Based Systems Engineering</b>	Adds the NX Integration for Model-Based Systems Engineering.
<b>Requirements Management - Quality Module</b>	Adds Teamcenter Quality functions to Requirements Management.
<b>Systems Modeling</b>	<p>Install this feature so that users can access functionality in the <b>Architecture</b> toolbar of the Client interface.</p> <p>This feature adds the <b>Systems Modeling</b> (<b>ase1systemsmodeling_template.xml</b>) template to the database.</p> <p>To enable the <b>Systems Modeling</b> feature, first select these features:</p> <ul style="list-style-type: none"> <li>• <b>Base Install→Active Workspace→Server Extensions→Active Workspace</b></li> <li>• <b>Extensions→Systems Engineering and Requirements Management→Systems Engineering</b></li> <li>• <b>Base Install→Active Workspace→Server Extensions→Active Content Structure</b></li> <li>• <b>Base Install→Active Workspace→Server Extensions→Systems Engineering</b></li> </ul> <p>You can then select the <b>Systems Modeling</b> feature.</p> <div style="border: 1px solid black; padding: 10px; margin-top: 10px;"> <p>Note:</p> <p>Selecting <b>Systems Modeling</b> automatically selects these features under <b>Extensions→Model-Based Systems Engineering</b>:</p> <ul style="list-style-type: none"> <li>• <b>Attribute and Parameter Base Definitions</b></li> </ul> </div>



Feature	Description
	<ul style="list-style-type: none"> <li>• <b>Measurable Attributes and Targets</b></li> </ul>
<b>4th Generation Design</b>	<p>Allows users to view, navigate, and configure collaborative designs and their content in the Client.</p> <p>It adds the <b>4th Generation Foundation</b> (<code>fgd0aw4gfoundation_template.xml</code>) and <b>4th Generation Design</b> (<code>fgd0aw4gdesign_template.xml</code>) templates to the database.</p> <p>The <b>4th Generation Design</b> feature is not selectable unless the <b>Active Content Structure</b> feature (under <b>Server Extensions</b>) and the <b>4th Generation Design</b> feature (under <b>Extensions</b>→<b>Advanced PLM Services</b>) are also selected.</p> <p>If you are installing this feature, you must also install the <b>4th Generation Design</b> feature when building the Client web application.</p>
<b>4th Generation Process Planning</b>	<p>Adds Advanced PLM Services capabilities to 4th Generation Design in Active Workspace.</p>
<b>Electronics Process Planner</b>	<p>Installs the Easy Plan Electronics Process Planner workspace in your environment.</p> <p>Electronics engineers use Production Process Planning for the systems integration of the electronics process planning tasks of Easy Plan and those of Valor Process Preparation Software. The systems integration is available as a dedicated, optional feature of Production Process Planning.</p> <p>Adds these templates to the database:</p> <ul style="list-style-type: none"> <li>• Manufacturing core for BVR/Data Model (<code>mbc0mfgbvrcore_template.xml</code>)</li> <li>• Production Process Planning Change Management/Data Model (<code>epc0mfgbvrmaturity_template.xml</code>)</li> <li>• Production Process Planning Line Balancing/Data Model (<code>elb0linebalancing_template.xml</code>)</li> <li>• Production Process Planning Time Analysis/Data Model (<code>ept0timeanalysis_template.xml</code>)</li> </ul>
<b>Manufacturing BOM Manager</b>	<p>Adds Multi-BOM Manager support to Active Workspace. Multi-BOM Manager enables you to link and assign content across representations for different lifecycle stages or uses for a product.</p> <p>For information about using this feature, see <i>Mutli-BOM Manager</i> in the Teamcenter help library.</p>

Feature	Description
<b>Partitions for Structure</b>	The <b>Partitions for Structure</b> feature is required to get the partitions that are in turn required to organize the contents of a product structure.
<b>Process Planning and Work Instructions authoring for ETO</b>	<p>Installs support for the Engineering to Order (ETO) workspace in your environment.</p> <p>This feature allows end users to author work instructions at the process station level.</p>
<b>NX P &amp; ID Active Workspace</b>	<p>Exposes NX P &amp; ID in Active Workspace so that users can view, navigate, and configure sheets and their content.</p> <p>This feature adds the <b>NX P &amp; ID Active Workspace</b> (<code>npx1pidaws_template.xml</code>) template to the database.</p> <p>This feature is not selectable unless these features are also selected:</p> <ul style="list-style-type: none"> <li>• <b>NX Foundation</b> under <b>Extensions</b>→<b>Teamcenter Integration for NX</b></li> <li>• <b>Active Content Structure</b> under <b>Server Extensions</b></li> <li>• <b>4th Generation Design BOM Management</b> under <b>Extensions</b>→<b>BOM Management</b></li> <li>• <b>MDConnectivity</b> under <b>Extensions</b>→<b>Advanced PLM Services</b></li> <li>• <b>System Modeling</b> under <b>Extensions</b>→<b>Advanced PLM Services</b></li> <li>• <b>Diagramming</b> under <b>Extensions</b>→<b>Advanced PLM Services</b></li> <li>• <b>NX Piping and Instrument Diagram (P&amp;ID) Design</b> under <b>Extensions</b>→<b>Teamcenter Integration for NX</b></li> <li>• <b>Systems Engineering</b> under <b>Server Extensions</b></li> </ul> <p>If you are installing this feature, you should also install the <b>NX Integration</b> feature (under <b>Client</b>) when building the Client web application.</p>
<b>Capital Asset Lifecycle Management AW</b>	Adds management of plant data to Active Workspace.
<b>Change Management</b>	Adds support for Change Management in Active Workspace. This feature requires corresponding Teamcenter Change Management features.
<b>Active Admin</b>	Features to support Active Admin capabilities.
<b>Active Admin Core</b>	Installs essential support for the active admin workspace.

Feature	Description
<b>Active Workspace User Management</b>	Adds support for user management in Active Workspace.
<b>Preference Management</b>	Adds server support for preference management in Active Admin. Preference management features in the active admin workspace are added by the corresponding Active Workspace client feature.
<b>Viewer Administration</b>	Adds viewer administration to the Active Admin workspace in Active Workspace.
<b>Active Architect</b>	Features to support Active Architect capabilities.
<b>Client Configuration</b>	Adds client configuration to the Active Admin workspace in Active Workspace.
<b>Logical Object</b>	Adds logical objectsupport to the Active Admin workspace in Active Workspace.
<b>XRT Editor</b>	Adds XRT Editor to the Active Admin workspace in Active Workspace.
<b>Aerospace and Defense</b>	<p>Aerospace and Defense features. These allow Active Workspace users to:</p> <ul style="list-style-type: none"> <li>• Create and view parts list: technical documents, parts, drawings, and designs.</li> <li>• Work with notes: create standard notes and custom notes and attach them to technical documents, parts, drawings, and designs.</li> <li>• Work with Aerospace and Defense-related changes: create, edit, and search change requests, change notices, and deviation requests in alignment with the out-of-the-box Change Management functionality.</li> </ul>
<b>Aerospace and Defense Foundation</b>	<p>Adds the Aerospace and Defense Foundation Active Workspace (<b>ads1awadsfoundation_template.xml</b>) template to the database. This feature is not selectable unless the following features are also selected:</p> <ul style="list-style-type: none"> <li>• <b>Teamcenter Change Management</b></li> <li>• <b>Teamcenter Change Management (for Active Workspace)</b></li> <li>• <b>Active Workspace Indexing Engine</b> under <b>Active Workspace</b>→<b>Indexing Server</b></li> <li>• <b>Active Workspace Indexer</b> under <b>Active Workspace</b>→<b>Indexing Server</b></li> <li>• <b>Active Workspace</b> under <b>Active Workspace</b>→<b>Server Extensions</b></li> <li>• <b>Active Content Structure</b> under <b>Active Workspace</b>→<b>Server Extensions</b></li> <li>• <b>Vendor Management</b> under <b>Active Workspace</b>→<b>Server Extensions</b></li> <li>• <b>Aerospace and Defense Foundation</b> under <b>Active Workspace</b>→ <b>Server Extensions</b>→<b>Aerospace and Defense</b></li> <li>• <b>Aerospace and Defense Foundation</b> under <b>Active Workspace</b>→<b>Client</b></li> <li>• <b>Active Workspace Client</b> under <b>Active Workspace</b>→<b>Client</b></li> </ul>

