Deployment Center 3.0
# Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction to Deployment Center</td>
<td>1-1</td>
</tr>
<tr>
<td>Getting started with Deployment Center</td>
<td>1-1</td>
</tr>
<tr>
<td>Deployment considerations</td>
<td>1-2</td>
</tr>
<tr>
<td>Deployment process</td>
<td>1-3</td>
</tr>
<tr>
<td>Deployment Center architecture</td>
<td>1-5</td>
</tr>
<tr>
<td>Installing and using Deployment Center</td>
<td>2-1</td>
</tr>
<tr>
<td>System requirements</td>
<td>2-1</td>
</tr>
<tr>
<td>Install Deployment Center</td>
<td>2-1</td>
</tr>
<tr>
<td>Upgrade Deployment Center</td>
<td>2-5</td>
</tr>
<tr>
<td>Participate in the Product Excellence Program</td>
<td>2-6</td>
</tr>
<tr>
<td>Uninstalling Deployment Center</td>
<td>2-7</td>
</tr>
<tr>
<td>Troubleshoot Deployment Center installation or upgrade</td>
<td>2-8</td>
</tr>
<tr>
<td>Start Deployment Center</td>
<td>2-8</td>
</tr>
<tr>
<td>Log on to Deployment Center</td>
<td>2-9</td>
</tr>
<tr>
<td>Troubleshoot Deployment Center operations</td>
<td>2-10</td>
</tr>
<tr>
<td>Backup and recovery procedures</td>
<td>2-10</td>
</tr>
<tr>
<td>Managing Teamcenter environments</td>
<td>3-1</td>
</tr>
<tr>
<td>Manage the repository</td>
<td>3-1</td>
</tr>
<tr>
<td>Troubleshoot the repository service</td>
<td>3-3</td>
</tr>
<tr>
<td>Manage environments</td>
<td>3-4</td>
</tr>
<tr>
<td>Verify software, applications, and components</td>
<td>3-4</td>
</tr>
<tr>
<td>Edit environment properties</td>
<td>3-5</td>
</tr>
<tr>
<td>Create environments</td>
<td>3-6</td>
</tr>
<tr>
<td>Register environments</td>
<td>3-6</td>
</tr>
<tr>
<td>Remove environments</td>
<td>3-7</td>
</tr>
<tr>
<td>Troubleshoot registering environments</td>
<td>3-8</td>
</tr>
<tr>
<td>View registered machines</td>
<td>3-8</td>
</tr>
<tr>
<td>Deploying software using Deployment Center</td>
<td>4-1</td>
</tr>
<tr>
<td>Installation, maintenance, and upgrade strategy</td>
<td>4-1</td>
</tr>
<tr>
<td>Deployment procedure</td>
<td>4-2</td>
</tr>
<tr>
<td>Maintain your environment</td>
<td>4-3</td>
</tr>
<tr>
<td>Upgrade or patch your environment</td>
<td>4-5</td>
</tr>
<tr>
<td>Deploy in a multiple operating system environment</td>
<td>4-6</td>
</tr>
<tr>
<td>Software task</td>
<td>4-7</td>
</tr>
<tr>
<td>Options task</td>
<td>4-7</td>
</tr>
<tr>
<td>Applications task</td>
<td>4-8</td>
</tr>
<tr>
<td>Components task</td>
<td>4-9</td>
</tr>
<tr>
<td>Deploy task</td>
<td>4-12</td>
</tr>
</tbody>
</table>
Run the deployment scripts ......................................................... 4-14
Troubleshoot the deployment script .............................................. 4-16

How to deploy the Business Modeler IDE templates on Teamcenter ........ 5-1
Deploy Business Modeler IDE packages ........................................ 5-1

Microservice Framework for Access Controls and Requirements Management .... 6-1
Running Access Controls and Requirements Management on the Microservice Framework ........ 6-1
Install the Microservice Framework .............................................. 6-1
Linux uses Docker for microservice nodes .................................... 6-7
Configure the Microservice Framework for HTTPS .......................... 6-9
Chapter 1: Introduction to Deployment Center

Getting started with Deployment Center

Deployment Center is a centralized web application for deploying software to Teamcenter environments. Deployment Center gives you access to multiple environments from a single, centralized location. It simplifies the process of installing and updating software and automates deployment. You can manage a variety of Teamcenter environments more efficiently.

Benefits

• Deploy, manage, and maintain multiple versions of software in multiple Teamcenter environments from a single web application with authenticated user access.

• Reduce the costs of managing Teamcenter environments by reducing the number of people, steps, and time involved in software deployment.

• Provide a centralized location to create and view Teamcenter environments and view deployment information.

• Ensure consistency and accuracy by reviewing and adjusting deployment configuration before rollout to Teamcenter environments.
• When setting up your environment, take advantage of entering parameters for one component and having those values shared with related components. For example, when you enter a machine name or port number for a component, that information is shared with other components that require the same information.

**Flexible administration of software deployment**

The software deployment process is accomplished in two phases.

1. Select the software and configure the deployment in the Deployment Center web application. You can manage responsibility for software deployment using a single administrator or share it among the site's deployment experts. For example:
   • A business analyst could make decisions about choosing the applications. Business analysts can choose the software versions and the applications needed by business units.
   • An administrator could make decisions about the machines, ports, URLs, user names, passwords, and component settings for the target machines in an environment. Administrators can make deployment selections based on knowledge of hardware infrastructure, number of users, and which applications are used by business units.

2. After reviewing the deployment configuration, generate the deployment scripts and deploy the software on the target machines.

**Manage software deployment in Teamcenter environments**

You can deploy a variety of software among your Teamcenter environments. A typical deployment process specifies software configuration and generates deployment scripts that install or update the software on target servers using software kits from a common repository.

**Documentation coverage**

This documentation describes how to create, deploy, and manage your Teamcenter environments using Deployment Center. To learn how to deploy your specific software or application, refer to the documentation for your version that you plan to install or update.

**Deployment considerations**

Information about enhancements and new features is in *What's new in Deployment Center 3.0.*

https://www.plm.automation.siemens.com/locale/docs/

**Find software support on GTAC**

For information about specific supported Teamcenter Foundation, Active Workspace, and other software, check compatibility in the *Internal Interoperability – Teamcenter Compatibility Matrix,* available from the GTAC hardware and software certifications page:

http://www.siemens.com/gtac/

**Deploying custom software with Deployment Center**

You can deploy Business Modeler IDE template packages created in Teamcenter 11.3 and later.
Limitations

You can upgrade directly to Deployment Center 3.0 only from these previous Deployment Center releases:

- 2.0, 2.1, 2.1.1, and 2.2

If your current Deployment Center version is not supported for direct upgrade to version 3.0, you must upgrade to a supported initial version, and then upgrade to Deployment Center 3.0. For example, if you are using Deployment Center 1.0, first upgrade to version 1.2, then to version 2.0, and finally to version 3.0.

For information on support of software specific to this release, consult the DeploymentCenter_3.0_README.pdf, available from the Siemens PLM Software GTAC download site where you obtained the Deployment Center 3.0 product download:

http://www.siemens.com/gtac/

To perform additional deployment actions that are not yet available in Deployment Center, run Teamcenter Environment Manager (TEM) on the target servers to complete your environment updates and configuration.

Deployment process

You can simplify your deployment process using Deployment Center. You can assess your registered Teamcenter environments to plan installations or updates. Your deployment choices are automatically saved in Deployment Center as you go, which allows time to confirm your settings before committing to a deployment. You can review the choices and parameter settings before generating the deployment scripts. You can also validate the deployment scripts before deploying.

Learn about the basic principles of the Deployment Center deployment process:

Start Deployment Center

Start and log on to Deployment Center. You can display information about all existing, registered Teamcenter environments. For each environment, you can view deployed software, applications, server components and their parameters, and environment properties.

Stage the software

Plan where to stage the Teamcenter and related software in a centralized repository location. Download your software installation kits and choose where to stage them for deployment:

Choose the best software staging option for your needs:

- Keep the software in the repository on the Deployment Center web server. Map or mount a drive to that location on each target server. You will specify that path in the -softwareLocation deployment script argument when you deploy. Use this option if you don't have spare server space located near the target servers.

- Copy the software to a shared location that is convenient to the target servers. Map or mount a drive to that location on each target server. You will specify that path in the -softwareLocation argument when you deploy. Plan for enough disk space to hold the entire set of deployment software for an environment. Use this option to have the software in one location that is easier to
access during installation. This option is best for environments with servers on multiple operating systems.

- Copy all of the software for a deployment to each target machine, ensuring each target server gets the complete set of software. Specify the path in the `-softwareLocation` argument when you deploy. Plan for additional disk space on each target server. Use this option to install software more quickly.

**Review repository software**

After you unzip and place the software kits in the repository, Deployment Center finds the software and adds it to the **Software Repositories** page in Deployment Center. The **Software Repositories** page lists all software kits available for deployment. Determine whether you have software dependencies, and be sure you have all of the necessary software kits downloaded for your environment.

**Register environments**

You can create and manage registered environments in Deployment Center. Registered Teamcenter environments are listed on the **Environments** page.

- **Create a new environment** if you are beginning with a new installation of Teamcenter. In a new environment, you can view basic properties before deployment.

- **Register your existing Teamcenter environments** if you are planning to update an existing Teamcenter system. The registration process provides information about existing software, applications, and components to Deployment Center. The information is displayed on the **Environments** page after registration. You can evaluate each environment to plan your deployment selections.

  Register your existing environment by running the `send_configuration_to_dc` script locally on your Teamcenter environment's corporate server. The script sends the environment information to Deployment Center.

**Review registered servers**

You can view all machines from your registered Teamcenter environments. On the **Machines** page, you can see all server machines used by the components in your registered environments.

**Choose software and applications**

In Deployment Center, choose the software and applications you want to install or update in the environment.

You can deploy software in a multiple server environment or on a single machine. If your components are on multiple servers, verify the server for each component.

**Configure server components**

In Deployment Center, you configure server components that are associated with your selected software and applications. Some components may be automatically selected for you. You can choose additional servers for deployment. Enter or confirm server configuration parameter values.
Because selections and settings are saved as you go, you can take your time to review and verify them. If you are unsure of your setup or you need to make changes, return later to complete or update your settings before deployment.

When you enter user names and passwords for server components in Deployment Center, passwords are encrypted using AES128 bit encryption in the generated deployment scripts.

