# TEAMCENTER

# Document Management — Deployment and Administration

Teamcenter 2412

SIEMENS

## About Siemens Digital Industries Software

Siemens Digital Industries Software is a global leader in the growing field of product lifecycle management (PLM), manufacturing operations management (MOM), and electronic design automation (EDA) software, hardware, and services. Siemens works with more than 100,000 customers, leading the digitalization of their planning and manufacturing processes. At Siemens Digital Industries Software, we blur the boundaries between industry domains by integrating the virtual and physical, hardware and software, design and manufacturing worlds. With the rapid pace of innovation, digitalization is no longer tomorrow’s idea. We take what the future promises tomorrow and make it real for our customers today. Where today meets tomorrow. Our culture encourages creativity, welcomes fresh thinking and focuses on growth, so our people, our business, and our customers can achieve their full potential.

Support Center: **support.sw.siemens.com**

Send Feedback on Documentation: **support.sw.siemens.com/doc_feedback_form**

# Contents

# 1. Deploying Document Management

Documents play an important role in the product lifecycle management. Teamcenter Document Management helps ensure that documents conform to your required formats, meet product deadlines, and are synchronized with the latest product information.

Authors can perform various tasks including authoring documents, reviewing and marking up files, printing files with the applicable distribution and print statements, and digitally signing PDFs. Each task requires certain additional PLM components such as Dispatcher, BMIDE, and Visualization Convert and Print in the deployment setup.

Additionally, authors use different applications such as Microsoft Office, Adobe Document Cloud, Adobe Creative Cloud, and Google Online. To edit files, you must integrate Teamcenter with the relevant applications.


To enable Teamcenter Document Management to meet your business and process requirements, you must perform certain configurations including setting up document templates and define naming conventions, configuring markups, setting up document rendering, and synchronizing attributes between Teamcenter and other applications.

### Where do I go from here?

| | |
|---|---|
| Business User | See *Document Management on Rich Client — Usage* and *Document Management on Active Workspace — Usage*. |
| Administrator | |

| | |
|---|---|
| Integrate Teamcenter with different applications | Teamcenter can be integrated with Microsoft Office (desktop and online versions), Google Online, Adobe Document Cloud, and Adobe Creative Cloud.<br><br>See **Integrating Teamcenter with Microsoft Office**, **Integrating Teamcenter with Adobe applications**, and **Setting up Google Online with Teamcenter**. |
| Synchronize attributes between Teamcenter and other applications | Attributes can be synchronized between Teamcenter and Microsoft Office. They can also be **synchronized between Teamcenter to Adobe PDF file and to other applications**. |
| Configure markups | You can set up applications and **specify markup information for them in Teamcenter rich client**. You can also enable and specify application settings for markups. Additionally, you can provide access privileges for marking up documents. |
| Configure digital signatures | To enable users open PDF files from Active Workspace and add digital signatures, you must first **configure digital signing** and then install the relevant applications on the client machines. You can also specify applications that may be launched from Microsoft Outlook for the user to sign documents digitally. |
| Setting up stamps | You can set up stamps to be added to printed and rendered documents. View information **about setting up stamps**. |
| Configure Document Management on Active Workspace | See **Setting up the Active Workspace for Document Management tasks**. |

# 2. Planning the Document Management deployment

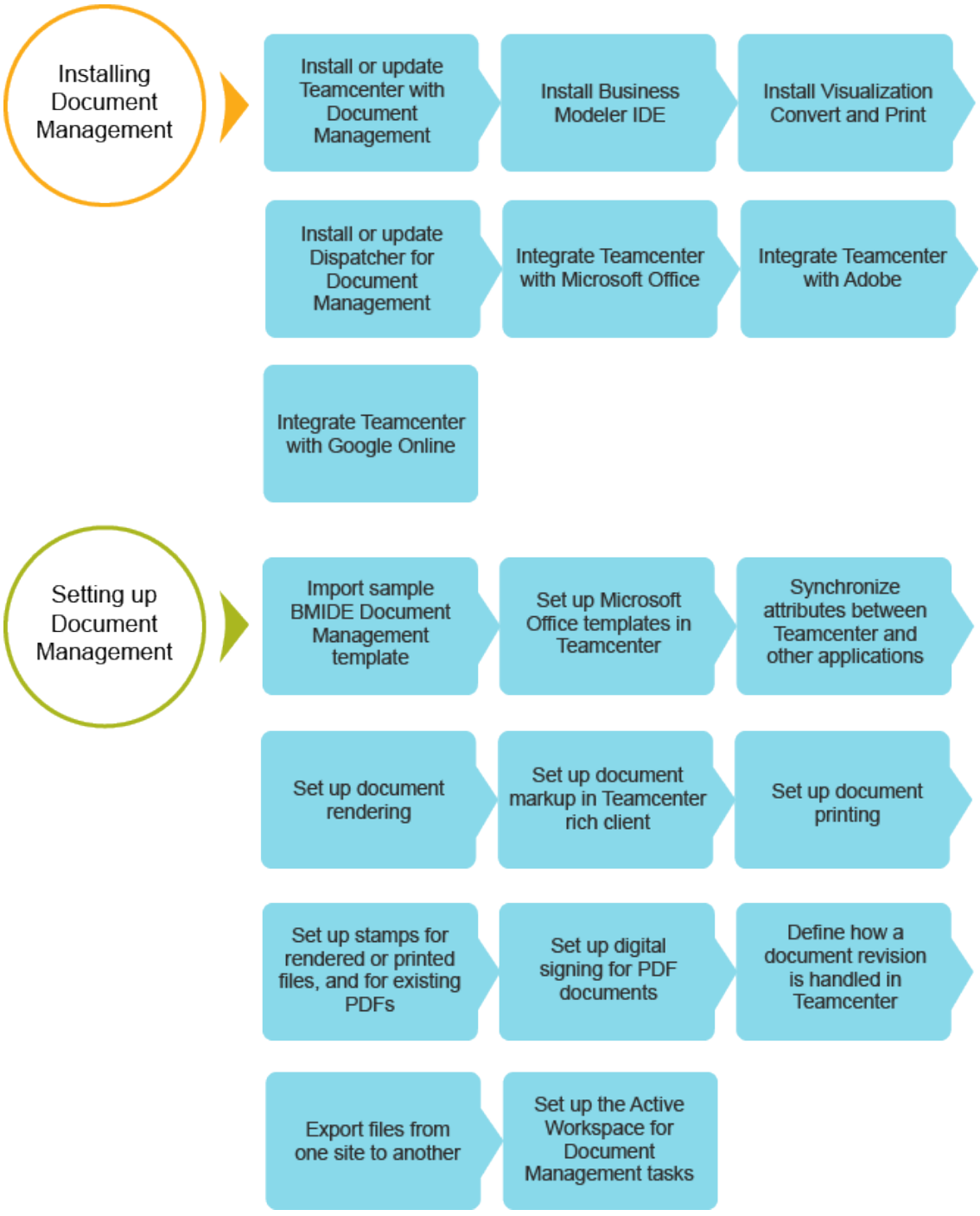
As per your site requirements, you must plan to install appropriate components on the server and client machines in your Teamcenter set up.