Feature	Description
	<ul style="list-style-type: none"> <li>• <b>Aerospace and Defense Change Management</b> under <b>Active Workspace→Client</b></li> </ul>
<b>Aerospace and Defense Change Management</b>	Installs the change management functionality for the <b>Aerospace and Defense Foundation</b> feature. This feature requires corresponding Teamcenter features.
<b>CAE Simulation Management</b>	Features to support management of computer-aided engineering (CAE) data.
<b>Simulation Process Management</b>	<p>Allows Active Workspace users to author Simulation structures in Active Workspace.</p> <p>It adds the <b>Simulation Process Management Server</b> (<code>cae1caeaws_template.xml</code>) template to the database.</p> <p>The <b>Simulation Process Management</b> feature is not selectable unless the <b>Active Content Structure</b> feature and the <b>Simulation Process Management</b> feature (under <b>Extensions</b>) are also selected.</p> <p>If you are installing this feature, you must also install the <b>Simulation Process Management Client</b> feature when building the Client web application.</p>
<b>Extended Simulation Process Management</b>	Add extended Simulation Process and Data Management capabilities to Active Workspace.
<b>Integrated Program Planning and Execution</b>	Integrated Program Planning and Execution (IPP&E) server extensions features. The IPP&E solution allows project planning that integrates cost, schedule, risk and technical requirements in a fully planned, resourced, and budgeted program. It allows configuration control, not only of products, but also of the project plan. It also communicates the status of requirements to users.
<b>Organization Breakdown Structure</b>	Adds Organization Breakdown Structure (OBS) support to IPP&E.
<b>Work Breakdown Structure</b>	Adds support for authoring Work Breakdown Structure (WBS) hierarchy in Active Workspace from top to bottom. For information about defining work breakdowns, see <i>Schedule Manager</i> in the Teamcenter help library.
<b>IPP&amp;E Foundation</b>	Provides essential functionality for Integrated Program Planning and Execution in Active Workspace.
<b>IPP&amp;E Contract Data Management Extension</b>	Adds Contract Data Management support to IPP&E.
<b>MRO</b>	Service Lifecycle Management features for Active Workspace.

Feature	Description
<b>MRO Core</b>	Provides core Service Lifecycle Management functionality for Active Workspace.
<b>Service Work Instructions</b>	Adds support for work instructions to Service Lifecycle Management in Active Workspace.
<b>As-Built for Active Workspace</b>	<p>Provides searching and BOM extensions necessary to support MRO As-Built capabilities.</p> <p>This feature is not available with Teamcenter Rapid Start.</p> <p>In addition to the <b>Active Workspace</b> feature, this feature requires:</p> <ul style="list-style-type: none"> <li>• <b>Extensions→Maintenance Repair and Overhaul→As-Built Management</b></li> </ul> <p>Adds these templates to the database:</p> <ul style="list-style-type: none"> <li>• <b>MRO Core, Active Workspace BOM Interface</b> (smr1mrocoreaw_template.xml)</li> <li>• <b>As-Built</b> (sab1asbuiltaw_template.xml)</li> </ul>
<b>As-Maintained</b>	<p>Provides searching and BOM extensions necessary to support MRO As-Maintained capabilities.</p> <p>This feature is not available with Teamcenter Rapid Start.</p> <p>In addition to the <b>Active Workspace</b> feature, this feature requires:</p> <ul style="list-style-type: none"> <li>• <b>Extensions→Maintenance Repair and Overhaul→As-Maintained Management</b></li> </ul> <p>Adds these templates to the database:</p> <ul style="list-style-type: none"> <li>• <b>MRO Core, Active Workspace BOM Interface</b> (smr1mrocoreaw_template.xml)</li> <li>• <b>As-Maintained</b> (sam1asmaintainedaw_template.xml)</li> </ul>
<b>Service Engineering</b>	Provides service engineering support for Service Lifecycle Management.
<b>Transaction Processing</b>	<p>Installs transaction processing functionality for Service Request Manager.</p> <p>This feature requires corresponding Teamcenter features.</p>
<b>Service Processing</b>	<p>Installs service processing capability for Service Request Manager.</p> <p>This feature requires corresponding Teamcenter features.</p>

Feature	Description
<b>Service Event</b>	<p>Provides searching and BOM extensions necessary to support MRO Service Event Management capabilities.</p> <p>This feature is not available with Teamcenter Rapid Start.</p> <p>In addition to the <b>Active Workspace</b> feature, this feature requires:</p> <ul style="list-style-type: none"> <li>• <b>Base Install→Active Workspace→Server Extensions→MRO→As-Maintained</b></li> <li>• <b>Extensions→Maintenance Repair and Overhaul→As-Maintained Management</b></li> <li>• <b>Extensions→Maintenance Repair and Overhaul→Service Event Management</b></li> </ul> <p>Adds these templates to the database:</p> <ul style="list-style-type: none"> <li>• <b>Transaction Processing, Active Workspace BOM Interface</b> (stp1transactionprocessingaw_template.xml)</li> <li>• <b>Service Processing, BOM Interface</b> (spr1serviceprocessingaw_template.xml)</li> <li>• <b>Service Event</b> (sem1serviceeventmgmtaw_template.xml)</li> </ul>
<b>Service Forecasting</b>	<p>Installs service forecasting capability for Service Request Manager.</p> <p>This feature requires corresponding Teamcenter features.</p>
<b>Service Planning and Service Processing Alignment</b>	<p>Installs the Service Planning functionality for Service Processing in Active Workspace.</p>
<b>Program Planning Execution</b>	<p>Program Planning Execution features for Active Workspace</p>
<b>Change Management Schedule Manager</b>	<p>Allows interaction between Schedule Manager and Change Management in Active Workspace. It allows Active Workspace users to relate schedules and change objects.</p> <p>This feature adds the <b>Change Management Schedule Manager</b> (csi1cmsmawinterface_template.xml) template to the database.</p> <p>This feature is not selectable unless these features are also selected:</p> <ul style="list-style-type: none"> <li>• <b>Change Management</b> under <b>Extensions→Enterprise Knowledge Foundation</b></li> </ul>