**Validate the deployment scripts**

After you verify your configuration information, you then generate the deployment scripts and copy each script to its corresponding target machine. Each script contains the designated target machine name, what to install or update on the target machine, and the software configurations you specify in Deployment Center.

You can deploy the software when you finish configuration, but you may want to validate the script in a test run. As a best practice, run the deployment scripts in diagnostic mode on each of the target machines to validate the configuration entered in Deployment Center. Check the log output for validation errors. No updates are made to the target machine during diagnostic validation. Make the corrections in Deployment Center, regenerate the deployment scripts, and run them in diagnostic mode until all errors are addressed.

**Deploy the software**

After validation, run the deployment scripts on each of the target machines. Be sure the software is available from the shared repository and that the target machines can access it.

**Update or maintain your environments**

You can update or maintain your Teamcenter environments:

- Install or update additional applications and components in Teamcenter environments managed from Deployment Center.

- Perform maintenance on component configuration or parameter values after installation or update in the environments managed from Deployment Center.

To perform additional deployment actions that are not yet available in Deployment Center, run Teamcenter Environment Manager (TEM) on the target servers to complete your environment updates and configuration.

**Deployment Center architecture**

The basic architecture of Deployment Center is comprised of several main parts that communicate with each other.

Jetty web server and the Deployment Center web application

A Jetty web server is automatically installed and configured for Deployment Center, and the installation automatically deploys and runs the Deployment Center web application. No additional installation or configuration is required for the Jetty web server or the Deployment Center web application. Access the web application from a web browser on any machine.
The H2 database is also automatically installed and configured for Deployment Center. The database stores information about the Teamcenter environments registered with Deployment Center. No additional installation or configuration is required for the H2 database.

The repository stores the downloaded software kits. Deployment Center uses the repository subdirectories when it registers Teamcenter environments and displays choices for installing and updating software and applications.

You provide the repository directory location during installation of Deployment Center. Be sure that repository location has adequate disk space available to store all the software kits needed for your deployments.

The repository subdirectories are automatically created:

- **dc_contributions**
  
  Contains the Deployment Center files for versions of Teamcenter, Active Workspace, Business Modeler IDE, and other supported software.

  **Caution**
  
  Do not make changes to this directory.

- **deploy_scripts**
  
  Contains a directory for each environment managed by Deployment Center. Generated deployment scripts and related files are placed in the appropriate environment subdirectory in a timestamped folder. The Deployment Center populates this directory structure for installing software.

  **Caution**
  
  Do not make changes to this directory.

- **software**
  
  Contains the software you want to install or update using Deployment Center. Download Teamcenter software from the GTAC software download site, unzip the archives, and then copy the unzipped folders into the `software` directory.

  Deployment Center automatically scans this directory and displays the list of selections on the **Software Repositories** page.

- **system**
  
  Stores Deployment Center software files.

  **Caution**
  
  Do not make changes to this directory.
The repository service is automatically installed when you install Deployment Center. The repository service runs automatically and monitors the repository. The repository service reports the software kits to Deployment Center and populates the list of available software selections.

**Staging area**

The staging directory is where Deployment Center stores the generated deployment script ZIP files that are created when you finish the deployment tasks. Copy the deployment scripts to each of the target machines. If you have configured components that run on multiple machines in the environment, there is one generated deployment script ZIP file for each machine.
Chapter 2: Installing and using Deployment Center

System requirements

<table>
<thead>
<tr>
<th>Operating systems</th>
<th>Free disk space</th>
<th>Free RAM</th>
<th>Third-party software</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows 7, Windows Server 2008 R2, Windows Server 2012, Windows Server 2016</td>
<td>300 MB Additional disk space is required to store software download kits</td>
<td>2 GB minimum</td>
<td>JRE 1.8 or later OpenJDK 11 or later</td>
</tr>
</tbody>
</table>

All systems must be 64-bit

System, hardware, and software support for your Teamcenter environments is available from GTAC. You must have a WebKey account to access GTAC.

- Certifications are available from the hardware and software certifications page:
  http://www.siemens.com/gtac/
- The Internal Interoperability – Teamcenter Compatibility Matrix is also available from the hardware and software certifications page.

Install Deployment Center

Prepare to install Deployment Center

Before you begin the Deployment Center installation, check:

- The server meets system requirements.
- The server directory path for Deployment Center has write access.

  Note
  Be sure the path does not contain spaces or quotation marks.

- The port you want to use for the Deployment Center web server is not in use.
- The repository directory for Deployment Center has sufficient disk space for software storage.
- The port that you want to use for the repository to communicate with the Deployment Center server is not in use.
- The installing user account has administrative privileges so that the Deployment Center service is installed correctly.
• The Windows service name you want to use does not conflict with existing service names.

**Install Deployment Center**

1. Download the installation ZIP file for Deployment Center from the GTAC software [Download](#) page.

2. Install the supported JRE version of Java and set JAVA_HOME to the location.

3. Extract the Deployment Center installation ZIP file to the server where you want to run Deployment Center. The server must be accessible to the Teamcenter environments where you want to install and manage software.

4. Open a command prompt window and navigate to the location where you extracted the Deployment Center installation software. Go to the deployment_center directory.

5. Run `deployment_center.bat` using the following arguments. Required arguments are noted; running Deployment Center as a service is optional.

   - **-install** (required)
     
     Run in install mode.

   - **-serverDir** (required)
     
     Specify the full path to the directory for the Deployment Center web server and the database. The path can't contain spaces.

   - **-serverPort** (required)
     
     Specify the port where the Deployment Center web server listens for requests.

   - **-repoDir** (required)
     
     Specify the full path to the directory for the Deployment Center repository. The path presumes that the required `software` subdirectory exists.

     You can choose to map a drive to an existing software repository.

   - **-repoPort** (required)
     
     Specify the port where the Deployment Center repository listens for requests from the Deployment Center server.

   - **-user** (required)
     
     Specify the user name for Deployment Center. The user views environments, sets up installations, and generates deployment scripts. Specify a user name and remember it for logon later.

   - **-password** (required)
     
     Specify the password for the Deployment Center user. Remember it for logon later.

   - **-repoServiceName** (optional)
Specify the name for the Deployment Center repository service. If you omit this argument, the Deployment Center names the service **RepositoryService**.

If the service name matches an existing service name, the installation fails and returns an error. You must either rename or remove the conflicting service name, or specify a different name.

**-serviceName** (optional)

Specify the service name for the Deployment Center web server. If you omit this argument, the server is not installed as a service and must be started manually.

**-serviceDName** (optional)

Specify the service display name for the Deployment Center web server.

For example:

```bash
deployment_center.bat -install -serverDir=D:\apps\deployment_center\server -serverPort=9090 -repoDir=D:\apps\deployment_center\repo -repoPort=9595 -user=dcadmin -password=dcadmin -serviceName=DC_Service -repoServiceName=DC_Reposerv
```

6. In this step, the script prompts you to choose whether to participate in the Product Excellence Program.

Choose **Y** to participate or **N** If you do not want to participate.

**Privacy statement**

The Product Excellence Program helps Siemens PLM Software understand how customers use our products and assists us in improving our products. The program is anonymous and participation is voluntary.

The Product Excellence Program is designed to protect the privacy of the user and the intellectual property created through the use of Siemens PLM Software products.

**How does it work?**

The Product Excellence Program is used to collect data about your installation, the features you use and how you use them. The data is sent to Siemens PLM Software for analysis. By examining usage patterns from a large number of people, we gain insight into how the products are used and how to improve the software in future releases. Data collection occurs in the background as you use the software and does not affect performance or functionality.

**What data is sent?**

The data collected can vary by product and by release as we gain more insight or add new capabilities. The Product Excellence Program may collect information on the functions utilized, the operating environment (for example, OS, RAM, graphics, etc.), product version, or other indications of user interactions. This data is solely used by Siemens PLM Software to improve our products and is never shared with any third party.

There is no contact information in the data and Siemens PLM Software will not contact you by phone or email as a result of the data collected. Absolutely no information about the data you create or manage is collected.
Participation is optional.

You can review and control your participation at any time.

You can opt out of the Product Excellence Program during installation or after installation. Change your decision to join or opt out at any time.

What does this mean for Deployment Center?

The following table provides examples of the type of usage data that is collected by Siemens PLM Software and, equally as important, clarifies what data is not collected by Siemens PLM Software.

<table>
<thead>
<tr>
<th>What we collect</th>
<th>What we do not collect</th>
</tr>
</thead>
<tbody>
<tr>
<td>The data collected contains information about how Deployment Center is deployed and used, for example:</td>
<td>The data collected does not contain user information or intellectual property, including:</td>
</tr>
<tr>
<td>• Deployment Center version</td>
<td>• Contact information</td>
</tr>
<tr>
<td>• Client browser type and locale</td>
<td>• Information about data created or managed in Deployment Center</td>
</tr>
<tr>
<td>• Participation count</td>
<td>• IT infrastructure identifiable information, including server IP addresses and host names</td>
</tr>
</tbody>
</table>

7. After the installation is complete, the script returns the URL to access Deployment Center. Record the URL to log on to Deployment Center. The URL has the form:

```
http://host:serverPort/deploymentcenter
```

The script also returns the location of the installation log files. The installation log file also contains the Deployment Center URL. In the Deployment Center root directory, navigate to:

```
deployment_center\logs\deployment_center_debug_timestamp.log
```

If you experience a problem in your installation or upgrade, see Troubleshoot Deployment Center installation or upgrade.

How to specify an existing repository during Deployment Center installation

If you have an existing directory that contains software kits on a separate server from Deployment Center, you may prefer not to duplicate that directory on the Deployment Center server. You have the option to create a mapped drive to the server and share the repository path with Deployment Center. During installation, specify the mapped drive to the shared location containing existing software.

On the server where you are installing Deployment Center, you need to ensure that the local system account can access the location where you want to map the drive. In the following procedure, the system user creates a scheduled task to mount the mapped drive to the repository.

1. Move your Teamcenter software into a software directory under your repository location. Be sure the software kits have been unzipped and their folders are inside the software directory.

2. Open a command window. Map a drive to your alternate repository using the following command:

```
schtasks /create /tr "net use network-drive-letter:
```
Be sure that the specified user has write access to the shared directory. You need to restart the system for the scheduled task to run.