| Site Requirements | Server machines | Client machines |
|---|---|---|
| Use Microsoft Office integrated with Teamcenter to create documents (based on templates) and to mark up documents during review | | Microsoft Office |
| Use Microsoft Office Online set within Active Workspace to view and edit documents. | Microsoft Office Online<br><br>Teamcenter Office Online | |
| Use **Google Online set within Active Workspace** to view and edit documents. | Teamcenter Google Online | |
| Use **Acrobat Reader and Adobe Acrobat integrated with Teamcenter** to mark up documents during review | | Acrobat Reader<br><br>Adobe Acrobat |
| Use **Adobe Creative Cloud applications integrated with Teamcenter** to manage artwork | | Illustrator<br><br>Photoshop<br><br>InDesign |
| **Render documents** to different file formats | Teamcenter Lifecycle Visualization Convert and Print<br><br>Dispatcher with RenderMgtTranslator and PreviewService translators | |
| **Print documents** | Teamcenter Lifecycle Visualization Convert and Print<br><br>Dispatcher with BatchPrint and | |

| Site Requirements | Server machines | Client machines |
| --- | --- | --- |
| | PreviewService translators | |
| Digitally sign PDFs | | The Active Workspace Launcher<br><br>Document Management Teamcenter Client PDF Sign |

After you install the required components, you must configure your deployment set up so that the end users can use Teamcenter to manage documents. You perform these configurations within a template in BMIDE. To help you with the configuration process, a sample BMIDE Document Management template is provided by default. To use this sample template, you must first **install it**. Next, you can **import this sample template** into BMIDE to create your own configurations.

# 3. Document Management deployment workflow



# 4. Installing Document Management on your Teamcenter environment

## Install Document Management using Deployment Center

Add the Document Management application to your existing Teamcenter environment.

### Prerequisites

- Microservice framework is installed.
- The Teamcenter environment is a distributed one so that some applications can be installed on multiple machines and the users can perform the tasks (including digitally signing and adding markups using Adobe Acrobat) from those machines.

### Procedure

1. Log on to Deployment Center and select the environment to which you want to add Document Management.
2. Go to the **Applications** tab. Click **Add or Remove Selected Applications**.
3. In the **Available Applications** panel, select the following applications, and then click **Update Selected Applications**.

   Deployment Center automatically selects any additional dependent applications. You can also use the web browser search to find these applications.

   - **Teamcenter > Teamcenter > Content and Document Management > Teamcenter Client Acrobat Launcher**

     When you select **Teamcenter Client Acrobat Launcher**, the **Acrobat/Reader Plugin** application is also selected automatically.

   - **Teamcenter > Teamcenter > Content and Document Management > Document Management Client Application**
   - **Teamcenter > Teamcenter > Document Management**
   - **Teamcenter > Teamcenter > Enterprise Knowledge Foundation > Render Document for Rich Client**
   - **Teamcenter > Teamcenter > Enterprise Knowledge Foundation > Dispatcher Client for Rich Client**

- **Teamcenter > Teamcenter > PLM Base > Sample files**
- **Teamcenter > Teamcenter > Sample Document Management**

4. Go to the **Components** tab.
5. In the **Selected Components** list, select the following components and then enter the values for their configuration parameters.
   - Click **Start** for **Sample Files** component, and then enter a machine name and OS.
   - Click **Start** for **Document Management Teamcenter Client PDF Sign** component and then select the following configuration parameters:
     - Click **Enable Mass Client Deploy?** so that the application can be installed on users machines and they can open the PDF files from Active Workspace to add digital signatures.
     - Enter a value for the **Instance Name** box.
     - Click the required operating system from the **OS** list.

   You can also install **Document Management Teamcenter Client PDF Sign** component **using a ZIP file on multiple machines**.

   When you finish entering values for each component, click **Save Component Settings**.
6. In the **Selected Components** list, note any remaining components whose configuration status is not **100%**. Select each incomplete component, enter required parameters, and save component settings until all components in the environment show a configuration status of **100%**.

   When all components are fully configured, the **Deploy** tab is enabled.
7. Go to the **Deploy** tab. Click **Generate Install Scripts** to generate deployment scripts you will use to update affected machines.

   When script generation is complete, note any special instructions in the **Deploy Instructions** panel.
8. Locate deployment scripts, copy each script to its target machine, and then run each script on its target machine.

   For more information about running deployment scripts, see *Deployment Center — Usage*.