Feature	Description
	<ul style="list-style-type: none"> <li>• <b>Schedule Manager</b> under <b>Server Extensions</b></li> </ul> <p>If you are installing this feature, you should also install the <b>Change Management Schedule Manager</b> feature (under <b>Client→Program Planning Execution Client</b>) when building the Client web application.</p>
<b>Program Change</b>	<p>Allows interaction between Program Planning Event Change and Change Management in Active Workspace. It allows Active Workspace users to relate programs, projects, and subprojects to change objects.</p> <p>This feature adds the <b>Program Change (pch0pchinterface_template.xml)</b> template to the database.</p> <p>This feature is not selectable unless these features are also selected:</p> <ul style="list-style-type: none"> <li>• <b>Change Management</b> under <b>Extensions→Enterprise Knowledge Foundation</b></li> <li>• <b>Program Planning</b> under <b>Server Extensions</b></li> <li>• <b>Program Planning Infrastructure</b> under <b>Extensions</b></li> </ul> <p>If you are installing this feature, you should also install the <b>Program Change Client</b> feature (under <b>Client→Program Planning Execution Client</b>) when building the Client web application.</p>
<b>Program Planning Event Change</b>	<p>Allows interaction between Program Planning Schedule Manager and Program Change Interface in Active Workspace. It allows Active Workspace users to relate events to change objects.</p> <p>This feature adds the <b>Program Planning Event Change (pec0ppeventchange_template.xml)</b> template to the database.</p> <p>This feature is not selectable unless these features are also selected:</p> <ul style="list-style-type: none"> <li>• <b>Change Management</b> under <b>Extensions→Enterprise Knowledge Foundation</b></li> <li>• <b>Program Planning Infrastructure</b> under <b>Extensions</b></li> <li>• <b>Program Planning</b> under <b>Server Extensions</b></li> <li>• <b>Program Change</b> under <b>Server Extensions</b></li> </ul> <p>If you are installing this feature, you should also install the <b>Program Planning Event Change Client</b> feature (under <b>Client→Program Planning Execution Client</b>) when building the Client web application.</p>

Feature	Description
<b>Program Schedule Manager</b>	<p>Allows interaction between Schedule Manager and Program Planning in Active Workspace. It allows Active Workspace users to create plan level items to schedules and supports automatic generation of schedules within a program.</p> <p>This features adds the <b>Program Schedule Manager</b> (<code>psi0ppsminterface_template.xml</code>) template to the database.</p> <p>This feature is not selectable unless these features are also selected:</p> <ul style="list-style-type: none"> <li>• <b>Change Management</b> under <b>Extensions</b>→<b>Enterprise Knowledge Foundation</b></li> <li>• <b>Program Planning Infrastructure</b> under <b>Extensions</b></li> <li>• <b>Program Planning</b> under <b>Server Extensions</b></li> <li>• <b>Schedule Manager</b> under <b>Server Extensions</b></li> <li>• <b>Program Change</b> under <b>Server Extensions</b></li> <li>• <b>Program Planning Event Change</b> under <b>Server Extensions</b></li> </ul> <p>If you are installing this feature, you should also install the <b>Program Schedule Manager Client</b> feature (under <b>Client</b>→<b>Program Planning Execution Client</b>) when building the Client web application.</p>
<b>Reuse and Standardization</b>	Features to support Reuse and Standardization in Active Workspace.
<b>Classification Server</b>	Allows users to access Classification data in the client interface.
<b>Classification AI</b>	<p>Installs the Classification artificial intelligence (AI) engine, which provides assistance in navigating to desired classes. Configuring classification artificial intelligence</p> <p>After being trained on a database, the engine receives object metadata from the Teamcenter server and returns the probabilities for potential classes. You can specify which classes are displayed in the user interface based on these probabilities.</p>
<b>Presentation Layer - Next Generation Classification Server</b>	Installs the Next Generation Classification foundation feature. This option installs the presentation layer for classification standard taxonomy and for library management.



Feature	Description
<b>Library Management Server</b>	Enables indexing for Library Management business objects. The <b>Library Management Server</b> feature is not selectable unless the <b>Library Management</b> feature (under <b>Extensions</b> → <b>Reuse and Standardization</b> ) is also selected.
<b>Teamcenter Quality data model</b>	Features to support Teamcenter Quality.
<b>Quality Action Management data model</b>	<p>Adds support for quality actions to Active Workspace. This feature is mandatory for all Teamcenter Quality solutions.</p> <p>For more information, see <i>Teamcenter Quality</i> in the Active Workspace help.</p>
<b>Training and Qualification</b>	Adds training and qualification actions to the Teamcenter Quality module, which allows you to create and manage qualifications, qualification profiles, qualification records, and related objects.
<b>Quality Manager data model</b>	<p>Adds support for Quality Manager to Active Workspace.</p> <p>For more information, see <i>Teamcenter Quality</i> in the Active Workspace help.</p>
<b>Control and Inspection Plan</b>	<p>Adds support for Control Plan to Teamcenter Quality.</p> <p>For more information, see <i>Teamcenter Quality</i> in the Active Workspace help.</p>
<b>Escalation Management Translator</b>	<p>Adds support for escalation workflows for quality actions.</p> <p>These allow responsible users or quality managers to receive alerts or e-mail notifications to remind them of their quality actions prior to the due date, thereby avoiding unnecessary delays or escalations. Additionally, escalation workflows can be initiated for quality actions that are not in the required state by the due date.</p> <p>To select <b>Escalation Management Translator</b>, ensure that you select <b>Quality Action Management</b> and <b>Dispatcher</b> features</p>
<b>Failure Mode Effect Analysis (FMEA) data model</b>	<p>Adds support for Failure Mode Effect Analysis (FMEA) to Teamcenter Quality.</p> <p>For more information, see <i>Teamcenter Quality</i> in the Active Workspace help.</p>
<b>Quality Issue Management and Problem Solving data model</b>	<p>Adds support for Issue Manager and Problem Solving to Active Workspace.</p> <p>For more information, see <i>Teamcenter Quality</i> in the Active Workspace help.</p>
<b>Quality Issue Management and Problem Solving search</b>	Adds the ability to search for Problem Solving objects in Active Workspace.

Feature	Description
<b>APQP Program Management</b>	<p>Adds support for Advanced Product Quality Planning (APQP) methodology to Teamcenter Quality.</p> <p>For more information, see <i>Teamcenter Quality</i> in the Active Workspace help.</p>
<b>Quality Audit</b>	<p>Enables capturing the status of quality actions in audit logs. A Teamcenter administrator logged into Active Workspace can view these event logs and navigate to specific quality actions.</p>
<b>Model-Based Systems Engineering</b>	<p>Features to support Model-Based Systems Engineering in Active Workspace.</p>
<b>Measurable Attributes and Targets</b>	<p>Configures Active Workspace to interact with <b>Measurable Attributes and Targets</b>, providing functionality such as retrieving the associated measurable attributes from a parent object or parent line.</p> <p>This feature adds the <b>Measurable Attributes and Targets for Active Workspace</b> (<code>att1attrtargetmgmtaw_template.xml</code>) template to the database.</p> <p>This feature is not available with Teamcenter Rapid Start.</p> <p>In addition to the <b>Active Workspace</b> feature, this feature requires:</p> <ul style="list-style-type: none"> <li>• <b>Extensions→Model-Based Systems Engineering→Verification and Validation Planning and Reporting</b> Selecting this feature automatically selects these features in the same feature group: <ul style="list-style-type: none"> <li><b>Measurable Attributes and Targets</b></li> <li><b>Attribute and Parameter Base Definitions</b></li> </ul> </li> <li>• <b>Base Install→Active Workspace→Server Extensions→Systems Engineering</b></li> <li>• <b>Base Install→Active Workspace→Server Extensions→Model-Based Systems Engineering→Verification and Validation Planning and Reporting</b> Selecting this feature automatically selects the <b>Measurable Attributes and Targets</b> feature in the same feature group.</li> </ul> <p>If you are installing this feature, you must also install the <b>Measurable Attributes and Targets Client</b> feature in the Client web application.</p>
<b>Physical Verification Management Active</b>	<p>Adds support for verification management in Active Workspace.</p>