**Example**

```bash
schtasks /create /tr "net use Z: \10.134.68.64\repo Pa22w0rd /USER:dcadmin /persistent:yes" /tn "MountZDrive" /sc onstart /RU SYSTEM
```

This example mounts `\10.134.68.64\repo` as the `Z:` drive, and presumes the repository path provided to Deployment Center is `Z:\`.

3. When you install Deployment Center, specify the `network-drive-letter` in the `-repoDir` parameter, for example, `-repoDir=Z:`.

**Note**

Be aware that after you set the repository directory location, you can't change it later.

Continue defining your `deployment_center.bat` command arguments for the Deployment Center installation.

**Upgrade Deployment Center**

When you upgrade, all Deployment Center data is preserved, including software in the repository and deployed environment information. Be sure to check supported upgrade paths for Deployment Center before you proceed.

To upgrade Deployment Center:

1. Download the upgrade ZIP file for Deployment Center from the GTAC software Download page.
2. Extract the Deployment Center ZIP file to the server where you installed Deployment Center.
3. Stop the Deployment Center web server, including the service if it is running.
4. Check the supported JRE version of Java and the value of `JAVA_HOME`.
5. Open a command prompt window and navigate to the location where you extracted the Deployment Center upgrade software. Go to the `deployment_center` directory.
6. Run `deployment_center.bat` using the following arguments. All arguments are required for an upgrade.
   - `-upgrade` (required)
   
   Run in upgrade mode.
-user (required)
Specify the user name for Deployment Center to validate the upgrade.

-password (required)
Specify the password for the Deployment Center user to validate the upgrade.

-serverDir (required)
Specify the full path to the directory for the Deployment Center installation.

For example:

```
deployment_center.bat -upgrade -serverDir=D:\apps\deployment_center\server
-user=dcadmin -password=dcadmin
```

7. The script prompts you to choose whether to participate in the Product Excellence Program. This program anonymously shares information with Siemens PLM Software about how you deploy Deployment Center software. This information assists us in improving future releases of Deployment Center. Data is collected in the background as you use the software, without affecting performance or functionality.

Participation in the program is voluntary. If you do not want to participate, enter N. You can change your decision to join or opt out at any time.

The script returns the status of the upgrade and the location of the upgrade log files. If you experience a problem in your upgrade, see Troubleshoot Deployment Center installation or upgrade.

**Participate in the Product Excellence Program**

During installation or upgrade of Deployment Center, you were offered the opportunity to participate in the Siemens PLM Software Product Excellence Program. This program anonymously shares information with Siemens PLM Software about how you deploy Deployment Center software. This information assists us in improving future releases of Deployment Center. Data collection occurs in the background as software is used and does not affect performance or functionality.

After installation or upgrade, you can change your participation choice. After you log on to Deployment Center, you can:

**Join the program**

Enter the URL:

```
http://dc_server:port/deploymentcenter/rest/softwareanalytics/updatesoftwareanalytics?enabled=true
```

Deployment Center responds that enabling participation succeeded.

**Opt out of the program**

Enter the URL:

```
http://dc_server:port/deploymentcenter/rest/softwareanalytics/updatesoftwareanalytics?enabled=false
```

Deployment Center responds that disabling participation succeeded.

After you confirm your request, log out of Deployment Center and log on again for the change to take effect.
Uninstalling Deployment Center

If you need to remove Deployment Center from your system, follow these direction to uninstall Deployment Center.

Gather the following information before you begin. Service names and directory locations are specified during Deployment Center installation using deployment_center.bat install script arguments. You can check for Deployment Center service names using Windows Services.

- The Deployment Center service name, which is optional. If a Deployment Center service was created during installation, it was specified using the -serviceName argument.

- The Deployment Center repository service name. This service was installed using either the default name RepositoryService or optionally specified using the -repoServiceName argument.

- The top level directory where you installed Deployment Center. The location was specified when you installed Deployment Center using the -serverDir argument.

- The Deployment Center software repository directory, which can be on a different machine. The location was specified when you installed Deployment Center using the -repoDir argument.

1. Open a command window on the server where Deployment Center is installed.

2. Stop the Deployment Center repository service:

   `net stop repository_service_name`

   Then delete the repository service:

   `sc delete repository_service_name`

3. If Deployment Center is not installed as a service, skip this step.

   If Deployment Center is installed as a service, stop it:

   `net stop deployment_center_service_name`

   Then delete the Deployment Center service:

   `sc delete deployment_center_service_name`

   After you have deleted the service, you may close the command window.

4. Delete the entire directory where you installed Deployment Center.

5. Delete the following directories from the Deployment Center software repository directory. The location was specified when you installed Deployment Center using the -repoDir argument.

   `dc_contributions`

   `deploy_scripts`

   `system`
Troubleshoot Deployment Center installation or upgrade

If you have difficulty with installing or upgrading Deployment Center, look for log files in the location where you extracted the Deployment Center installation ZIP file:

The logs are in deployment_center\logs. The logon URL is available from the installation log.

- **dc_install_error.log**
  Provides a description of the installation or upgrade failure.

- **dc_install_debug.log**
  Provides detailed information about the installation or upgrade operation.

- **dc_database_upgrade_error.log**
  Provides a description of the database upgrade failure.

- **dc_database_upgrade_debug.log**
  Provides detailed information about the upgrade operation.

If the default RepositoryService service name or a service name specified by -repoServiceName matches an existing service name, the installation returns an error. Check the service names on the server for conflicts. You can either change a conflicting service name or specify a different repository service name using -repoServiceName.

Be sure you run the Deployment Center installation script from a location with no spaces or quotation marks in the path. For example, if your installation script is located in:

D:\DC Kits\deployment_center_2.2\deployment_center

you may get the following error:

```
Could not find or load main class Kits\deployment_center_2.2\deployment_center\jar\com.dc.jrechecker.jar
'JAVA_HOME' is set to an unsupported version of Java.
```

The D:\DC portion of the path is ignored. Remove the space in the path:

D:\DCKits\deployment_center_2.2\deployment_center

Start Deployment Center

Before you access the Deployment Center web application from a web browser, start the Deployment Center web server. You can choose either of the following ways:

- Automatically start the server as a service
  If you specified the -serviceName argument during installation, the Deployment Center web server is installed as a service and started automatically.
  The serviceName argument, if specified, provides the Services display name. Otherwise, the service name defaults to the internal name specified by the serviceName argument.

- Start the server manually
If you did not specify the service arguments, start the Deployment Center service from its startup script. Navigate to the server directory specified by the `-serverDir` argument when you installed Deployment Center and run the `startdc.bat` script.

If you experience a problem starting Deployment Center, see Troubleshoot Deployment Center operations.

Log on to Deployment Center

1. Enter the Deployment Center URL in the web browser. The form of the URL is:

   http://host:serverPort/deploymentcenter

   `host` is the server where Deployment Center is installed.

   `serverPort` is the port number specified by the `-serverPort` argument for the Deployment Center installation script.

   **Tip**

   The URL is displayed in the command window after Deployment Center installation finished.

   The URL is also in the installation log file. In the Deployment Center directory, look for:

   `deployment_center\logs\deployment_center_debug_timestamp.log`

2. Enter the user name and password as specified by the `-user` and `-password` arguments for the Deployment Center installation script.

   The Deployment Center home page appears.
If you experience a problem in logging on to Deployment Center, see Troubleshoot Deployment Center operations.

**Troubleshoot Deployment Center operations**

You can consult log files if you have difficulty with Deployment Center operations, such as:

- Inability to log on to Deployment Center.
- Inability to add software, applications, or components to a Deployment Center environment.
- Failure to create a deployment script.
- Deployment Center operation failures such as internal server error, display problems, or missing configuration files.

Find the log files on the Deployment Center server in `deployment_center_server_dir\logs`:

- **web_server_debug.log**
  Provides a detailed description of Deployment Center operations.

- **web_server_warn.log**
  Provides a description of operation failures as well as other warnings.

- **web_server_error.log**
  Provides a description of operation failures.

- **spring_api.log**
  Provides third-party application information.

- **hibernate_api.log**
  Provides third-party application information.

**Backup and recovery procedures**

You can back up key Deployment Center software files for recovery in case of a failure. You can recover from a configuration error that cannot be otherwise changed in the system. You can also recover from a database corruption.

Deployment Center installation paths and parameters are referenced in the procedures. You need to be familiar with the Deployment Center installation process. In the procedures, `DC-install-dir` is the installation path to the Deployment Center.
Back up the Deployment Center system

**Note**
Best practices for backup frequency are after each deployment and when you add software to Deployment Center.

1. Stop the Deployment Center server.
2. If you set up a repository service, stop the service. The service name (and display name if specified) were set when you installed Deployment Center using the `-serviceName` and `-serviceDName` arguments.
3. Copy these database files to a safe location:
   - `DC-install-dir\serverDir\db\deploy_center.h2.db`
   - `DC-install-dir\serverDir\db\deploy_center.trace.db`
4. Record the list of software packages you downloaded to the repository. The repository directory is set to the location specified by the `-repoDir` during Deployment Center installation, for example:
   - `DC-install-dir\repository\software`
   **Note**
   If you already recorded software packages and there are no changes to the repository since the last backup, you can skip this step.
5. Restart the repository service and/or start Deployment Center.
   If you encounter a problem, stop Deployment Center and the repository service and replace the database files from your backup. If that doesn't fix the problem when you restart Deployment Center, download and replace the software packages in the repository.

Reinstall Deployment Center

If your recovery is not successful, you may need to reinstall Deployment Center.
1. Stop Deployment Center, and delete the repository service.
2. Either move or rename the Deployment Center installation, in case you need to access the software directories to repopulate the repository.
3. Reinstall Deployment Center into the same location, using the same paths and ports as the original installation. To restore the database files, the repository path specified by `-repoDir` must be identical to the previous installation.
4. Update the repository with the same software packages that you recorded from the previous backup. If the repository software packages are safely available from the installation that you moved or renamed, copy those files.
   Wait for Deployment Center to update and recognize the software.
5. Stop the Deployment Center server.