# Install Document Management using Teamcenter Environment Manager (TEM)

Add the Document Management application to your existing Teamcenter environment.

1. Run Teamcenter Environment Manager (TEM) from your *TC_ROOT\install* directory.
   The *TC_ROOT* directory is the folder containing your Teamcenter installation, for example, *C:\app\tc*.

2. In the **Maintenance** panel, select **Configuration Manager** and click **Next**.

3. In the **Configuration Maintenance** panel, select **Perform maintenance on an existing configuration** and click **Next**.

4. In the **Old Configuration** panel, select an existing configuration on which you want to install Document Management and click **Next**.

5. In the **Feature Maintenance** panel, select **Add/Remove Features** and click **Next**.

6. In the **Features** panel:

   - Select **Server Enhancements→Server Manager** (available only for a 4-tier rich client installation), **Sample Files**, and **Teamcenter Security Service**.

   - Select **Extensions→Sample Document Management**.

   - Select **Extensions→Enterprise Knowledge Foundation→Teamcenter Client for Microsoft Office**, **Dispatcher Client for Rich Client**, and **Render Document for Rich Client**.

7. Enter information as needed in the subsequent panels.

8. In the **Multiplexing Proxy (MUX)** panel, click **Advanced**.

   In the **SSL Configuration** dialog box, select **Enable SSL** to ensure that the Office Online Server runs in the Secure Socket Layer (SSL) mode. Next, enter the keystore and truststore details, and click **OK**.

9. Enter information as needed in the subsequent panels.

10. In the **Security Services** panel, you configure Security Services for Teamcenter, which provides single sign-on authentication. Once you enable these services, the users need not enter logon credentials each time they launch a different Teamcenter product during a single session.

    Select **Enable Security Services** and enter the following information:

| | |
|---|---|
| **Login URL** | Specifies the complete URL of the Security Services Login Service Web application. |
| **Service URL** | Specifies the complete URL of the Security Identity Service Web application. |
| **Application ID** | Specifies the application ID of this instance of Teamcenter in the Security Services application registry. |

If you use Security Services in the Teamcenter web tier application, you must configure Security Services using Web Application Management as described in the appropriate server installation guide.

11. Click **Next** and enter information, as needed, in the subsequent panels.

12. In the **Confirmation** panel, click **Start**.

# 5. Installing Business Modeler IDE

You must install Business Modeler IDE to configure Document Management. In BMIDE, you can either **import the sample BMIDE Document Management template** (**sdm0sampledocmgt**) or create a new *dependent* template from the sample template.

You can install Business Modeler IDE as a standalone application or install it within your existing Eclipse (Java IDE) environment.

> Caution:
>
> Install Business Modeler IDE on a machine separate from the one containing your corporate server.

# 6. Installing Visualization Convert and Print

For rendering documents to different file formats, to generate thumbnails for documents, and to batch print documents, you must install Visualization Convert and Print. Visualization Convert and Print is a part of the Lifecycle Visualization installation. During the installation, ensure that you select the **Convert & Print** option.

For more information about installation, see Lifecycle Visualization Installation. For more information about using Convert and Print, see *Teamcenter Convert and Print*.

# 7. Installing or updating Dispatcher for Document Management

You require Dispatcher to render documents in different file formats, to generate thumbnails, and for batch printing by using the appropriate conversion and print translators.

### Prerequisites

Dispatcher is already installed in the Teamcenter environment.

### Procedure

1. Enable the following print and conversion translators.

   - **Document Management (DocMgt) Translators→RenderMgtTranslator** and **BatchPrint**
   - **TcVis Translators→PreviewService**

2. Modify the Lifecycle Visualization vvcp.ini file with the following settings:

   - Set **ExitOnError=on** in the **[Prepare]** section for rendering the documents and in the **[PVPrint]** section for batch printing.
   - Set **OfficeAutomation=on** in the **[Converter]** section for rendering the documents and batch printing.
   - Set **StampFormat=pdf** in the **[Prepare]** section for rendering the documents and in the **[PVPrint]** section for batch printing.

   > Note:
   >
   > Not all MDS commands are supported for PDF stamping mode (*StampFormat=pdf*). Thus, set the MDS command appropriately in the MDS template file. Restart the Dispatcher Client if the DMTemplates's dataset is modified or updated.

# 8. Integrating Teamcenter with Microsoft Office

You must integrate Teamcenter with Microsoft Office so that users can edit and view files (saved as datasets in Teamcenter) in different Microsoft Office applications. Depending upon your site requirements, they can edit and view files:

- In the desktop version of Microsoft Office installed on their computers.

  For this, you must install and configure Teamcenter Client for Microsoft Office.

- Within Active Workspace by leveraging the Microsoft Office Online features.

  For this, you must set up Microsoft Office Online within Teamcenter.

Note:

- If the users in your organization need a full Microsoft Office editing experience, install Teamcenter Client for Microsoft Office. This application works with the Microsoft Office application installed on the users' clients. This integration supports the most document management features.
- If users need to edit in the browser, or in the Active Workspace client, install Teamcenter Office Online and its prerequisite, Microsoft Office Online Server. This version of Microsoft Office contains limited editing features of Microsoft Office on the users' client. Some document management features are not available.
- When the users need to synchronize attributes between Teamcenter and Microsoft Office files (ExcelX, PowerpointX, and MSWordX), as an administrator, you can configure **logical objects**. You can also synchronize attributes by using **Client for Office**; however, the two approaches for synchronizing attributes cannot be used together.

# 9. Integrating Teamcenter with Adobe applications

## About integrating Teamcenter with Adobe applications

You can integrate Teamcenter with Adobe Acrobat Pro DC, Adobe Acrobat Standard DC, Acrobat Reader DC, and Adobe Creative Cloud (CC) applications such as Illustrator, Photoshop, and InDesign. By doing so, end users of these applications can perform Teamcenter tasks from within the applications, without the need to log on to Teamcenter to do so.