Feature	Description
<b>Workspace Extension</b>	
<b>LMS System Synthesis Modeling</b>	<p>Exposes the System Synthesis artifacts, such as model templates and core architecture, in Active Workspace. These artifacts are created in Teamcenter by System Synthesis, a framework for the numerical integration of heterogeneous behavioral models issued by different authoring platforms, such as LMS Amesim or MATLAB/Simulink.</p> <p>This feature adds the <b>System Synthesis Modeling for Active Workspace</b> (<code>ssm1systemsynthesismodelingaw_template.xml</code>) template to the database.</p> <p>This feature is not selectable unless the <b>System Synthesis Modeling</b> feature (under <b>Extensions</b>→<b>Systems Driven Product Development</b>) is also selected.</p>
<b>Parameter Management Active Workspace</b>	Adds parameter management support to Active Workspace.
<b>Teamcenter Test Management</b>	Adds test management support to Model-Based Systems Engineering (MBSE).
<b>Physical Verification Management</b>	Adds support for inspections and physical tests.
<b>Active Workspace Linked Data Framework Services</b>	Features to support Linked Data Framework Services in Active Workspace.
<b>LDF Foundation</b>	<p>Provides a framework to link Teamcenter business objects with an artifact of a remote linked data enabled system. It enables rendering the remote system's delegated user interfaces in Active Workspace.</p> <p>This feature adds the <b>Linked data Framework for Active Workspace</b> (<code>ldf0ldfaws_template.xml</code>) template to the database.</p> <p>This feature is not selectable unless the <b>Linked Data Framework Support Infrastructure</b> feature (under <b>Extensions</b>→<b>Platform Extensibility</b>→<b>Linked Data Services</b>) is also selected.</p> <p>If you install this feature, you should also install the <b>Linked Data Framework</b> feature (under <b>Client</b>) when building the Client web application.</p>
<b>Active Workspace LDF Change Management Integration</b>	Installs the Change Management integration module of Linked Data Services (LIS). The feature requires corresponding Teamcenter features.

Feature	Description
<b>Active Workspace LDF Requirements Management Integration</b>	<p>Installs the Requirements Management integration module of Linked Data Services (LIS).</p> <p>This feature requires corresponding Teamcenter features.</p>
<b>LDF Polarion Types Integration</b>	<p>Installs Polarion data types for Linked Data Services (LIS) integration in Active Workspace.</p> <p>This feature requires corresponding Teamcenter features.</p>
<b>Active Workspace LDF Embedded Software Management Integration</b>	<p>Installs the Embedded Software Management integration module of Linked Data Services (LIS).</p> <p>For more information, see <i>Integrating Applications Using Linked Data Framework</i> in the Teamcenter help collection.</p>
<b>Consumer Packaged Goods</b>	<p>Features to support CPG in Active Workspace.</p> <p>For more information, see <i>Getting Started with CPG</i> in the Teamcenter help collection.</p>
<b>Brand Management</b>	<p>Installs the Brand Management template for CPG.</p> <p>For more information, see <i>Getting Started with CPG</i> in the Teamcenter help collection.</p>
<b>Packaging and Artwork</b>	<p>Installs packaging and artwork functionality for CPG.</p> <p>For more information, see <i>Getting Started with CPG</i> in the Teamcenter help collection.</p>
<b>Specification Management</b>	<p>Installs specification management functionality for CPG.</p> <p>For more information, see <i>Getting Started with CPG</i> in the Teamcenter help collection.</p>
<b>Consumer Product Management Active Workspace</b>	<p>Installs consumer product management functionality for CPG.</p> <p>For more information, see <i>Getting Started with CPG</i> in the Teamcenter help collection.</p>

## Additional Active Workspace server extensions features

The following Active Workspace server extensions features are available under **Base Install**→**Extensions** in the **Features** panel in TEM.

Feature	Description
<b>Color Appearance Active Workspace</b>	Provides the ability to define standard color attributes in Active Workspace.
<b>Content Management Active Workspace</b>	Adds server-side Content Management support to Active Workspace.
<b>Engineering Views Active Workspace</b>	Adds engineering views to Active Workspace.
<b>Product Master Automation for Active Workspace</b>	Installs product master automation for Active Workspace.

## Active Workspace Client features

Active Workspace client features are available in the **Features** panel in Teamcenter Environment Manager (TEM), under **Base Install**→**Active Workspace**→**Client**.

To search for a feature by name, type the name or a partial name in the search box, and then click the search icon.

Feature	Description
<b>Active Workspace Gateway</b>	Installs the <b>Active Workspace Gateway</b> , a web application framework that resides between the Active Workspace client application and your browser. The Gateway communicates with the Teamcenter server, the File Repository, and the volume server, and routes incoming requests for static and dynamic content to the appropriate services handling those requests.
<b>Active Workspace Client</b>	Selecting this feature builds the Active Workspace client.
<b>Partitions for Structure</b>	Installs client support for partitions and partition schemes for structures.
<b>3D Visualization</b>	Adds 3D visualization to the Active Workspace client.
<b>Active Collaboration Client</b>	Adds workflow capabilities of Active Collaboration to the client interface. If you are installing this feature in the Client, you must install the <b>Active Collaboration</b> Server Extensions feature in the corporate server.
<b>Active Collaboration for Retail Solution Client</b>	Installs <i>Active Collaboration</i> for Retail Footwear and Apparel users.
<b>Active Workspace Assistant</b>	Installs client components of the Assistant, a guided navigation tool for Active Workspace. It predicts the next likely actions a user will perform based on the user's history, group, and role, and presents suggested actions in the <b>Assistant</b> panel.

Feature	Description
	This feature requires the <b>Active Workspace Assistant</b> server extension feature on the Teamcenter server.
<b>Active Workspace Visualization 2D Viewer</b>	Adds the 2D Viewer to the Universal Viewer in Active Workspace. Requires the <b>Visualization Server</b> .
<b>Audit</b>	Adds Audit Manager features to the Active Workspace client. For more information, see <i>Audit Manager</i> in the Teamcenter help collection.
<b>Content Management Active Workspace</b>	Adds Content Management support to Active Workspace.
<b>DPV for Active Workspace Client</b>	Adds Dimensional Planning and Validation functionality to the client interface.
<b>Document Management Client</b>	Installs Document Management support for the Active Workspace client.
<b>Electronic Design Automation</b>	Adds Electronic Design Automation (EDA) support in the Active Workspace client.
<b>Google Online Viewer Client</b>	Installs the Google Online Viewer, which enables Google Online features within Active Workspace, such as viewing and editing Microsoft Office documents without the need for Microsoft Office applications.
<b>Logical Object</b>	Allows users to access logical object data in the client interface.
<b>Multisite Integration</b>	Enables certain Multi-Site Collaboration capabilities in Active Workspace. The <b>AWC_Enable_Multisite</b> preference must be set to <b>true</b> and the <b>TC_force_legacy_multisite</b> preference must be set to <b>false</b> on each site in the Multi-Site Collaboration network.
<b>Office Online Viewer Client</b>	Installs the Office Online Viewer Client that enables you to edit Microsoft Office documents within Active Workspace.  This feature requires Microsoft Office Online to be installed in your environment.
<b>Part Manufacturing</b>	Adds part manufacturing support in the Active Workspace client.
<b>Product Configurator</b>	Adds Product Configurator features to the Active Workspace client. Product Configurator enables you to formally introduce and manage variability across your product suite.  For more information, see <i>Product Configurator</i> in the Teamcenter help collection.
<b>Program Planning</b>	Enables the Program Management capability in Active Workspace. This feature provides the ability to manage business investments, from planning to execution, in terms of time, reuse, volume, cost targets, and weight targets.