6. If you set up a repository service, stop the service.

7. Copy the backup database files to your current Deployment Center installation:

   \texttt{DC-install-dir\serverDir\db\deploy\_center.h2.db}

   \texttt{DC-install-dir\serverDir\db\deploy\_center.trace.db}

8. Restart the repository service and/or start Deployment Center.

   Deployment Center should be restored to the state it was in from the last backup.
Chapter 3: Managing Teamcenter environments

Manage the repository

Deployment Center stores the software kits that you need when you want to install, patch, or upgrade your Teamcenter environment. Deployment Center helps you determine whether you have everything you need for an installation, patch, or upgrade. Deployment Center displays messages explaining dependencies on missing software and how to proceed.

The Deployment Center software repository makes the software available when you want to perform installations or updates. The software kits provide the application choices that you select in Deployment Center when you install, patch, or upgrade. To access the repository, log on to Deployment Center and click SOFTWARE REPOSITORIES.

Locate the repository

The Deployment Center repository directory structure contains a software directory for your unzipped software downloads. The -repoDir argument of the Deployment Center installation script specifies the repository directory path to the software location where you store the downloaded software.

Manage software kits

Deployment Center scans for software in unzipped subdirectories of the software directory. Deployment Center can read directories nested one level under software. You can organize your software in subdirectory categories, such as softwareActiveWorkspace. These subdirectories must contain unzipped software directories.

Deployment Center does not scan further than the first level subdirectories of software. Software nested under another software folder is not supported. Software that is in a ZIP file or placed in another location on the server is not scanned.

Review repository contents

The Contents tab displays information about the software kits that are scanned and recognized by Deployment Center. The Software Media table displays information about each software kit. Deployment Center refers to the base Teamcenter software as Foundation software.

Software kit information includes the name and version of the software, its release type and platform, and whether it has a dependency, as noted in the DEPENDENCIES column. If so, retrieve the missing software and copy it (unzipped) into the repository. Dependent software must be present to successfully choose software to install or update. Software kit information includes the Teamcenter environments where the software is installed, as noted in the USAGE column.

The repository services polls for software at 10 second intervals and updates this list as necessary.
Put software kits into the repository

1. Download the software kits for the software versions that you want to deploy in your Teamcenter environment. Be sure to get the major release, the latest minor release, and any patches you want to apply to your Teamcenter environment.

   For example, if your target Teamcenter version is 12.2, download the Teamcenter 12 major release and the Teamcenter 12.2 minor release.

2. Unzip the software kits and copy the unzipped directories to the software subdirectory in the repository.

   **Tip**

   The **Repositories** page displays the first platform it finds for a specific software download version. If your environment contains multiple operating systems, you need only one software kit for each version of software in the repository. For example, if you have multiple software kits for the same software version on Windows, Linux, and UNIX, put only one of the operating system kits in the repository.

   However, be sure you place all software kits for the same version for all operating systems in the location specified by the `-softwareLocation` argument when you deploy the generated scripts.

3. Log on to Deployment Center, and click **SOFTWARE REPOSITORIES**.

   The **Software Repositories** page opens the **Contents** of the repository and displays the **Software Media** table.

4. Check the list of software to verify that it is correct and complete for your planned deployment. Note whether there are missing dependencies as noted. If so, retrieve the missing software and copy it (unzipped) into the repository and check again.

If you experience a problem in adding software to the Deployment Center repository, you can try to troubleshoot the repository service.

Remove a software kit from the repository

When a software kit is no longer being used in a registered Teamcenter environment, you can delete it from the list of software in the repository:

1. Stop the Windows repository service. This is the service specified when you installed Deployment Center, named **RepositoryService** by default.

2. Delete the obsolete software kit subdirectory from the repository software directory.

3. Delete the software kit entry in the Deployment Center repotool.repository_checksum.xml file.

4. Delete the software kit from the Deployment Center. Highlight the software kit entry in the **Software Media** table of the repository and click **Remove**.

5. Restart the repository service.
Update repository properties

The **Overview** tab provides helpful information about the server and the repository. The system information includes disk space, which you can monitor when you put software kits in the repository. The available and used disk space values are updated when you add or change software kits in the repository.

You can edit some of the repository properties.

1. Click **Start Edit** to change the following information:

2. You can change the following information:

   - **Repository Name**
     Enter or update a name for the repository.

   - **Location**
     Enter or update a location for the repository. You can use any convention you wish, such as geographical location, the name of a facility, or another value that helps you define the location.

   - **Comments**
     Enter or update additional information that might be helpful to an administrator.

3. Click **Save Edits**.

   To cancel your changes, click **Cancel Edits**.

Troubleshoot the repository service

If you have difficulty with software not appearing on the **Software Repositories** page, you may have repository scanning issues or software file problems. The repository log files are located on the Deployment Center server in `deployment_center_server_dir\repotool\logs`:

- **media.Scanner.debug.log**
  Provides a detailed description of Deployment Center software scanning operations.

- **media.Scanner.error.log**
  Provides a description of software scanning operation failures.

- **tem.config.rest.service.log**
  Provides the communications information between the Deployment Center server and the repository scanning utility.
Manage environments

The Environments page lists all environments being tracked by Deployment Center. You can view, create, and delete environments. To access the list of environments, log on to Deployment Center and click ENVIRONMENTS.

Deploy software in an environment

The Deploy Software tab displays the selected software used by environments currently registered, which is necessary to proceed with deploying updates. If an environment is new, the selected software list may be empty. This is where you begin the software deployment process.

View an environment's properties

The Overview tab displays the environment's properties, such as the architecture, site, software version, applications, and components installed in the environment. Click an environment to learn more about it.

- Review the Properties for the environment.
  
  If an environment is new, the Overview displays the information used to create it.

  You can edit some environment properties.

- Review the Software, Applications Installed, and Components for an existing environment. You can click a selection to display more information in the right pane.
  
  If an environment is new, Software, Applications Installed, and Components may not be populated, so no Teamcenter environment information may be displayed.

Verify software, applications, and components

On the Environments page, you can review the software, applications, and components for each environment.

Software

You can verify the status of software for the selected environment. The list includes installed and pending software. You can select the software package to see additional software details.

Applications Installed

You can verify the status of applications for the selected environment. The list includes installed and pending applications. You can select an application package to see additional application details.

Applications are associated with their installed software, such as Search for Active Workspace.
Components

You can verify the status of components for the selected environment. The list includes installed and pending components. You can select the component package to see additional information about component settings. Components are associated with their applications, such as Indexing Engine and Indexer for Search.

The component information in the right pane offers two views of the information. In the upper right corner, you can choose the view:

Show all parameters

Required parameters view displays only required parameter information. Click to expand the view to display both required and optional parameters.

Show only required parameters

All parameters view displays both required and optional parameter information. Click to collapse the view to required parameters.

Edit environment properties

On the Environments page, you can edit some environment properties. Click the Overview tab to display the properties of the selected environment.

1. Click Start Edit to display the editable fields.

2. You can change the following information:

   • Environment Name
     Displays the name provided during setup for the environment. The environment name is specified in the -environment argument of the `send_configuration_to_dc` script.
     You can update the name for the environment.

   • Environment Type
     Displays the type of the environment. The available types are Integration, Development, Production, Test, and Training. The type is set to Production by default when the environment is registered, but you can select another type.

   • Location
     Displays the location of the environment. You can enter or update the geographical location for the environment, such as a city, the name of a facility, or another value that helps you define the location of the environment.

   • Comments
     Displays additional information entered by the administrator. You can enter or update information about the environment.
3. Click **Save Edits**.

   To cancel your changes, click **Cancel Edits**.

**Create environments**

You can create an environment for your planned deployment. When you are ready to add software to your new environment, Deployment Center displays only the versions of **Available Software** that are supported in a new environment.

**Create an environment**

1. Log on to Deployment Center, and click **ENVIRONMENTS**.
   The **Environments** page lists currently planned and registered environments.

2. On the far right below the command bar, click **Add**.

3. The new environment appears highlighted in the list. Choose **Overview** to display its information.

4. You can edit some of the properties, such as **Name** and **Type**. On the command bar:
   - Click **Start Edit** to edit properties. To save your changes, click **Save Edits**.
   - To cancel your changes, click **Cancel Edits**.

**Register environments**

Register your environments in Deployment Center by running the `send_configuration_to_dc` script on the corporate server that hosts each Teamcenter environment. If the environment is distributed across multiple servers, you must run the script on each machine that is part of the specific Teamcenter environment. The script sends configuration information about the applications and components that are currently installed to Deployment Center.

After the environment is registered, you can view its configuration information and verify the content. Deployment Center saves information about server machines deployed in your environments.

- **View the machines** used in deployed Teamcenter environments from the **MACHINES** tile on the Deployment Center home page.

- **Select a machine from a list of servers** when configuring components.

**Caution**

Before updating an existing registered Teamcenter environment, be sure that you run the `send_configuration_to_dc` script to update the environment information. Configuration changes performed locally on Teamcenter servers since the last time the `send_configuration_to_dc` script ran could be overwritten.
1. On the machine hosting the Teamcenter environment, install the supported version of the JRE and set JAVA_HOME to the location.

2. Open a command prompt window, and set TC_ROOT to the Teamcenter installation directory if it's not already set.

3. From the location where you installed Deployment Center, navigate to the additional_tools\Teamcenter\send_configuration_to_dc directory and find the send_configuration_to_dc.zip file. Copy and unzip the file. Place the extracted directory on the machine hosting the Teamcenter environment.

4. In the command prompt window on the Teamcenter host, navigate to the send_configuration_to_dc directory. Run send_configuration_to_dc.bat (Windows) or send_configuration_to_dc.sh (Linux or UNIX) using the following arguments. Required arguments are noted.

   -dcurl (required)
   Specify the URL you use to access Deployment Center.

   -dcusername (required)
   Specify the user name for Deployment Center as defined when installing Deployment Center.

   -dcpassword (required)
   Specify the password for Deployment Center as defined when installing Deployment Center.

   -environment (required)
   Specify a name to identify the environment being scanned. Because an environment is ordinarily identified by its site ID, this argument allows you to create a readable label that makes it easier to identify the Teamcenter environment.

   -config (optional)
   Specify the ID value for the configuration used when installing the Teamcenter environment. Specify this argument if multiple configurations are installed in a single TC_ROOT location using the TEM installer.

   For example:
   
   send_configuration_to_dc.bat -dcurl=http://dc_host:9000/deploymentcenter
   -dcusername=dcadmin -dcpassword=dcadmin -environment=Sandbox

   After the scan completes, the script returns the message:
   
   All operations completed successfully.

   If you experience a problem in registering environments with Deployment Center, see Troubleshoot registering environments.