To integrate Teamcenter with Adobe Acrobat Pro DC, Adobe Acrobat Standard DC, and Acrobat Reader DC, you must **install Teamcenter Acrobat/Reader Plugin** on the client machines.

To integrate Teamcenter with Adobe CC applications, you must **set up Adobe CC Integration** on the client machines. This integration hosts Active Workspace within the Adobe CC applications.

## Integrate Teamcenter with Adobe Acrobat Pro DC, Adobe Acrobat Standard DC, and Adobe Acrobat Reader DC

1. Run Teamcenter Environment Manager (TEM) from your *TC_ROOT\install* directory.
   The *TC_ROOT* directory is the folder containing your Teamcenter installation, for example, *C:\app\tc*.

2. In the **Maintenance** panel, select **Configuration Manager** and click **Next**.

3. In the **Configuration Maintenance** panel, select **Perform maintenance on an existing configuration** and click **Next**.

4. In the **Old Configuration** panel, select an existing configuration on which you want to install Document Management and click **Next**.

5. In the **Feature Maintenance** panel, select **Add/Remove Features** and click **Next**.

6. In the **Features** panel:

   - Select **Extensions→Content and Document Management→Acrobat/Reader Plugin**.
     Ensure that you have installed Adobe Acrobat Pro/Standard/Reader Document Cloud applications.

7. Click **Next** and enter information, as needed, in the subsequent panels.

8. In the **Confirmation** panel, click **Start**.

# Set up Active Workspace within Adobe Creative Cloud applications

Active Workspace can be hosted within Adobe Creative Cloud (Adobe CC) applications, such as Illustrator, Photoshop, and InDesign. To do so:

- **Install Teamcenter Adobe CC Integration for Active Workspace**.
- **Specify the URL for accessing Active Workspace from the Adobe applications**.
- Copy the plugin to launch the Adobe files from Active Workspace to appropriate Adobe CC applications:

  Copy *\additional_applications\adobe\Win64\Launch.aip* to *C:\Program Files\Adobe\Adobe Illustrator CC 2019\Plug-ins\Tools*.

If you wish to discontinue hosting Active Workspace within Adobe Creative Cloud applications, **remove Teamcenter Adobe CC Integration**.

## Install Teamcenter Adobe CC Integration

Use one of the following methods for integrating Teamcenter with Adobe Creative Cloud applications:

- Adobe ExMan command line tool method

  1. Ensure that Active Workspace is installed. For more information on this, refer to *Active Workspace Deployment* in the Active Workspace documentation.
  2. Download the Adobe ExMan Command Line Tool ZIP file from **https://partners.adobe.com/exchangeprogram/creativecloud/support/exman-com-line-tool.html**. Download the version for Windows. For more information on Adobe's Extension Manager command line utility and the available commands, see **https://helpx.adobe.com/extension-manager/using/command-line.html**.
  3. Extract the contents of the *ExManCmd_win.zip* file into any folder of your choice, for example, **ExMan_root**.
  4. In the software distribution image of your Teamcenter version, locate the folder where you downloaded the Active Workspace distribution package and navigate to: *additional_applications\adobe*.

     In the *adobe* folder, verify that the following files exist:

     - *AWIntegration.zxp* (this is the Active Workspace Adobe extension file)
     - *config.xml* (this is the Active Workspace Adobe configuration file that can be configured for automated deployments)

5. Copy the *AWIntegration.zxp* file into the **ExMan_root** folder so that this is in the same location as the **ExManCmd.exe** utility.

6. Close all active Adobe applications that are compatible with the extension.

7. If the previously installed version of Teamcenter Adobe CC Integration exists, **remove Teamcenter Adobe CC Integration**.

8. Open the command prompt and enter the following command:
   `ExManCmd /install AWIntegration.zxp`

- Adobe Unified Plugin Installer Agent method:

1. Ensure that Active Workspace and Adobe Creative Cloud applications are installed.

   For more information on Active Workspace installation, see *Active Workspace Deployment* in the Active Workspace documentation.

2. In the software distribution image of your Teamcenter version, locate the folder where you downloaded the Active Workspace distribution package and navigate to: *additional_applications\adobe*.

   In the *adobe* folder, verify that the following files exist:

   - *AWIntegration.zxp* (this is the Active Workspace Adobe extension file)

   - *config.xml* (this is the Active Workspace Adobe configuration file that can be configured for automated deployments)

3. Close all active Adobe applications that are compatible with the extension.

4. Open the command line and navigate to the Unified Plugin Installer Agent application folder.

   The Unified Plugin Installer Agent is automatically installed with the Adobe Creative Cloud desktop app version 5.5 and above. For example, *C:\Program Files\Common Files\Adobe\Adobe Desktop Common\RemoteComponents\UPI\UnifiedPluginInstallerAgent*.

   For more information on Unified Plugin Installer Agent command-line basics, see **https://helpx.adobe.com/creative-cloud/help/working-from-the-command-line.html**.

5. On the command prompt, enter `UnifiedPluginInstallerAgent.exe`.

6. On the command prompt, enter the following command: `./UnifiedPluginInstallerAgent /<path of Active Workspace distribution package>/AWIntegration.zxp –install`

For example, if the Active Workspace distribution package is downloaded at C:\apps\tc\tc13\TR\install\additional_applications\adobe, enter the following command:

```
./UnifiedPluginInstallerAgent /C:/apps/tc/tc13/TR/install/
additional_applications/adobe/AWIntegration.zxp –install
```

### Specify the URL for accessing Active Workspace from Adobe applications

The steps below are optional. If the *config.xml* file is not present in the correct location, the user can set the values while executing the Adobe extension for the first time. However, as an administrator, if you wish to automate the deployment of the Adobe extension, you can perform the following steps to pre-configure the settings in the *config.xml* file for the user and to deploy the file in the proper location:

1. In the software distribution image of your Teamcenter version, locate the folder where you downloaded the Active Workspace distribution package and navigate to: *additional_applications\adobe*.