Feature	Description
	If you are installing this feature, you should also install the <b>Program Planning</b> feature under <b>Server Extensions</b> .
<b>Reactive Logging</b>	<p>Installs valuable troubleshooting tools that enable you to record a problem in a log file to share with the Teamcenter administrator.</p> <p>By default, reactive logging provides the user with the log file location and machine information. To configure this behavior, set the following preferences:</p> <p><b>TC_reactive_logging_notification_list</b></p> <p>Notifies administrator regarding log ZIP file.</p> <p><b>TC_reactive_logging_file_download</b></p> <p>Displays message to the user with a link to the downloaded file in Active Workspace.</p>
<b>Reporting</b>	<p>Adds the ability to view report templates, generate reports based on selected criteria, style sheets, or both, and view them in HTML, Excel, or raw XML formats in the client.</p> <p>If you are installing this feature in the Client, you must install the <b>Reporting</b> Server Extensions feature in the corporate server.</p> <p>Additionally, to allow for asynchronous report generation, install the <b>AsyncService</b> translator in the Dispatcher Server.</p>
<b>Schedule Manager</b>	<p>Enables Schedule Manager capabilities in Active Workspace.</p> <p>If you are installing this feature, you must also install the <b>Schedule Manager</b> feature under <b>Server Extensions</b> on the corporate server.</p>
<b>Stock Material</b>	<p>Adds stock material management features for the Active Workspace client.</p> <p>Many parts are made from stock materials such as bar stock, tubing stock and sheet stock. This features enables you to manage stock materials in Teamcenter, performing actions like creating libraries of stock materials and assigning stock materials to parts.</p> <p>For more information, see <i>Aerospace and Defense Solution</i> in the Teamcenter help collection.</p>
<b>Subscription</b>	<p>Allows users to manage subscriptions and notifications in the client.</p> <p>If you are installing this feature in the Client, you must install the <b>Subscription</b> feature under <b>Server Extensions</b> in the corporate server.</p>
<b>Work Package Management</b>	<p>Installs work package management, which helps designers to create and maintain a work package as a revisable collection of CAD files and documentation.</p> <p>To install this feature, you must install the <b>Work Package Management</b> feature for both server and client features for Active Workspace.</p>

Feature	Description
<b>Contract Data Management</b>	<p>Installs Contract Data Management, which helps contractors manage the creation, review, and delivery of contracts. A <i>contract</i> is a structured procurement document that lists milestones and schedule dates.</p> <p>If you are installing this feature, you must also install the <b>Contract Data Management</b> Server Extensions feature on the server and the <b>Contract Data Management</b> feature under <b>Enterprise Knowledge Foundation</b>.</p>
<b>Markup</b>	Enables markup capabilities in the client interface.
<b>Relationship Browser</b>	If you are installing this feature, you must also install the <b>Relationship Viewer</b> Server Extensions feature on the corporate server.
<b>Workflow</b>	<p>For users to access <b>Inbox</b> components and workflow functionality in the Client interface, this feature must be installed.</p> <p>If you are installing this feature, you must also install the <b>Workflow</b> feature under <b>Server Extensions</b> on the corporate server.</p>
<b>Active Content</b>	<p>Adds structure search functionality to the client interface.</p> <p>If you are installing this feature in the Client, you must install the <b>Active Content Structure</b> Server Extensions feature in the corporate server.</p>
<b>Digital Signatures</b>	<p>Adds digital signatures functionality to the client interface.</p> <p>Before selecting this feature, see the additional setup requirements for digital signatures in <i>Configuration and Extensibility</i>.</p> <p>If you are installing this feature in the Client, you must install the <b>Digital Signatures Server</b> Server Extensions feature in the corporate server.</p>
<b>Easy Plan - Product Configurator Client</b>	<p>Adds support for Product Configurator in Easy Plan. Product Configurator allows you to enable definitions for variant formulas on operations.</p> <p>Please see the Product Configurator (Administrator) documentation for installation details.</p> <p>To use Product Configurator variants, you must run Teamcenter 11.5 or later. By default, Easy Plan uses classic variants.</p>
<b>Linked Data Framework</b>	<p>Provides a framework to link a Teamcenter business objects with an artifact of a remote linked data enabled system. It enables rendering the remote system's delegated UIs in Active Workspace.</p> <p>If you are installing this feature, you should also install the <b>Linked Data Framework</b> feature under <b>Server Extensions</b>.</p>
<b>Teamcenter Manufacturing Access Client</b>	<p>Adds Teamcenter Manufacturing Access functionality to the Active Workspace client.</p> <p>Teamcenter Manufacturing Access provides solutions for various manufacturing planning tasks and business processes. It is designed as a cross-industry tool; use it in any environments where Process Simulate is used.</p>



Feature	Description
	For more information about Teamcenter Manufacturing Access, see Easy Plan help.
<b>Viewer Snapshot Tool</b>	Installs the viewer snapshot tool for 3D visualization in Active Workspace.
<b>4th Generation Design</b>	<p>Allows users to view, navigate, and configure collaborative designs and their content in the Client.</p> <p>If you are installing this feature in the Client, you must also install the <b>4th Generation Design</b> Server Extensions feature, the <b>Active Content Structure</b> Server Extension feature, and the <b>4th Generation Design</b> (under <b>Advanced PLM Services</b>) feature on the corporate server.</p> <p>In addition to the <b>Active Workspace Client</b> feature, requires the <b>Active Content</b> Client feature.</p>
<b>Architecture Modeler</b>	In addition to the <b>Active Workspace Client</b> feature, requires the <b>Active Content</b> Client feature.
<b>Briefcase Browser</b>	<p>Adds support for Briefcase Browser in the Active Workspace client.</p> <p>For more information, see <i>Briefcase Browser</i> in the Teamcenter help collection.</p>
<b>Briefcase Export and Import</b>	Enables export of an NX assembly from Active Workspace as an unmanaged briefcase export file for stand-alone NX. The exported file honors any revision rule and variant configuration applied prior to exporting.
<b>Change Management</b>	<p>Adds the ability to work with Change Management objects in the client.</p> <p>If you are installing this feature in the Client, you must install the <b>Extensions→Enterprise Knowledge Foundation→Change Management</b> feature in the corporate server.</p>
<b>Concurrent Modeling</b>	Adds concurrent modeling support in Active Workspace. Concurrent modeling allows you to manage models from supported 1D modeling tools in Teamcenter, using the Teamcenter MBSE Integration Gateway MBSE framework.
<b>Easy Plan - Process Planning for BTO/BTS/CTO Client</b>	Installs client support for the Process Planning workspace for build-to-order (BTO) or build-to-stock (BTS), or configure-to-order (CTO) in your environment.
<b>Engineering Views</b>	Adds support for engineering views in Active Workspace.
<b>Finish Management</b>	<p>Installs Finish Management support for Active Workspace. A <i>finish</i> represents a finishing process on a part. It may be used to improve appearance, adhesion, corrosion resistance, tarnish resistance, chemical resistance, wear resistance, and remove burrs and so on.</p> <p>For more information, see <i>Aerospace and Defense Solution</i> in the Teamcenter help collection.</p>