Remove environments

You can remove an environment from the Environments list in Deployment Center.
1. Log on to Deployment Center, and click **ENVIRONMENTS**.
   The **Environments** page lists all environments.

2. Select the environment you want to remove.

3. Click **Delete** on the command bar and confirm the deletion.

   The environment is only removed from Deployment Center tracking. The Teamcenter system remains intact.

### Troubleshoot registering environments

You can consult log files if you have difficulty with sending an environment configuration to Deployment Center (using the `send_configuration_to_dc` script), such as:

- Sending configuration to Deployment Center fails.
- Inability to communicate with a Deployment Center server.
- Invalid credentials passed when sending environment configuration.

The registration configuration log files are located on the Teamcenter environment server in `send_configuration_to_dc_dir/logs`:

- `tem_config_scanner_error_timestamp.log`
  Provides a description of environment scanning operation failures.

- `tem_config_rest_service_timestamp.log`
  Provides the communications information between the Deployment Center server and the `send_configuration_to_dc` utility.

### View registered machines

View the machines used in deployed Teamcenter environments. Select a machine name from the list to view its properties, such as operating system, disk capacity, and free disk space.

1. Log on to Deployment Center, and click **MACHINES**. The **Machines** page lists all servers used by components in deployed environments. When you select a machine, the **Overview** provides information about the server where one or more software components are installed.

2. Verify that the properties for the server machine are what you expect.

Properties for a machine from a deployed environment include:

- **Machine Name** and **IP Address**
  Identifies the machine by the server name and IP address.

- **OS** and **OS version**
Displays the operating system type and version installed on the machine.

- **Local Time**
  Displays the current date, time, and time zone at the machine’s location.

- **Disk Capacity and Disk Free**
  Displays the total disk space and the free space available. The pie chart to the right displays the same information visually.

- **Last Update**
  Displays the last time information about this server was refreshed. The information is obtained and sent by the `send_configuration_to_dc` utility.
Chapter 4: Deploying software using Deployment Center

Installation, maintenance, and upgrade strategy

You can use Deployment Center to install, upgrade, or maintain a Teamcenter environment. You can choose which scenario applies to your needs.

Install a new Teamcenter environment

You can install software into a new environment. To prepare for a new installation:

- Be sure you have a Teamcenter environment set up in Deployment Center.
  - Create a new environment if you are starting with a new Teamcenter installation.
  - Register an existing environment if you are installing new software in an existing environment. This might be the case if you are installing Active Workspace in an existing Teamcenter environment.

- Put your unzipped installation software kits in the repository. If you have an existing environment, put the software you used to install it into the repository as well.

Begin the installation by following the deployment procedure.

Maintain an existing registered Teamcenter environment

You can update software or components in an existing registered environment. Maintaining an existing registered Teamcenter environment means updating software or components for your current version, as permitted by Deployment Center. Be sure you send existing environment configuration information to Deployment Center. You must have the source software for your environment in the repository.

Upgrade or patch an existing Teamcenter environment

You can upgrade or patch software in an existing environment. Software dependencies are noted in the Deployment Center repository and in the Software deployment task. Deployment Center displays messages explaining issues with upgrade software or dependencies.

To prepare for a software upgrade or patch in an existing environment:

- Be sure you have a registered existing environment in Deployment Center before you start an upgrade.

- Be sure to send existing configuration information from the environment to Deployment Center using the send_configuration_to_dc script.

- Put your unzipped installation software kits in the repository and check for messages about dependencies.
Deployment procedure

The **Deploy Software** page displays each step in the deployment progress bar. In each task, Deployment Center prompts you to make selections and provide information. At any time in the process, you can save your work and exit. The settings are stored in Deployment Center, and you can return to the deployment process at your convenience.

- *Dark blue* means the task is available and currently selected.
- *Medium blue* means the task is available.
- Gray means the task is not available yet. These tasks become available as steps within the previous tasks are completed.

Click a chevron to go to that task. You can revisit any task you previously completed to make changes. For example, if you completed the **Components** task, you can still return to the **Applications** task and make changes.

**Caution**

Before proceeding to update an existing registered Teamcenter environment, be sure that you run the `send_configuration_to_dc` script to provide current environment information in Deployment Center. Configuration changes performed locally on Teamcenter servers since the last time the `send_configuration_to_dc` script ran could be overwritten.

**Deploy software**

1. Open the **Environments** page and choose the environment where you want to install or update software. The **Deploy Software** page provides access to the deployment tasks.

2. **Software**

   Choose the software to install or update. The software determines the list of available applications. For example, if you choose Active Workspace software, you can install the applications it provides.

   - **Pending Install** software will be installed or updated during deployment.
   - **Pending Update** software is already installed in your environment, but it needs an update to support other selected software.
   - **Installed** software is already installed in your environment and needs no updates.

3. **Options**

   Choose single server or multiple server deployment for **Environment Type**.

   Choose the Java or Windows architecture for **Architecture Type**.

4. **Applications**
Choose the applications. The list of available applications depends on the software selected in the **Software** task. Some applications may be automatically selected for you by default.

Each software package can have one or more applications in its bundle. Applications provide business logic, data model, work processes, and administration data for specific business uses, industries, or integrations. In this step, you do not need to know details of your network or configuration of software or hosts.

5. **Components**

Choose and configure components. Components run on the specified servers in your environment. Some components are automatically selected for you as required by the applications selected in the **Application** task. You need to configure any component that is not listed as **100%** complete.

Component status:

- **Pending Install** components will be installed during deployment.
- **Pending Update** components are already installed in your environment, but they need updates to support your selected applications.
- **Installed** components are already installed in your environment, and they don't need any updates.

You need to know the server hosts on which components will be installed or updated, user names and passwords for the server component, and component URLs. Some components may have additional required or optional settings.

6. **Deploy**

Generate deployment scripts. This task is not available until the **Components** task is complete.

Deployment scripts contain the information you configured in Deployment Center for each of the servers in your environment. The scripts install the software, applications, and components onto each target machine in your environment.

7. **Run the deployment scripts.**

After the scripts and software ZIP files are generated, copy them to each target machine and run them.

You can run the deployment script in diagnostic mode to determine whether your script has any errors before updating the target machine.

**Maintain your environment**

Deployment Center can maintain applications or components in a registered Teamcenter environment. Maintenance is making changes to your existing environment. You are not performing upgrades or applying patches. Before you perform maintenance, you must:

- Be sure you have a **registered existing environment** in Deployment Center before you perform maintenance on its components.
• Send existing configuration information about a registered environment to Deployment Center using the send_configuration_to_dc script.

• Put your current environment's software in the repository and check for dependencies.

Perform maintenance

The tasks for updating software or components are similar to installation:
1. Open the Environments page and choose the environment you want to maintain. Begin the Deploy Software tasks.
2. Software
Choose software from the Available Software list. The software selections determine the list of available applications that you can update. Applications that must be updated are automatically selected. The Selected Software list displays currently installed versions.
3. Options
Either single server or multiple server deployment is selected for Environment Type. If you previously had a Single Box environment, you can choose Distributed; however, you will need to update the server information for affected components.

If an environment is already deployed on multiple servers, Single Box is not available.

Architecture Type for your environment is automatically selected and can't be changed.
4. Applications
Applications that are already installed are automatically displayed. You can add applications from the list. Applications that display a Pending Install status are waiting for deployment. Applications that are installed but need updates to support your selected software display the Pending Update status.
5. Components
It's possible that a selection from the current update may cause a previously configured component to need more information, especially if you add applications or move from a Single Box environment to a Distributed environment.

Components that are not yet installed display the Pending Install status. Components that are installed but need updates to support your selected applications display the Pending Update status.

Components display the % configured. If it's less than 100%, complete the required parameter values. Components that are not impacted can be ignored (showing 100% configured).
6. Deploy
Generate deployment scripts for the update. This task is available when the other tasks are complete.

Deployment scripts contain the information you configured in Deployment Center for the selected environment.
7. Run the deployment scripts
After the scripts and software ZIP files are generated, copy them to each target machine and run them.

Upgrade or patch your environment

Deployment Center can upgrade or patch software on a registered Teamcenter environment.

Deployment Center software upgrades follow a process that is similar to installation. When you want to upgrade or patch your environment, choose the target version of software you want to apply. Deployment Center determines what to upgrade or patch based on what is required by the selected target release and selected applications.

You may not be required to put source release software kits in the repository. Deployment Center constructs source to target release mapping using the environment's current configuration files sent in the report from `send_configuration_to_dc`. Deployment Center analyzes the target release information to construct the mapping at the time you choose to upgrade. Sometimes, Deployment Center may require the source software if the target release doesn't provide adequate mapping information. If there is missing software that is required, Deployment Center displays messages telling you about the dependency and how to proceed.

Before you perform an upgrade, you must:

- Run the `send_configuration_to_dc` script on the target servers to send the latest environment configuration information to Deployment Center.

- Download, unzip, and put the software upgrade or patch kits in the Deployment Center repository. Check the repository for software dependencies and messages about missing software.

Perform upgrades or patches

The procedure for upgrading is similar to installation.

1. Open the Environments page and choose the environment where you want to upgrade software. Begin the Deploy Software tasks.

2. **Software**

   Choose the target upgrade software from the Available Software list. The Selected Software list displays currently installed versions and latest pending versions for the environment. If the software you need is not available, check whether it was listed in the repository.

   If missing software is required, Deployment Center tells you about the dependency and how to proceed.

3. **Options**

   Either single server or multiple server is selected for Environment Type. If you previously had a Single Box environment, you can choose Distributed; however, you will need to update the server information for components.

   If an environment is already deployed on multiple servers, Single Box is not available.

   The Architecture Type for your environment is automatically selected and can't be changed.

4. **Applications**
Applications that are already installed are automatically included for upgrade. You can add other applications from the list. Applications that display a Pending Install status are waiting for deployment. Applications that are installed but need updates display the Pending Update status.

5. **Components**

Components that are not yet installed display the Pending Install status. Components that are installed but need updates to support your selected applications display the Pending Update status.

It's possible that a selection from the current upgrade may cause a previously configured component to need more information.

Components display the % configured. If it's less than 100%, complete the required parameter values. Components that are not impacted can be ignored (showing 100% configured).

6. **Deploy**

Generate deployment scripts for the upgrade. This task is available when the Components task is complete.

Deployment scripts contain the information you configured in Deployment Center for the selected environment.