2. Copy the *config.xml* file into:

   *%APPDATA%\Siemens\Teamcenter\AdobeExtension*

3. Open the **config.xml** file and modify the **<ActiveWorkspaceUrl>** tag to specify Active Workspace URL. Ensure that **?ah=true** is appended at the end of the URL.

   Example:

   ```
   <ActiveWorkspaceUrl>http://host:port/awc/
   ?ah=true</ActiveWorkspaceUrl>
   ```

4. (Optional) Modify the **<WorkingDirectory>** tag to specify the directory to use when downloading or uploading files.

   Example:

   For Windows:

   ```
   <WorkingDirectory>c:\temp</WorkingDirectory>
   ```

5. Save and close the **config.xml** file.

### Remove Teamcenter Adobe CC Integration

1. For Windows, open a command prompt.

2. Change the directory to the location where you saved the **ExManCmd** utility, for example, **ExMan_root**.

3. To verify the name of the Teamcenter extension that is already installed, enter the following command:
   `ExManCmd /list all`

   This command lists all installed extensions.

4. Identify and note the Teamcenter extension that you want to remove.

5. To uninstall the Teamcenter extension, enter the following command:
   `ExManCmd /remove "<Teamcenter_Extension>"`

   For example, if the Teamcenter extension that you identified in the previous step is Teamcenter5, then enter the following command:
   `ExManCmd /remove "Teamcenter5"`

   > Note:
   >
   > If you encounter an error while uninstalling the Teamcenter extension, ensure that you are using the latest version of the ExManCmd utility obtained from Adobe.

## Configure single sign-on for Teamcenter within Adobe applications

You can configure single sign-on (SSO) for Teamcenter within Adobe applications if your organization has implemented it. You can configure SSO when you are configuring the Login and Identity Services for your site using Teamcenter Web Application Manager.

### Procedure

1. Clear the value of the **tcsso.frame_ancestors** context parameter.

   > Caution:
   >
   > Do not click **None**. The parameter value should be empty. The **tcsso.frame_ancestors** context parameter is present in the Login Service context parameter worksheet.

   A new *.war* file is generated.

2. Redeploy the newly generated *.war* file.

## Authenticate Adobe clients if Teamcenter uses IIS

If a Teamcenter server integrating with Adobe uses Internet Information Services (IIS) for its web tier, perform the following steps to authenticate the Adobe clients.

### Procedure

1. In the web tier's *web.config* file, modify `<sessionState cookieless="false" mode="InProc" timeout="1440"/>` to `<sessionState cookieless="false" mode="InProc" timeout="1440" cookieSameSite="None"/>`.

2. Save the *web.config* file.

3. Reset IIS.

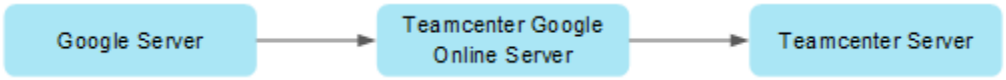
# 10. Setting up Google Online within Teamcenter

## About setting up Google Online within Teamcenter

Using the Google Online feature, users can edit and view documents within Active Workspace. Use Document Management when your organization is already using Google Suite or Google Workspace for Docs, Sheets and Slides authoring. You require an Internet access to connect to Google Drive.

> Note:
>
> Not all Document Management features are supported in Google Online. Scenarios that involve using attribute exchange, rendering to PDF, and cover page management must be thoroughly tested to ensure Google Online supports your business requirements.

### Components required for setting up Google Online

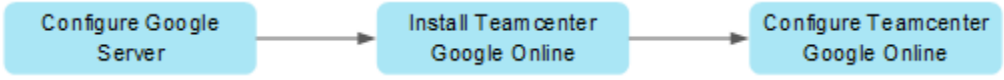



To access Google Viewer within Active Workspace, you required internet access to the Google server. Active Workspace connects to both the Google server and the Teamcenter server through the Teamcenter Google Online server.

When a user selects a file to view it in Active Workspace viewer:

1. Active Workspace connects to the Teamcenter Google Online server to get the launch URL.
2. Teamcenter Google Online server moves the document from Teamcenter to Google Drive for viewing or editing.
3. Active Workspace connects to the Google server with the launch URL to obtain the file to be displayed in its viewer.

### Process flow to set up Google Online within Teamcenter




After setting up Google Online, you must **set up templates for the Google documents** to ensure that a consistent style is followed in all documents.

## Configure the Google server

To access Google Docs, Google Sheets, and Google Slides, you must:

- Configure your Active Workspace URL on the Google Web project administration page at **https://console.cloud.google.com/apis/credentials**.
- Select your web application on the page and add your Active Workspace base URI in the **Authorized redirect URIs** and **Authorized JavaScript Origins** sections. You must use a fully qualified name in these sections.

## Install Teamcenter Google Online through Deployment Center

Add the Teamcenter Google Online application to your existing Teamcenter environment.

### Prerequisites

- Microservice framework are installed.
- Internet access so that Teamcenter Google Online can access the Google server.