Feature	Description
<b>MCAD Integration</b>	In addition to the <b>Active Workspace Client</b> feature, requires the <b>Active Content</b> Client feature.
<b>Material Management</b>	<p>Enables Active Workspace users to:</p> <ul style="list-style-type: none"> <li>• Associate a material revision with a vendor part or an item revision.</li> <li>• Associate a substance with a material revision.</li> <li>• View the material/substance dashboard.</li> </ul> <p>If you install this feature, you should also install the <b>Material Management</b> feature (under <b>Server Extensions</b>) on the corporate server.</p>
<b>NX Integration</b>	<p>Enables users to access NX integration functionality from the client interface.</p> <p>If you are installing this feature, you should also install the <b>NX for Active Workspace</b> feature (under <b>Server Extensions</b>) on the corporate server.</p>
<b>PLMXML Export Import</b>	<p>Adds support for Active Workspace import and export in Active Workspace.</p> <p>PLM XML is an open standard for exchanging product life cycle information using several supported XML schemas. Data represented in these schemas includes product structure, geometry, visualization, features, application associativity, data ownership, and deltas (changes).</p>
<b>Product Master</b>	Installs product master support for Product Master Manager in the Active Workspace client.
<b>Product Master Automation</b>	Installs product master automation for the Active Workspace client.
<b>Requirements Management</b>	<p>Allows users to author a requirement structure in the <b>Content</b> tab of the client.</p> <p>In addition to the <b>Active Workspace Client</b> feature, requires the <b>Active Content</b> feature.</p> <p>If you are installing this feature in the Client, you must install the <b>Requirements Management</b> and <b>Systems Engineering</b> features under <b>Server Extensions</b> on the corporate server.</p> <div style="border: 1px solid black; padding: 10px; margin-top: 10px;"> <p>Note:</p> <p>This feature is required to enable the Export to Excel button in Active Workspace.</p> </div>
<b>Resource Manager for Active Workspace Client</b>	<p>Adds Manufacturing Resource Library support for the Active Workspace client.</p> <p>For information about using Manufacturing Resource Library, see the Teamcenter help library.</p>

Feature	Description
<b>Service Engineering Client</b>	Provides service engineering support for Service Lifecycle Management in the Active Workspace client.
<b>Service Manager</b>	Allows users to view disposition and utilization history of physical parts in the Client.
<b>Vendor Management</b>	<p>Allows Active Workspace users to:</p> <ul style="list-style-type: none"> <li>• Associate vendors with vendor parts.</li> <li>• Associate vendor parts with commercial part revisions.</li> </ul> <p>If you are installing this feature, you should also install the <b>Vendor Management</b> feature under (<b>Server Extensions</b>) on the corporate server.</p>
<b>Aerospace and Defense Foundation</b>	<p>Adds the Aerospace and Defense Foundation feature to the Active Workspace client.</p> <p>This feature is not selectable unless the following features are also selected:</p> <p>If you are installing this feature in the Client, you must install the <b>Aerospace and Defense Foundation</b> Server Extension feature in the corporate server and the following Active Workspace Client features:</p> <ul style="list-style-type: none"> <li>• <b>Workflow</b></li> <li>• <b>Active Collaboration</b></li> <li>• <b>Active Content</b></li> <li>• <b>Vendor Management</b></li> </ul>
<b>Capital Asset Lifecycle Management</b>	Adds management of plant data for the Active Workspace client.
<b>Easy Plan - Electronics Client</b>	<p>Installs client support for the Easy Plan Electronics Process Planner workspace in your environment.</p> <p>Electronics engineers use Production Process Planning for the systems integration of the electronics process planning tasks of Easy Plan and those of Valor Process Preparation Software. The systems integration is available as a dedicated, optional feature of Production Process Planning.</p>
<b>Easy Plan - Process Planning for ETO Client</b>	<p>Installs client support for the Engineering to Order (ETO) workspace in your environment.</p> <p>This feature allows end users to author work instructions at the process station level.</p>
<b>Embedded Software Management</b>	Adds Embedded Software Solutions functionality, which allows you to represent embedded software artifacts using Active Workspace.

Feature	Description
<b>Manufacturing BOM Manager Client</b>	<p>Adds Multi-BOM Manager support to the Active Workspace client. Multi-BOM Manager enables you to link and assign content across representations for different lifecycle stages or uses for a product.</p> <p>For information about using this feature, see <i>Mutli-BOM Manager</i> in the Teamcenter help library.</p>
<b>Next Generation Planning Client</b>	Adds Next Generation Planning (NGP) support to Active Workspace.
<b>Electronic Work Instructions Client</b>	Installs the Electronic Work Instructions Client workspace in your environment.
<b>System Modeler</b>	<p>Install this feature so that users can access functionality in the <b>Architecture</b> toolbar of the Client interface.</p> <p>If you are installing this feature in the Client, you must install the <b>Systems Modeling</b> Server Extension feature in the corporate server and the following Active Workspace Client features:</p> <ul style="list-style-type: none"> <li>• <b>Workflow</b></li> <li>• <b>Active Content</b></li> <li>• <b>Architecture Modeler</b></li> </ul>
<b>Program Work Breakdown Structure Active Workspace</b>	Adds support for authoring Work Breakdown Structure (WBS) hierarchy in Active Workspace from top to bottom. For information about defining work breakdowns, see <i>Schedule Manager</i> in the Teamcenter help library.
<b>Active Admin</b>	Features to support active admin capabilities.
<b>Preference Management</b>	Adds preference management to the active admin workspace.
<b>CAE Simulation Management</b>	Features to support management of computer-aided engineering (CAE) data.
<b>Simulation Process Management</b>	<p>In addition to the <b>Active Workspace Client</b> feature, requires the <b>Active Content</b> Client feature.</p> <p>If you are installing this feature, you must also install the <b>Simulation Process Management</b> Server Extensions feature on the corporate server.</p>
<b>Extended Simulation Process Management</b>	Extends Simulation Process Management capabilities in Active Workspace.

Feature	Description
<b>Integrated Program Planning and Execution</b>	Integrated Program Planning and Execution (IPP&E) client features. The IPP&E solution allows project planning that integrates cost, schedule, risk and technical requirements in a fully planned, resourced, and budgeted program. It allows configuration control, not only of products, but also of the project plan. It also communicates the status of requirements to users.
<b>IPP&amp;E Contract Data Management Extension</b>	Adds Contract Data Management support to IPP&E.
<b>Organization Breakdown Structure</b>	Adds Organization Breakdown Structure (OBS) support to IPP&E.
<b>IPP&amp;E Foundation</b>	Provides essential functionality for Integrated Program Planning and Execution in Active Workspace.
<b>Work Breakdown Structure</b>	Adds Work Breakdown Structure (WBS) support to IPP&E.
<b>Program Planning Execution Client</b>	Program Planning Execution features for the Active Workspace client.
<b>Change Management Schedule Manager Client</b>	<p>Allows interaction between Schedule Manager and Change Management in Active Workspace. It allows Active Workspace users to relate schedules and change objects.</p> <p>If you are installing this feature in the Client, you must install the <b>Change Management Schedule Manager</b> Server Extension feature in the corporate server and the following Active Workspace Client features:</p> <ul style="list-style-type: none"> <li>• <b>Schedule Manager</b></li> <li>• <b>Workflow</b></li> <li>• <b>Change Management</b></li> </ul>
<b>Program Change Client</b>	<p>Allows interaction between Program Planning Event Change and Change Management in Active Workspace. It allows Active Workspace users to relate programs, projects, and subprojects to change objects.</p> <p>If you are installing this feature in the Client, you must install the <b>Program Change</b> Server Extension feature in the corporate server and select the following Active Workspace Client features:</p> <ul style="list-style-type: none"> <li>• <b>Program Planning</b></li> <li>• <b>Workflow</b></li> </ul>