7. **Run the deployment scripts**

After the scripts and software ZIP files are generated, copy them to each target machine and run them.

---

**Deploy in a multiple operating system environment**

If your deployment environment has multiple operating systems, you must place the software kits for only one of the operating systems into the Deployment Center repository. Deployment Center only recognizes the first kit it finds for a specific version. For example, if you have Teamcenter 11.4 software kits for Windows, Linux, and UNIX, put only one system's software kit in the repository.

1. Put the software kits for one of the operating systems for your environment in the Deployment Center repository. Don't drop all the software kits for the same version into the Deployment Center repository, as you can't choose which operating system Deployment Center finds first.

2. Run Deployment Center and enter your configuration settings. Deployment Center prompts for the selections and parameters, regardless of operating system, to deploy the applications and components.

3. Generate the deployment scripts. Deployment Center creates the deployment scripts correctly for the supported platforms in your environment.

4. **Stage the software kits** for all of the operating systems in the repository, and be sure it's accessible to the target environment servers.

5. Run the `deploy.bat` script on each of the target servers, specifying the repository location in the `-softwareLocation` argument.
Software task

**Deploy Software** Overview

*1 Software*  *2 Options*  *3 Applications*  *4 Components*  *5 Deploy*

**Selected Software**

In this task, select the software to install from the list of installable applications. The software you select determines the list of applications available in the **Applications** task. The **Selected Software** list displays both current and pending installations for the environment.

Selecting a minor version of software also automatically adds its underlying major version. For example, choosing the minor version Teamcenter 12.1 automatically adds the major version Teamcenter 12.0.

A selected version of Teamcenter software can make additional software available. The **Available Software** choices are dependent on a version compatible with the selected version of Teamcenter. For example, if Teamcenter 11.6 is selected, Active Workspace 4.1 could become available.

1. In the **Software** task, click **Edit Selected Software** to add software. This **Available Software** panel displays the software choices.

2. The **Available Software** panel lists software from the repository. The software status displays information about the software kit. After making your selections, click **Update Selected Software** to add them to **Selected Software**.

   If the software you need is not listed, you must **add it to the repository**. Add software as needed, but you may have to choose applications and configure components before deployment.

3. When your **Selected Software** list is complete, go to the **Options** task.

Options task

**Deploy Software** Overview

*1 Software*  *2 Options*  *3 Applications*  *4 Components*  *5 Deploy*

**Options**

In this task, choose the deployment options for your environment.

1. Choose the **Environment Type**.
   - Choose **Single box** to install all components on a single server.
     After you define **Machine Name**, **OS**, and **Teamcenter Installation Path** for one of the components, those values are adopted by the other components.
     If an environment is already deployed on multiple servers, this type will not be available.
   - Choose **Distributed** to install components on any server in an environment. **Machine Name**, **OS**, and **Teamcenter Installation Path** configuration values are shared only with other components that are required to be on the same server.
This type may be selected for you if your environment is already set up as a distributed environment.

You can change the value from Distributed to Single box if an install or an update is not in progress. For configured components that are not yet installed, Machine Name, OS, and Teamcenter Installation Path are changed to the values specified for the corporate server component.

2. Choose Architecture Type.
   - Choose Java EE to filter component choices to the Java EE architecture.
   - Choose .NET to filter component choices to the Windows .NET architecture.

   If your environment already has deployed one of the architectures, the type is selected and can't be changed.

3. When your selections are complete, click Save Environment Options to go to the Applications task.

Applications task

In this task, choose applications for the software you selected. The list of available applications is determined by the Selected Software packages. Each software package includes one or more applications as a part of its bundle. The applications contain components, which you select later in the Components task.

Some applications are automatically selected based on your Selected Software. For example, if you choose Active Workspace, the Selected Applications list displays the applications that are configured as required for installation.

Applications that have a Pending Install status are waiting for installation deployment to complete. Applications that have a Pending Update status are already installed but need an update to support other selections.

   1. In the Applications task, click Edit Selected Applications to add applications.

   The Available Applications panel displays the application choices.

   2. In Available Applications, choose the applications to install. If you choose an application that has one or more required applications associated with it, the associated applications are automatically selected. Click Update Selected Applications to add them to the Selected Applications list.
3. You can add or remove applications as long as they are not already installed. Selected applications show the Pending Install status. When your Selected Applications list is complete, go to the Components task.

**Components task**

In this task, configure components for installation. Components provide the functionality for your environment. The Selected Components list displays required components that are automatically added from the Selected Applications list. Selected Components also displays optional components that were either already installed or previously selected. You can add more optional components from the Available Components panel.

Components that have a Pending Install status are waiting for installation deployment to complete. Components that have a Pending Update status are already installed but need an update to support other selections.

Some administrative tasks require that you have server names, user names, passwords, URLs, and other information available for the deployment. The following conditions may apply during component configuration:

- If a server machine was previously deployed in another environment or is specified in the current deployment for a different component, you can select it from the Machine Name list.

- If a component has a dependency on another component that is already defined, those values are shared with dependent components. This means that the component displays some percentage of completion.

- If you have not configured a component, the state may be either Start or some percentage complete if it has a shared dependency.

- When you are defining parameter values, some fields may not be editable. For example, if the component is already deployed in an environment, some parameters can't be changed (such as Machine Name and OS).

- If you selected the Single Box environment type in the Options task:

  - Specifying a Machine Name, OS, and Teamcenter Installation Path for one component shares those values with the remaining components. If you change these values on one of the components, the changes are propagated to the other components when you save.

  - If a component is already selected or installed, it is only listed as an available component if multiple instances of that component are supported.

**Add a component**

1. In the Components task, click Add component to your environment to add components.
The **Available Components** panel displays the optional component choices.

2. In **Available Components**, select the components to install. Then click **Update Selected Components** to add them to the **Selected Components** list.

3. In **Selected Components**, the **COMPLETE** column displays the state of completion for required component settings.

4. Click a component in the list to display its parameters in the right panel. This panel initially displays only required parameters. You must enter values for settings that appear in required parameters view. You can toggle the view between required parameters and all parameters.

   **Show all parameters**

   Required parameters view displays only required parameter information. Click to expand the view to display both required and optional parameters.

   **Show only required parameters**

   All parameters view displays both required and optional parameter information. Click to collapse the view to required parameters.
5. Completing all of the required settings pushes the state to 100% complete. If you don't have all the information you need, you can save your settings at any time and return to finish them.

For example, if you are installing the corporate server, required parameters include machine name, platform, Teamcenter installation path, and administrative user information. If you expand to Show all parameters, the corporate server displays additional optional settings.

6. When you are finished entering settings, click Save Component Settings.

7. The next component that is not complete appears in the right-side panel. When all Selected Components are 100% configured, go to the Deploy task. The Deploy task is not available until the Selected Components are all complete.
Remove a component

You can remove a component from the list, provided that the component:

- Is optional.
- Has a status of Pending Install.

Dependent components that were added to the environment with the main component are also removed for the same server. Other components of the same type are not removed. For example, if you have two server pools, removing one server pool removes its dependents but the other server pool remains.

1. In Selected Components, click the component you want to remove.

2. From the command bar, click Remove.

You are prompted to confirm removing the component from the list.

Deploy task

In this task, generate deployment scripts for the components you want to install. Deployment scripts contain the information you configured in Deployment Center for the selected environment. The scripts install the software, applications, and components on to each target machine in your environment.

When the scripts are finished, the Deploy Instructions panel displays information about the deployment and instructions for proceeding with the deployment. You must copy the scripts to each target machine and run them to complete the installation.

1. To generate deployment scripts, click Generate Install Scripts.

   Deployment Center generates installation scripts, and reports information about the scripts in the right panel.
2. In the **Deploy Instructions** panel, you can view the report about the deployment, including the location of the deployment scripts and the instructions for continuing the deployment.

   - **Script Generation Date** displays the time stamp for the local date and time of script generation.
   - **Deployment Overview** describes the deployment covered by the scripts.
   - **Software To Be Installed** lists the software required to deploy the components.
   - **Software Needed For Install** lists software that is already installed on the machine but is still needed for this process to deploy other components.
• **Deploy Script Directory** displays the path to the location of the ZIP files containing the generated scripts. Go to the ZIP file directory and check for one or more ZIP files corresponding to the machines in your Teamcenter environment. Look for the **Deploy_Instructions.html** file, which contains the same information and instructions that you reviewed in the report.

• **Deploy Scripts** displays the ZIP files that were generated for each server along with the associated component names. Each ZIP file contains the installation scripts for a single server.

If all components are to be installed on the same machine, there is only one ZIP file. The ZIP file name ends with the target machine name where you run the script. For example, if the ZIP file is named `20180511_202452EDT__Sandbox_LM6W006.zip`, it runs on the LM6W006 machine. Run an installation script only on its designated machine.

3. After you determine that the scripts you need are in the directory, you can proceed with the deployment.

### Run the deployment scripts

The generated scripts are saved in the repository staging area on the Deployment Center server. The `-repoDir` argument of the **deployment_center.bat** installation script set the path to the **staging_directory** location. The **Deploy_Instructions.html** file is saved to the same repository. It contains the same information and instructions that you reviewed in the Deployment panel of the **Deploy** task.

**Caution**

Before you deploy on an existing Teamcenter environment, be sure that you ran the **send_configuration_to_dc** script to update the configuration information in Deployment Center before making changes. Configuration changes performed locally on Teamcenter servers since the last time the **send_configuration_to_dc** script ran could be overwritten.

1. In the **Deploy Instructions** panel of the **Deploy** task, find the **Deploy Script Directory** section and take note of the path to the repository staging directory.

   On the Deployment Center server, open a file explorer and navigate to **staging_directory**. There may be one or more subdirectories under the staging directory following the pattern:

   ```
   environment_name\install\timestamp
   ```

   *environment_name* is the name of the Teamcenter environment for your deployment, and *timestamp* is the date and time that the deployment scripts were generated.

2. Determine which subdirectory you need, and find the installation ZIP files and the **Deploy_Instructions.html** that you generated. The ZIP files have the naming convention:

   ```
   timestamp_environment_host-name.zip
   ```

3. Before you run the deployment scripts, make them available to the designated servers in your Teamcenter environment.
Copy the ZIP files to a directory that is accessible to the servers in your environment using one of these methods:

- **Copy the ZIP files directly to each server and unzip them**
  
  Select this method if you want to run the deployment locally on the machine. You must copy the correct ZIP file that matches each server. Be sure the server host name matches the `host_name` in the ZIP file name.