### Procedure

1. Log on to Deployment Center and select the environment to which you want to add Teamcenter Google Online.
2. Go to the **Applications** tab. Click **Add or Remove Selected Applications**.
3. In the **Available Applications** panel, select the **Teamcenter Google Viewer** application, and then click **Update Selected Applications**.
4. Go to the **Components** tab.
5. In the **Selected Components** list, perform the following tasks:
   - Click the percent link available for the **Corporate Server** component, and then enter the values for its configuration parameters in the **Teamcenter Google Online Settings** section.

| Field | Action |
|---|---|
| **Use Sponsored Authentication** | Click **True** if you want to use the sponsored authentication and then enter the credentials of the Teamcenter *Sponsored Authentication* user. Otherwise, click **False**. |

| Field | Action |
| --- | --- |
| | Note:<br>For a sponsor, do not use a user account that is typically created for administrative purposes, such as infodba, dba, administrator, or dcproxy. Instead, you must always create a new **Sponsor User ID** and **Sponsor User Password** that is specific for usage with sponsored authentication for Teamcenter Office Online.<br>Moreover, for security reasons, Siemens Digital Industries Software does not recommend the usage of IDs that can be easily guessed such as administrator, infodba, dba, or dcproxy. |

- Click the **Microservice Node** and specify the following information under **Teamcenter Google Online Service**:

| Field | Action |
| --- | --- |
| **Google Application Name** | Enter the name of your Google web application. |
| **Google Client ID** | Enter and then confirm the client ID of your Google web application. |
| **Google Client Secret** | Enter the client secret password of your Google web application. |
| **Protocol** | Click **HTTP** or **HTTPS** considering your security settings.<br>If you click **HTTPS**, then enter the certificate path and credentials. |

6. After you finish entering values for each component, click **Save Component Settings**.

7. In the **Selected Components** list, note any remaining components whose configuration status is not **100%**. Select each incomplete component, enter required parameters, and save component settings until all components in the environment show a configuration status of **100%**.

   When all components are fully configured, the **Deploy** tab is enabled.

8. Go to the **Deploy** tab. Click **Generate Install Scripts** to generate deployment scripts you will use to update affected machines.

   When script generation is complete, note any special instructions in the **Deploy Instructions** panel.

9. Locate deployment scripts, copy each script to its target machine, and then run each script on its target machine.

   For more information about running deployment scripts, see *Deployment Center — Usage*.

## Install Teamcenter Google Online through TEM

Add the Teamcenter Google Online application to your existing Teamcenter environment.

### Prerequisites

- Microservice framework are installed.
- Internet access so that Teamcenter Google Online can access the Google server.

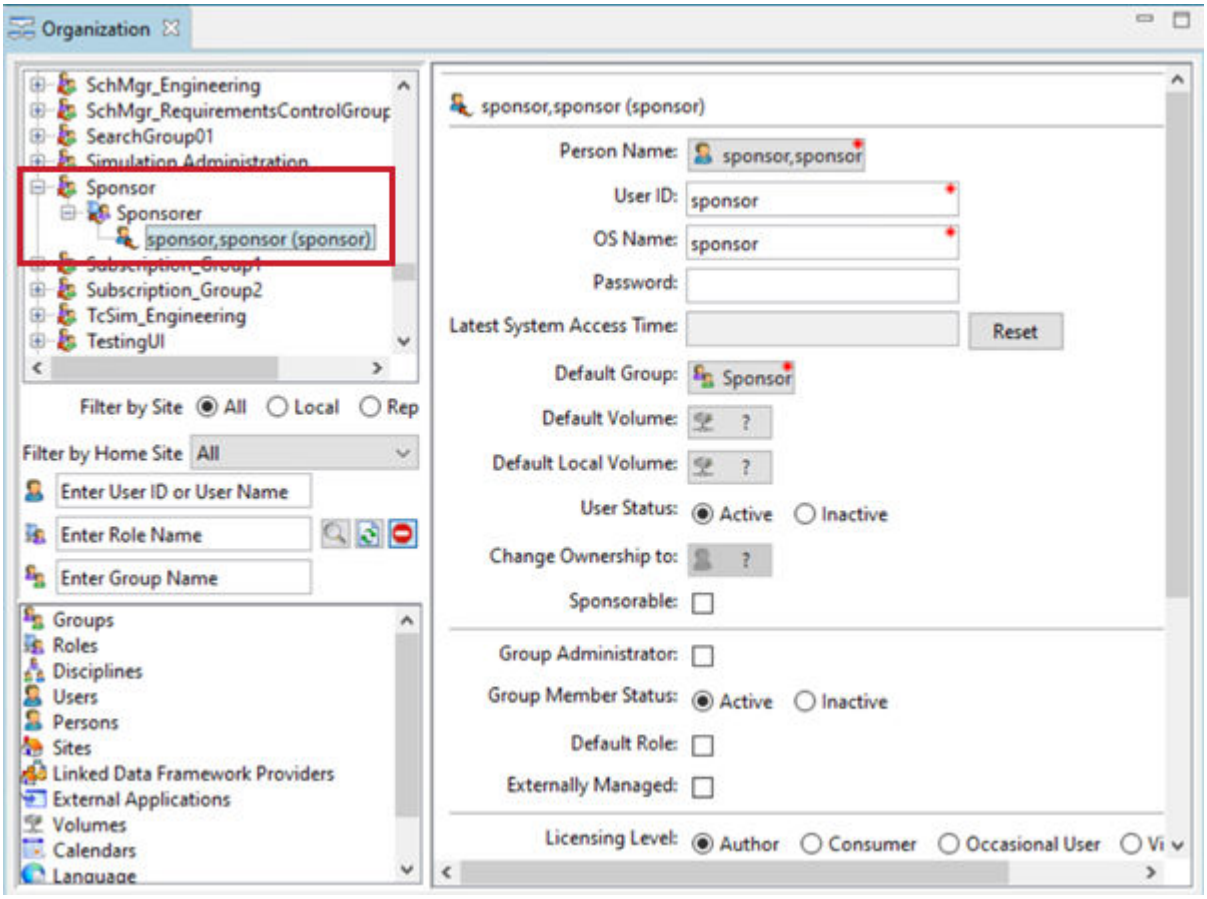
### Procedure

1. Launch TEM.
2. In the **Features** panel, select the following options:
   - **Microservices→Teamcenter Google Online Microservice**
   - **Extensions→Google Viewer Integration**
   - **Active Workspace→Client→Google Online Viewer Client**
   - **Server Extensions→Google Online Viewer**
3. In the **Teamcenter Google Online Microservice** panel, enter the following information:

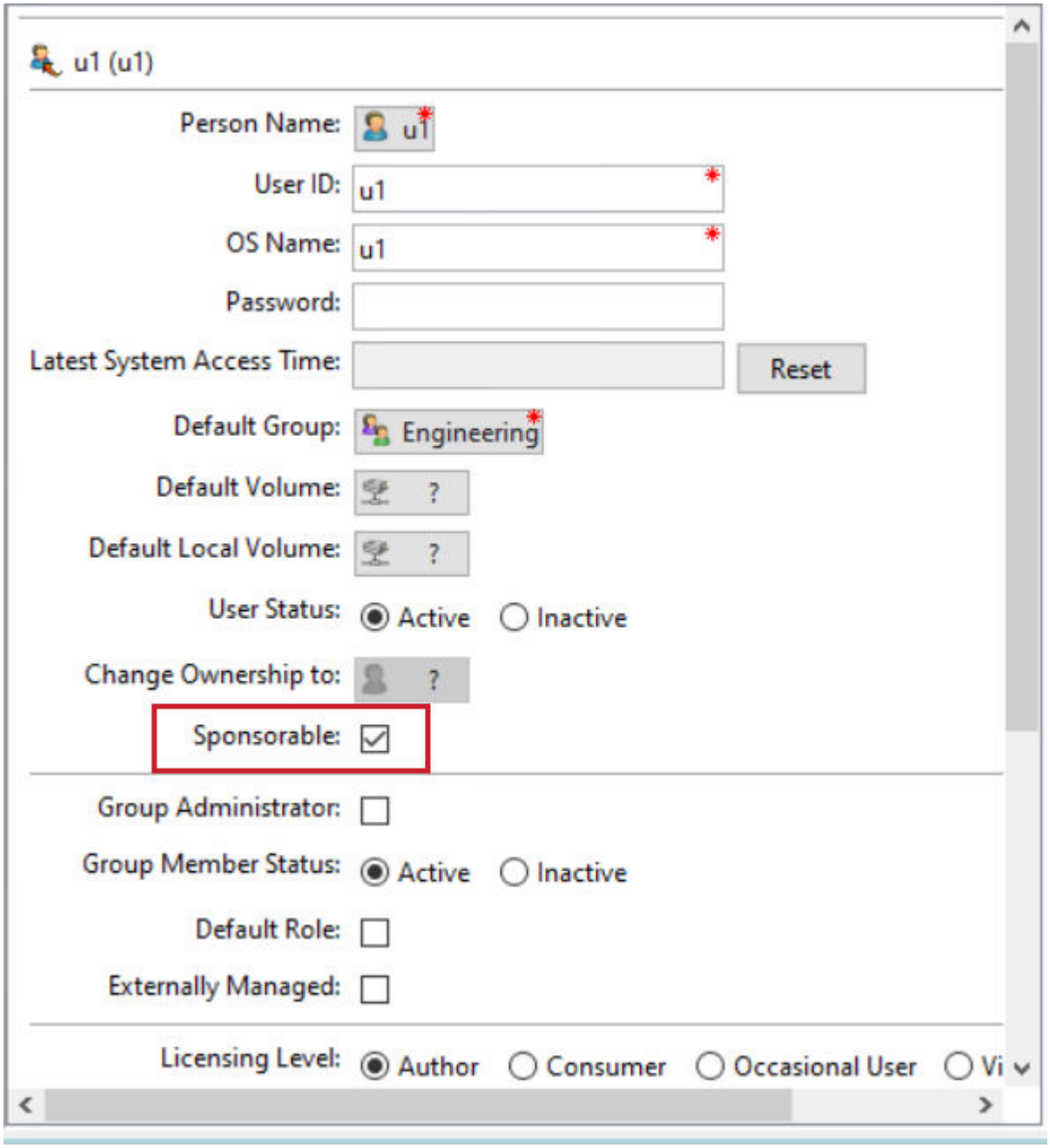
| Field | Action |
|---|---|
| **Google Application Name** | Enter the name of your Google web application. |
| **Google Client ID** | Enter the client ID of your Google web application. |
| **Google Client Secret** | Enter the client secret password of your Google web application. |

4. Click **Next** until you reach the **Confirmation** panel. Click **Start** to begin installing Teamcenter Google Online.
5. If you want to use the sponsored authentication user in Teamcenter, you must manually create this privileged user in the Teamcenter Organization application under the **Sponsor** group

and **Sponsorer** role. If you install Teamcenter Google Online through Deployment Center, the *Sponsored Authentication* user is automatically created.




For users to view and edit files in Active Workspace, ensure that each user is *sponsorable* by enabling the sponsorable flag.




# Configuring Teamcenter Google Online

## About configuring Teamcenter Google Online

The Teamcenter Google Online is configured automatically when you deployed it. To make any changes later, you can use Deployment Center or Teamcenter Environment Manager. Alternatively, manually change the configurations as described below:

- **Update the sponsored authentication user credentials**.

- **Update the Google application information**.
- **Update the Gateway and Service Dispatcher URLs**.
- **Update the web.config file of Teamcenter web-tier if .NET (IIS) is used for deployment**.

Further, if you have set up Secure Socket Layer and single sign-on in your environment, you can optionally perform the following:

- **Update the Secure Socket Layer (SSL) protocol**.
- **Enable single sign-on**.

## Update the sponsored authentication user credentials

If you selected to use the sponsored authentication and then created the user while installing Teamcenter Office Online and Teamcenter Google Online, use the following information to update its credentials.