Feature	Description
<b>Program Planning Event Change Client</b>	<ul style="list-style-type: none"> <li>• <b>Change Management</b></li> </ul> <p>If you are installing this feature in the Client, you must install the <b>Program Planning Event Change</b> Server Extension feature in the corporate server and select the following Active Workspace Client features:</p> <ul style="list-style-type: none"> <li>• <b>Program Planning</b></li> <li>• <b>Workflow</b></li> <li>• <b>Change Management</b></li> <li>• <b>Program Change Client</b></li> </ul>
<b>Program Schedule Manager Client</b>	<p>If you are installing this feature in the Client, you must install the <b>Program Change</b> Server Extension feature in the corporate server and select the following Active Workspace Client features:</p> <ul style="list-style-type: none"> <li>• <b>Program Planning</b></li> <li>• <b>Schedule Manager</b></li> <li>• <b>Workflow</b></li> <li>• <b>Change Management</b></li> <li>• <b>Program Change Client</b></li> <li>• <b>Program Planning Event Change Client</b></li> </ul>
<b>Reuse and Standardization</b>	Features to support Reuse and Standardization in the Active Workspace client.
<b>Classification Client</b>	Allows users to access Classification data in the client interface.
<b>Library Management</b>	Installs the client component required to view classification libraries. Classification libraries contain subsets of a classification hierarchy pertinent to a particular role, project, or use case.
<b>Teamcenter Quality</b>	Features to support Teamcenter Quality.
<b>Control and Inspection Plan</b>	<p>Installs support for control and inspection planning in the Active Workspace client.</p> <p>Control and inspection planning allows you to manage critical characteristics of Failure Mode Effect Analysis (FMEA) and create a control plan that generates bill of process (BOP) elements.</p>

Feature	Description
<b>Failure Mode Effect and Analysis (FMEA)</b>	<p>Adds support for Failure Mode Effect Analysis (FMEA) standards to the Active Workspace client.</p> <p>For more information, see <i>Quality</i> in the Active Workspace help.</p>
<b>Quality Action Management</b>	<p>Adds support for quality actions in the Active Workspace client.</p> <p>For more information, see <i>Quality</i> in the Active Workspace help.</p>
<b>Quality Audit</b>	<p>Adds support for quality auditing in the Active Workspace client.</p> <p>For more information, see <i>Quality</i> in the Active Workspace help.</p>
<b>Quality Manager</b>	<p>Adds Quality Manager to the Active Workspace client.</p> <p>For more information, see <i>Quality</i> in the Active Workspace help.</p>
<b>Training and Qualification</b>	Adds client support for training and qualification actions in Active Workspace.
<b>Quality Issue Management and Problem Solving</b>	Adds Issue Manager and Problem Solving support for the Active Workspace client.
<b>APQP Program Management Client</b>	<p>Adds support for Advanced Product Quality Planning (APQP) methodology in the Active Workspace client.</p> <p>For more information, see <i>APQP Program Management</i>.</p>
<b>Model-Based Systems Engineering</b>	Features to support Model-Based Systems Engineering in the Active Workspace client.
<b>LMS System Synthesis Modeling</b>	<p>Exposes the System Synthesis artifacts, such as model templates and core architecture, in Active Workspace. These artifacts are created in Teamcenter by System Synthesis, a framework for the numerical integration of heterogeneous behavioral models issued by different authoring platforms, such as LMS Amesim or MATLAB/Simulink.</p> <p>If you are installing this feature in the Client, you must install the <b>LMS System Synthesis Modeling</b> Server Extension feature in the corporate server and the following Active Workspace Client features:</p> <ul style="list-style-type: none"> <li>• <b>Workflow</b></li> <li>• <b>Active Content</b></li> </ul>
<b>Parameter Management</b>	Adds parameter management to Active Workspace.
<b>Verification and Validation</b>	Allows system engineers and validation engineers to validate key performance measurements of objects.

Feature	Description
<b>Planning and Reporting</b>	<p>If you are installing this feature in the Client, you must install the <b>Verification and Validation Planning and Reporting</b> Server Extension feature in the corporate server and the following Active Workspace Client features:</p> <ul style="list-style-type: none"> <li>• <b>Workflow</b></li> <li>• <b>Active Content</b></li> <li>• <b>Architecture Modeler</b></li> </ul>
<b>Teamcenter Test Management</b>	Adds test management support to Model-Based Systems Engineering (MBSE).
<b>Physical Verification Management</b>	Adds support for inspections and physical tests in the Active Workspace client.
<b>Active Architect</b>	Features to support Active Architect for Active Workspace.
<b>UI Builder</b>	<p>Installs UI Builder components of active architect. UI Builder adds new declarative pages to the global navigation toolbar in Active Workspace, including:</p> <ul style="list-style-type: none"> <li>• Command builder</li> <li>• Panel builder</li> </ul>
<b>Consumer Packaged Goods</b>	Features to support Consumer Packaged Goods in the Active Workspace client.
<b>Brand Management</b>	Installs the Brand Management template for Consumer Packaged Goods.
<b>Packaging and Artwork</b>	Installs packaging and artwork functionality for Consumer Packaged Goods.
<b>Specification Management</b>	Installs Specification Manager functionality for Consumer Packaged Goods.
<b>Consumer Product Management Active Workspace</b>	Installs consumer product management functionality for Consumer Packaged Goods.

## Active Workspace microservices features

Active Workspace client features are available in the **Features** panel in Teamcenter Environment Manager (TEM), under **Microservices**.

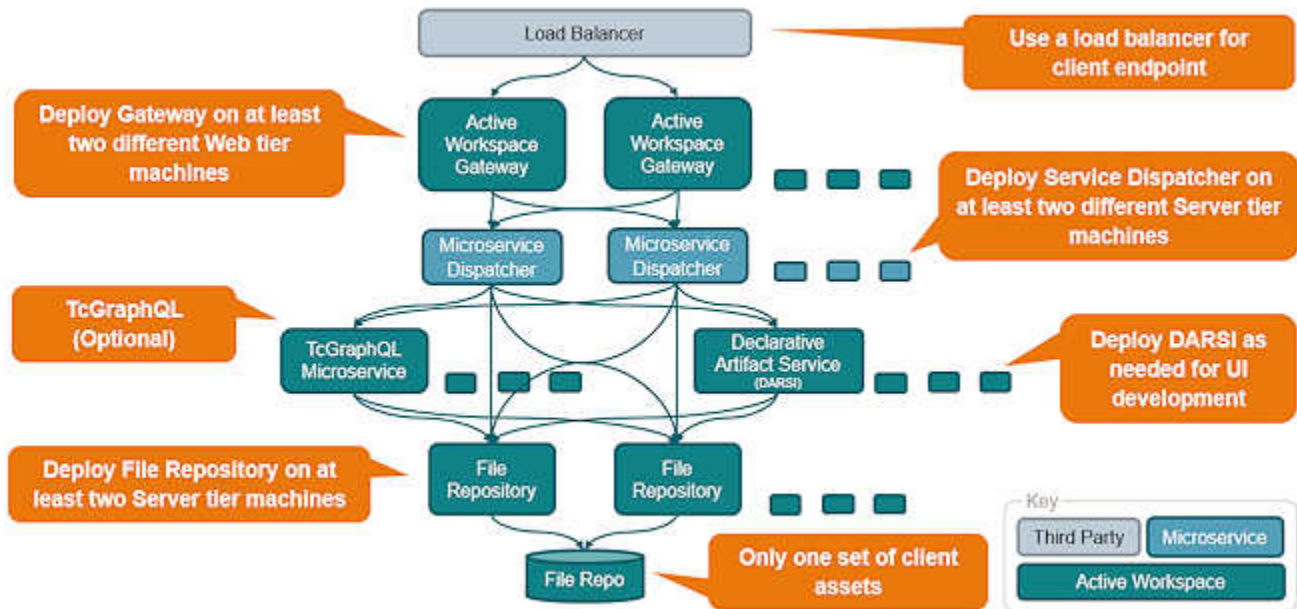


These features require **installation of Microservice Framework**.