- **Copy the ZIP files to a shared location, unzip them, and map a drive on each server**
  
  Select this method if you want to run the deployment from a common location accessible to all the servers in the Teamcenter environment. You must share the deployment location by mapping a drive to it on each server.
  
  Choose this method when you have multiple operating systems in your environment.

If you unzip on a UNIX or Linux system, be aware that path and file names are in mixed case. Avoid converting path and file names to lowercase, as paths are case sensitive on these systems. See the documentation for your ZIP utility for information.

4. Set `JAVA_HOME`.

   `JAVA_HOME` can be set to the Java JDK location or the Java JRE location.

   If you are installing the Active Workspace Client Java EE, `JAVA_HOME` must be set to the Java JDK location.

5. If a Teamcenter server manager is running, stop it.

6. Open a command prompt window and navigate to the location where you unzipped the files.

   Be sure you logged in as an Administrator or a user with administrative privileges before executing the deployment script.

7. Run the `deploy.bat` (Windows) or `deploy.sh` (UNIX or Linux) script.

   On a UNIX or Linux system, be sure to run the script in the KornShell (`ksh`) to avoid an error.

   The `deploy` script uses the following arguments:

   - **-dcusername**
     
     Specify the user name for Deployment Center as defined when installing Deployment Center.

   - **-dcpassword**
     
     Specify the password for Deployment Center as defined when installing Deployment Center.

   - **-diagnosticChecks** (Optional)
     
     Run a diagnostic validation of the deployment. Diagnostic mode checks whether the deployment tasks in the `deploy` script can be completed successfully on the target machine. Diagnostic mode does not perform any updates during validation.

     As a best practice, run the deploy script in diagnostic mode to validate operating system and database credentials, ports, installation paths, FSC unique IDs, and so on.
The log output provides success and failure information. Make any necessary corrections in Deployment Center and regenerate your deployment scripts, and run them again in diagnostic mode. Repeat until all the errors are addressed.

-softwareLocation
Specify the location in the -softwareLocation argument, for example,

-softwareLocation=D:\deploy_software

Tip
On Windows, you can omit specifying a software location when you run the deployment script. By default, the deployment script looks for the installation files on the M drive. The script takes a -softwareLocation argument, which you can omit if you map the shared drive to M. Be sure that the M drive is free on each target machine when you set up the mapped drive for the shared location.

Example
On a UNIX system:

./deploy.sh -softwareLocation=/kits/
-dcusername=dcadmin -dcpassword=dcadmin

When the installation is complete, the command prompt returns the message:

Deployment action successfully completed
Log files are in the directory where the installation ZIP file was unzipped.

Tip
If you have multiple Teamcenter servers in your environment, you can run deployment scripts on them in parallel. Running deployment scripts on Teamcenter servers in parallel is supported in releases 11.4 and later.

First, run the deployment script on the primary business logic server (the corporate server). Deployment Center uploads a dataset to the volume that contains key items from the TC_DATA directory, which provides the necessary information for deploying on the remaining servers. Then you can subsequently run the install or update deployment script on the rest of the servers concurrently, saving a significant amount of time.

If you experience a problem in running the deploy script, see Troubleshoot the deployment script.

Troubleshoot the deployment script
As a best practice, run each deployment script in diagnostic mode to test the deployment tasks without making changes to the system. Running the deployment scripts in diagnostic mode on the target machines validates the configuration entered in Deployment Center. Review the log files for
the **deploy.bat** or **deploy.sh** scripts and check for errors. No updates are made to the target machine during diagnostic validation. After you make the corrections in Deployment Center, regenerate the deployment scripts and run them in diagnostic mode. Repeat the cycle until all errors are addressed.

The log files are located in **deployment_script_dir\logs** on the Teamcenter machine where you are running **deploy.bat** or **deploy.sh**:

- **deployer_timestamp.log**
  - Provides a detailed description of the deployment operation.

- **scanner_timestamp.log**
  - Provides information about the software scanning performed during deployment.

Some of the more common errors you may see:

- Inability to locate the software required by the script (as specified by **-softwareLocation** or in the Windows mapped **M:** drive location).

- Inability to communicate with a Deployment Center server.

- A deployment failure.

When the deployment script runs on a Teamcenter server, it may encounter errors that are outside the scope of Deployment Center. If an error occurs when you run **deploy.bat** or **deploy.sh**, check the log file **deployer_timestamp.log**:

1. Look for a section titled **Diagnostic Checks Details**, which provides validation information.

2. Look for the point of failure and check whether the error notification is followed by a path to another log file on the Teamcenter server. For example, a Business Modeler IDE error can generate its own log file, which is saved on the Teamcenter server. The path to other log files is provided in the **deployer_timestamp.log**.

3. If the script reports an error without an explanation, check Teamcenter server error logs, which are located in either **TC_ROOT\logs** or the server's temporary directory (for example, **TEMP** or **TMP**).

In the Teamcenter error logs, look for an error that has a timestamp for approximately the same time as the error logged in **deployer_timestamp.log**.

**Tip**

Be sure that your environment software is not running when you deploy.

On supported AIX systems, you may see an error that a file can't be overwritten because it's associated with a running process. Change the permissions on the file to write (**755**), and you can complete the deployment.
Chapter 5: How to deploy the Business Modeler IDE templates on Teamcenter

Deploy Business Modeler IDE packages

Users can generate a Business Modeler IDE template package in Teamcenter 11.3 or later that can be deployed to Teamcenter environments using either Deployment Center or Teamcenter Environment Manager (TEM). This consolidated output directory contains templates, libraries, and deployment configuration files.

To deploy a Business Modeler IDE template package, obtain the directory of the template package output generated by the Business Modeler IDE. Place the Business Modeler IDE output directory in the software subdirectory of the Deployment Center repository.

To ensure you have a supported template package, check:

- **Directory naming convention**
  
  `template-internal-name_OS_template-version_build-version_YYY_MM_DD_HH-MM-SS`
  
  An optional template version may be assigned by the Business Modeler IDE user to track the versions of a template package. If the Business Modeler IDE user assigns a build number, the template is in development. The build version tracks iterative testing before the template is ready for production. Template versions and build versions are expressed as integers separated by periods, up to four places.

- **artifacts** subdirectory
  
  Contains the template software ZIP files for deployment.

- **dc_contributions** subdirectory
  
  Contains the template bundle information (called packages) for deployment by Deployment Center. If you use TEM, this directory is ignored.

- **tem_contributions** subdirectory
  
  Contains the template bundle information for deployment by TEM. If you use Deployment Center, this directory is ignored.

- **media_teamcenter_template-package-name.xml** file
  
  Provides the application names to both TEM and Deployment Center for deployment.

The Deployment Center repository displays **Dependencies** as specified within Business Modeler IDE packages using package IDs.

For information on creating and updating Business Modeler IDE packages, refer to the Business Modeler IDE documentation included with Teamcenter.
Chapter 6: Microservice Framework for Access Controls and Requirements Management

Running Access Controls and Requirements Management on the Microservice Framework

In order to run Access Controls and Requirements Management in Active Workspace, you must install the Teamcenter Microservice Framework.

With the framework, you can configure multiple replicas of application microservices. The replicas can be deployed across a set of machines for scalability and redundancy.

Install the Microservice Framework

Use Deployment Center to configure Microservice Framework installation and generate microservice node deployment scripts. Microservice Framework cannot be installed using Teamcenter Environment Manager.

Installation sequence

1. Choose the host operating system (OS) for microservice nodes.

   Considerations:

   All microservice nodes within an environment must run on the same OS type.

   Your choice of host OS type for the nodes is independent of whether your Teamcenter web-tier architecture is Java EE or .NET.

   The following table compares the two types of OS supported by Microservice Framework.

<table>
<thead>
<tr>
<th></th>
<th>Linux 64-bit</th>
<th>Windows 64-bit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Level of recommendation by Siemens PLM Software</td>
<td>Recommended</td>
<td>Supported</td>
</tr>
<tr>
<td></td>
<td>Linux 64-bit</td>
<td>Windows 64-bit</td>
</tr>
<tr>
<td>----------------------</td>
<td>-------------------------------------------------------------------------------</td>
<td>----------------</td>
</tr>
<tr>
<td>Third party software</td>
<td>Docker If Microservice Framework is to be installed on a Linux host, then Docker must be installed on the host before Microservice Framework deploy scripts are run on the host.</td>
<td>None</td>
</tr>
<tr>
<td>Management of secondary nodes</td>
<td>Managed by the primary node (Docker Swarm Manager)</td>
<td>Configured in Deployment Center</td>
</tr>
</tbody>
</table>

2. Download a Teamcenter Microservice Framework kit and place it in a Deployment Center software repository.

**Tip**

Download the kit that corresponds to the host OS of the server where you intend to create a microservice node.

If you manage multiple environments, and the microservice node host OS differs between environments, then either kit version can be placed in the Deployment Center software repository.

The repository must also contain Teamcenter Foundation software media.

3. In Deployment Center, open or create an environment and on the **Software** page add **Microservice Framework**.

   Teamcenter **Foundation** must be included in the software list.

4. On the **Applications** page add the applications that you want to install in the environment.

5. On the **Components** page, specify values for the **Microservice Node** options.
For this option | Do this
---|---
**Installation Path** | Enter the path to the Teamcenter installation root folder on the microservice node host machine.

**Machine Name** | Enter the name of the server machine on which to install the node.

**OS** | Identify the OS on the server machine specified in **Machine Name**, either **lnx64** or **wntx64**. All Microservice Nodes must be installed on machines with the same OS. Hybrid OS is not supported.

If you choose **wntx64**, then an option **Install Microservice Manager as a Windows service** appears. Installation as a Windows service is recommended to automate starting the service.

**Microservice Node Type** | Choose one of two node types:

- **Primary**
  - The primary microservice node in the Teamcenter environment. Exactly one primary-type Microservice Node component is required in an environment.

- **Secondary**
  - A secondary microservice node in the Teamcenter environment.
  - You can add secondary-type Microservice Node components as needed.

**Microservice Authentication Password and Confirm Password** | Enter a password to be used for generating the .p12 files that contain keys for signing and validating authentication tokens.