Feature	Description
<b>Microservices Framework</b>	Installs the <b>Microservices Framework</b> .
<b>Classification AI Serving</b>	Installs the classification AI microservice that connects the Active Workspace Gateway to the classification AI engine, which provides artificial intelligence (AI) to recommend the class in which to classify new objects.
<b>Command Prediction Service</b>	Installs the Command Prediction Service, which manages data for the <b>Active Workspace Assistant</b> and a database for the microservice to store data for the Active Workspace Assistant. TEM prompts you for the necessary database creation values during installation.
<b>Declarative Artifact Service</b>	Installs the microservice that stores your changes to the declarative definitions of Active Workspace.  The Declarative Artifact Service allows you to easily create new declarative definitions by automatically creating new files, and also make changes to existing declarative definitions without changing their original source code.
<b>File Repository Service</b>	Installs the File Repository microservice, which facilitates transfer of Active Workspace client content from the File Repository to Active Workspace client hosts through the Active Workspace Gateway.
<b>Requirements Management Compare Service</b>	Installs the microservice that provides compare capability for Requirements Manager.
<b>Requirements Management Export Service</b>	Installs the microservice that provides export capability for Requirements Manager.
<b>Requirements Management Import Service</b>	Installs the microservice that provides import capability for Requirements Manager.
<b>Teamcenter GraphQL Service</b>	Installs the Teamcenter GraphQL microservice, which processes client queries passed from the Active Workspace Gateway to the File Repository.
<b>iModel Viewer Service</b>	Provides viewing capability for iModel data.
<b>Teamcenter Google Online Microservice</b>	Provides interaction between the Google Online Viewer and the Teamcenter server.

Feature	Description
<b>Teamcenter Office Online Microservice</b>	Provides the capability to use the Microsoft Office Online Server features that allows users to edit and view documents within Active Workspace instead of using the desktop version of the Microsoft Office applications.
<b>OData Microservice</b>	Installs the microservice that supports the Teamcenter OData API Framework.

## B. Load balancing Active Workspace components



Load balancing Active Workspace components provides failover support. This allows the system to continue to operate when hardware or connectivity failures occur, and it allows for maintenance of hardware without production downtime.

You should deploy the Active Workspace Gateway on at least two different physical machines and use a commercial third-party load balancer configuration. The load balancer is used to enable the browser clients to use one and only one URL to access Active Workspace and Teamcenter.

During your installation process, you should configure the Gateway to address at least two microservice dispatchers on two different physical machines. If you chose to deploy the Teamcenter GraphQL service<sup>1</sup>, you may be less concerned about failover and install on only one machine.

The Declarative Artifact Service is only used for user interface development using the UI Builder. Therefore, you may choose to deploy it only as needed in development environments. The File Repository service is an essential service and should be deployed on at least two different physical machines to support failover. The actual file volume for the client assets must be on one and only one machine.

In all cases, microservices can be deployed on more machines to achieve greater scalability. Any service that will have heavy usage may warrant having more than two instances.

<sup>1</sup> The Teamcenter GraphQL service is optional.

If you install the File Repository Service microservice on multiple machines, the file repository it has to be on a shared drive.

If you install multiple instances of the File Repository Service microservice, all instances must reference the same physical storage location. For example, if you install the File Repository Service microservice on multiple machines, the file repository storage location must be on a shared drive.

## C. Required RPM package managers

If you use the visualization server manager (VSM) on a Linux machine, make sure the following required RPM package managers are available on the machine.

### SUSE Linux:

```
fontconfig-2.11.1-7.1.x86_64
glibc-2.22-15.3.x86_64
glibc-32bit-2.22-15.3.x86_64
libbz2-1-1.0.6-29.2.x86_64
libexpat1-2.1.0-21.3.1.x86_64
libexpat-devel-2.1.0-21.3.1.x86_64
libfreetype6-2.6.3-7.15.1.x86_64
libgcc_s1-8.2.1+r264010-1.3.3.x86_64
libGLU1-9.0.0-18.1.x86_64
libICE6-1.0.8-12.1.x86_64
libjpeg8-8.1.2-31.7.4.x86_64
libpng16-16-1.6.8-14.1.x86_64
libSM6-1.2.2-3.59.x86_64
libstdc++6-8.2.1+r264010-1.3.3.x86_64
libuuid1-2.29.2-7.14.x86_64
libX11-6-1.6.2-12.5.1.x86_64
libXau6-1.0.8-4.58.x86_64
libxcb1-1.10-4.3.1.x86_64
libXext6-1.3.2-4.3.1.x86_64
libXft2-2.3.1-9.32.x86_64
libXm4-2.3.4-4.15.x86_64
libXmu6-1.1.2-3.60.x86_64
libXp6-1.0.2-3.58.x86_64
libXrender1-0.9.8-7.1.x86_64
libXt6-1.1.4-3.59.x86_64
libz1-1.2.11-1.27.x86_64
Mesa-libGL1-18.0.2-6.28.x86_64
```

#### Note:

On SUSE Linux, the `/usr/lib64/libGLdispatch.so.0` file is not owned by any package.

Also, the `/usr/lib64/libGLX.so.0` file is not owned by any package.

### RedHat Linux:

```
bzip2-libs-1.0.6-13.el7.x86_64
expat-2.1.0-10.el7_3.x86_64
expat-devel-2.1.0-10.el7_3.x86_64
fontconfig-2.13.0-4.3.el7.x86_64
```

freetype-2.8-12.el7.x86\_64  
glibc-2.17-260.el7.x86\_64  
libgcc-4.8.5-36.el7.x86\_64  
libglvnd-1.0.1-0.8.git5baa1e5.el7.x86\_64  
libglvnd-glx-1.0.1-0.8.git5baa1e5.el7.x86\_64  
libICE-1.0.9-9.el7.x86\_64  
libjpeg-turbo-1.2.90-6.el7.x86\_64  
libpng-1.5.13-7.el7\_2.x86\_64  
libSM-1.2.2-2.el7.x86\_64  
libstdc++-4.8.5-36.el7.x86\_64  
libuuid-2.23.2-59.el7.x86\_64  
libX11-1.6.5-2.el7.x86\_64  
libXau-1.0.8-2.1.el7.x86\_64  
libxcb-1.13-1.el7.x86\_64  
libXext-1.3.3-3.el7.x86\_64  
libXft-2.3.2-2.el7.x86\_64  
libXmu-1.1.2-2.el7.x86\_64  
libXp-1.0.2-2.1.el7.x86\_64  
libXrender-0.9.10-1.el7.x86\_64  
libXt-1.1.5-3.el7.x86\_64  
mesa-libGLU-9.0.0-4.el7.x86\_64  
motif-2.3.4-14.el7\_5.x86\_64  
zlib-1.2.7-18.el7.x86\_64



# Siemens Digital Industries Software

## Headquarters

Granite Park One  
5800 Granite Parkway  
Suite 600  
Plano, TX 75024  
USA  
+1 972 987 3000

## Americas

Granite Park One  
5800 Granite Parkway  
Suite 600  
Plano, TX 75024  
USA  
+1 314 264 8499

## Europe

Stephenson House  
Sir William Siemens Square  
Frimley, Camberley  
Surrey, GU16 8QD  
+44 (0) 1276 413200

## Asia-Pacific

Suites 4301-4302, 43/F  
AIA Kowloon Tower, Landmark East  
100 How Ming Street  
Kwun Tong, Kowloon  
Hong Kong  
+852 2230 3308

## About Siemens Digital Industries Software

Siemens Digital Industries Software is a leading global provider of product life cycle management (PLM) software and services with 7 million licensed seats and 71,000 customers worldwide. Headquartered in Plano, Texas, Siemens Digital Industries Software works collaboratively with companies to deliver open solutions that help them turn more ideas into successful products. For more information on Siemens Digital Industries Software products and services, visit [www.siemens.com/plm](http://www.siemens.com/plm).

This software and related documentation are proprietary and confidential to Siemens.

© 2021 Siemens. A list of relevant **Siemens trademarks** is available. Other trademarks belong to their respective owners.