**Note**
- Record and keep secure the password for potential use should you want to open and edit the keys.

**Protocol** | Choose the protocol used for communication with the microservice service dispatcher, either **http** or **https**.

If you choose **https**, then you will need to perform some additional steps to **configure the service dispatcher for https**.

**Port** | Enter the port number for communication with the microservice service dispatcher.
For this option | Do this
---|---
Number of Service Instances to Run | For each of the microservices listed, enter the number of service instances to run on this microservice node.

**Note**
Deployment Center displays options here for the selected applications that employ microservices that depend on the Microservice Framework.

6. Save the component settings.

7. Complete configuration of the environment and deploy the software.

**Note**
Before you run the scripts generated by Deployment Center, ensure that the appropriate OS version of the Microservice Framework kit is available in the location that you identify using the `deploy` command `-softwareLocation` argument.

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

8. (When installed in a Distributed environment type) After completing installation of the primary Microservice Node, copy the private key (signer) to the Webtier deployment directory.

**Note**
This step is not required for a Single Box environment type.

A message similar to the following appears at the end of running a deployment script for a primary microservice node.

**Example**
The files in the signer_config directory are required to sign authentication tokens provided when sending a request to a microservice. This directory should be copied to the Teamcenter web tier host machine prior to executing the web tier deployment script. Please refer to documentation for details on J2EE Webtier deployment. It should also be copied into $TC_DATA of the Teamcenter installation.

J2EE architecture | Copy the signer_config directory to the directory where the JavaEE web tier deployment script is to be run.
.NET architecture | On the host where IIS is installed, before starting IIS, copy the `signer_keystore.p12` to `C:\windows\System32\inetsrv`. 

9. (For installation on Windows) Copy the public keystore (validator) from the primary Microservice Node machine to secondary Microservice Node machines.

**Note**

A message similar to the following appears at the end of running a deployment script for a microservice node.

**Example**

The files in the `[InstallationPath]\microservices\secrets` directory are required by microservices to validate authentication tokens in a request. This directory should be copied to other Microservice Node machines.

10. (When using .NET architecture) On the machine where IIS is running, in the App Pool set **Load User Profile** to True.
Starting the Microservice Manager (Windows only)

If you install a microservice node on a Windows host, then microservice processes are started by a Microservice Manager. Depending on your choice in Deployment Center for the **Install Microservice Manager as a Windows service** option, after you run the deployment scripts the Microservice Manager either can be started as a Windows service, or you must run a startup file to start the Microservice Manager.

**Windows service**

The Microservice Manager appears in the Windows service list as **Microservice Manager PoolA**.

By default, the script generated by Deployment Center installs the service with Startup Type=Automatic, which starts automatically on the next system startup. If the deployment script detects that the service is already installed, then the script does not change the parameters.

**Startup file**

Run the Microservice Manager startup file:

```
[installation directory]\microservice_manager\start_manager.bat
```
Starting the microservice stack (Linux only)

If you install a microservice node on a Linux host, then you must start the microservice Docker stack.

Linux uses Docker for microservice nodes

If you intend to use a Linux host for microservice nodes, then you must install Docker on your Linux host before running the Deployment Center deployment scripts that install the microservice stack.

Installing Docker

- Starting the microservice Docker stack
- Working with Docker containers
- Docker troubleshooting

Installing Docker

For certified versions of Suse, Redhat, CentOS, and Docker software, refer to the Teamcenter Software Certifications section of the hardware and software certifications page on GTAC.

Deploy Docker based on the installation instructions at https://docs.docker.com. Deployment is a two-phase process.

1. Install Docker.

2. Configure Docker to restart on system boot.

Starting the microservice Docker stack

After you run the Deployment Center script that installs the microservice stack, and perform any necessary post-install script installation tasks, you must start the stack.

1. On the microservice node machine, change to the Docker <installation-path>/container directory.

2. Enter the command to start the microservice stack.

   `docker deploy -f tc_microservice_framework.yml <myStackName>`

Working with Docker containers

You can work with Docker containers from the command line and in Portainer.
To view container logs at the command line, use the `docker service logs` command.

**Example**

1. To obtain service identifiers, run
   
   ```bash
docker service ls
   ```

2. To display the last five lines that were output by a particular service, run
   
   ```bash
docker service logs -f --no-task-ids --tail 5 service_id
   ```

   where `service_id` is a service identifier obtained from step 1.

Portainer ([https://portainer.io/](https://portainer.io/)) is an open source product that provides a web-based UI to more 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 adjust scaling of services across the swarm.

**Docker troubleshooting**

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

If when running any Docker command you receive the following error

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

then do the following.

1. To check whether `dockerd` is running, run
   
   ```bash
   ps -ef | grep dockerd
   ```

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

<table>
<thead>
<tr>
<th>If <code>dockerd</code> is</th>
<th>Then do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Not running</td>
<td>Restart Docker, and configure <code>dockerd</code> to restart on the next boot:</td>
</tr>
<tr>
<td></td>
<td><code>sudo systemctl start docker</code></td>
</tr>
<tr>
<td></td>
<td><code>sudo systemctl enable docker</code></td>
</tr>
<tr>
<td>Running</td>
<td>The user is likely not a member of the Docker Linux group.</td>
</tr>
<tr>
<td></td>
<td>Add the user to the group. Ignore any error output from <code>groupadd</code>.</td>
</tr>
<tr>
<td></td>
<td><code>sudo groupadd docker</code></td>
</tr>
<tr>
<td></td>
<td><code>sudo usermod -aG docker $USER</code></td>
</tr>
</tbody>
</table>
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, then add the `-debug` option and run the command again, and then view the log.

```
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.

---

**Configure the Microservice Framework for HTTPS**

**Note**

The server side of an HTTPS connection needs a private key/certificate. A signed private certificate can be obtained from a Certificate Authority (CA), or tools such as openssl can be used to produce a self-signed certificate. A self-signed certificate is potentially less secure than a CA certificate and requires additional configuration steps.

The client side of an HTTPS connection needs a public key/certificate that enables it to verify the private certificate provided by the HTTPS server. The standard Java trust store contains the public keys needed to verify CA-signed certificates. For a self-signed certificate, an alternate trust store containing the required key(s) can be provided.

**Note**

HTTPS is not supported for Microservice Framework on Teamcenter .NET web tier architecture.

---

If when configuring a Microservice Node in Deployment Center you choose the HTTPS protocol for the Microservice Service Dispatcher, then in the extended properties for the Teamcenter Web Tier (Java EE), in the Microservice Communications Settings section you must enable SSL.

1. Select the Teamcenter Web tier (Java EE) component.

2. Enable Show All Parameters.
3. Under **Microservice Communication Settings**, select **Enable SSL**.

When you select the **Enable SSL** checkbox, the **Trust Store, Trust Store Type**, and password property boxes are enabled. If you are planning to use a CA-signed certificate, then you do not need to enter or change the default values; the standard Java trust store will work. However, if you intend to use a self-signed certificate, then you must enter appropriate values for the options.

<table>
<thead>
<tr>
<th>For this option</th>
<th>Do this</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trust Store</td>
<td>Enter the location of a keystore (jks or pkcs12) holding the private server-side key for SSL.</td>
</tr>
<tr>
<td></td>
<td>The key store can be created using any industry-standard tools. It can contain certificates that are either CA-signed or self-signed.</td>
</tr>
<tr>
<td></td>
<td>If the value is empty, then the standard Java trust store is used.</td>
</tr>
<tr>
<td>Trust StoreType</td>
<td>Enter JKS (default) or PKCS12, depending on the type for your trust store.</td>
</tr>
</tbody>
</table>

**Manual step following running of the Deployment Center deploy scripts - Web tier**

Ensure that the trust store used by the Teamcenter web tier contains the certificates required to establish an SSL connection to the Service Dispatcher.

**Manual steps following running of the Deployment Center deploy scripts - Service Dispatcher**

To configure HTTPS between the web tier and the Service Dispatcher when using a self-signed certificate, edit the appropriate file to extend the ARGS value for the service_dispatcher to provide values for protocol, keystore, kpassword and optionally the keystoreType (if not jks).

**When the microservice node is on a Windows host**

Edit the file:

```
\microservice_installation_path\microservices\services_config\service_dispatcher.json
```

**Example**

**Original file content:**

```
{
  "service_dispatcher":{
    "image":"service_dispatcher-1.2.0",
    "environment":{
      "ARGS=-Dport=9090 -Dkspassword=<keystore_password> -Dprotocol=https -Dkeystore=<path_to_keystore_file> -Deureka.serviceUrl.default=http://msnode1:8080/eureka/v2"
    }
  }
}
```

**ARGS setting after editing to include security properties:**

```
ARGS=-Dport=9090 -Dprotocol=https -Dkeystore=<path_to_keystore_file> -Dkspassword=<keystore_password> -Deureka.serviceUrl.default=http://msnode1:8080/eureka/v2
```

**When the microservice node is on a Linux host**

In the Service Dispatcher (Eureka) Docker container, edit the file:

```
installation-path/container/tc_microservice_framework.yml
```
Example

ARGS setting after editing to include security properties:

```
service_dispatcher:
  image: siemens/teamcenter/service_dispatcher:1.2.0
deploy:
  replicas: 1
environment:
  - ARGS=-Dport=9090 -Dprotocol=https -Dkeystore=<path_to_keystore_file>
  - Dkeystore_password=<keystore_password> -Deureka.serviceUrl.default=
    http://eureka:8080/eureka/v2/
```
Siemens Industry 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 PLM Software**

Siemens PLM Software, a business unit of the Siemens Industry Automation Division, is a leading global provider of product lifecycle management (PLM) software and services with 7 million licensed seats and 71,000 customers worldwide. Headquartered in Plano, Texas, Siemens PLM Software works collaboratively with companies to deliver open solutions that help them turn more ideas into successful products. For more information on Siemens PLM Software products and services, visit www.siemens.com/plm.

© 2019 Siemens Product Lifecycle Management Software Inc. Siemens and the Siemens logo are registered trademarks of Siemens AG. D-Cubed, Femap, Geolus, GO PLM, I-deas, Insight, JT, NX, Parasolid, Solid Edge, Teamcenter, Tecnomatix and Velocity Series are trademarks or registered trademarks of Siemens Product Lifecycle Management Software Inc. or its subsidiaries in the United States and in other countries. All other trademarks, registered trademarks or service marks belong to their respective holders.