You can update these credentials in the `TcSecurity` section of the *appsettings.Security.json* file. However, as this section is encoded by default, you must first decode it by running the following command in the command prompt window:

```
JsonSettingsConfigUtility -decode JsonSettingsConfig.json
```

Next, update the credentials:

```
"TcSecurity":{
   "Sponsorer_Name": "Sponsor1",
   "Sponsorer_Password": "default*PW!shouldBeChanged",
}
```

After changing the credentials, you must encode the `TcSecurity` section by running the following command in the command prompt window:

```
JsonSettingsConfigUtility -encode JsonSettingsConfig.json
```

### Guidelines to update the secret of the running tcooweb docker swarm service

The *appsettings.Security.json* file is copied to each running container as a secret at startup. To update the secret of a running tcooweb docker swarm service, use the following guidelines:

- To get the name of the running tcooweb service:

  ```
  docker service ls
  ```

- To get a list of current secrets:

```
docker secret ls
```

- Create a new secret:

```
docker secret create <new secret name> <path to and name of updated secret
file>
```

- Update the service to use the new secret:

```
docker service update --secret-rm <current secret name> --secret-
add source=<new secret name>,target=/run/secrets/appsettings.Security.json
<service name>
```

Example:

Suppose that the name of the existing secret is `tcooweb-appsettings.Security.json`, the name of the service is `myStack_tcooweb_service`, and the folder where the updated *appsettings.Security.json* file is located at is `tcooweb_service-2.0.0`. In such a case, use the following commands:

```
docker secret create tcooweb1-appsettings.Security.json
$TC_ROOT/microservices/tcooweb_service-2.0.0/
appsettings.Security.json
docker service update --secret-
rm tcooweb-appsettings.Security.json
--secret-add source=tcooweb1-appsettings.Security.json,target=/run/
secrets/appsettings.Security.json myStack_tcooweb_service

successful output:
overall progress: 1 out of 1 tasks

1/1: running
==================================================>]

verify: Service converged
```

Note:

You must update the same user information in the Teamcenter Organization application.

## Update the Google application information

You update the Google application information in the `GoogleConfig` section of the *appsettings.Security.ison* file. However, as this section is encoded by default, you must first decode it by running the following command in the command prompt window:

```
JsonSettingsConfigUtility -decode JsonSettingsConfig.json
```

Next, update the following section:

```
"GoogleConfig":{
   "applicationName": "@CHANGE_ME_APP_NAME_INFORMOF_[COMPANY-ID]-[APP-
NAM]-[APP-VERSION]@",
   "client_id": "@CHANGE_ME_CLIENT_ID@",
   "client_secret": "@CHANGE_ME_CLIENT_SECRET@",
}
```

After updating the `GoogleConfig` section, you must encode it by running the following command in the command prompt window:

```
JsonSettingsConfigUtility -encode JsonSettingsConfig.json
```

## Update the Gateway and Service Dispatcher URLs

You can update the Gateway and Service Dispatcher URLs in the `TcConnection` section of the *appsettings.json* file. This file is located at:

- **Windows**

  *TC_ROOT\microservices\tcgoogleweb-version\TcGoogleWeb*

- **Linux**

  *TC_Root/microservices/tcgoogleweb-version*

  The *appsettings.json* file is copied to each running container at startup. Therefore, when the file is updated, restart the containers for the changes to take effect.

In the `TcConnection` section, you must change the Gateway URLs (**GatewayUrls**) and the Service Dispatcher URLs (**ServiceDispatcherUrls**), as necessary. The Gateway and Service Dispatcher URLs can be a list of comma-separated or semicolon-separated URLs.

## Update the web.config file for Teamcenter web tier

As IIS blocks certain characters by default, Google Online users cannot save the changes to their Google documents. Therefore, if the Teamcenter web tier uses .NET (IIS) for deploying Google Online, update the *web.config* file of the web tier so that Google Online users can save the changes to their Google documents.

### Procedure

1. Open the *web.config* file of the .NET web tier in an editor.

> Tip:
>
> You might want to use Notepad to edit the *web.config* file.

2. Search for the `httpRuntime` statement in the *web.config* file and then update the statement with the following information:

   ```
   requestPathInvalidCharacters=">,*,%,\"
   ```

   > Example:
   >
   > *Before:*
   >
   > ```
   > <httpRuntime enableVersionHeader="false" executionTimeout="7200"
   > maxRequestLength="131072" />
   > ```
   >
   > *After:*
   >
   > ```
   >  <httpRuntime enableVersionHeader="false"
   > executionTimeout="7200" maxRequestLength="131072"
   > requestPathInvalidCharacters=">,*,%,\" />
   > ```

3. Save and close the *web.config* file.

4. Reset IIS.

## Update the Secure Socket Layer protocol

Secure Socket Layer (SSL) is necessary for secure communication between Active Workspace and Google Online Service. As the connections go through the Active Workspace gateway, which is a single entry point to go to the Teamcenter servers, you must configure SSL here.

You can also configure the Teamcenter Google Online microservice with SSL.

To do this manually:

1. Acquire the SSL certificate. The certificate is in the **X.509** format.

2. Open the *appsettings.json* file located at:

   - **Windows**

     *TC_ROOT\microservices\tcgoogleweb-version\TcGoogleWeb*

   - **Linux**

*TC_Root/microservices/tcgoogleweb-version*

The *appsettings.json* file is copied to each running container at startup. Therefore, when the file is updated, restart the containers for the changes to take effect.

3. Edit the **Kestrel** section:

   a. Change **http** to **https**.

   b. Specify the certificate file name in the certificate **Path** value, or rename the certificate file name to *certificate.pfx* and retain the certificate **Path** value as is.

Example:

```
"Kestrel":{
   "Endpoints":{
      "Https":{
         "Url": "https://0.0.0.0:0",
         "Certificate":{
            "Path": "certificates/certificate.pfx"
          }
       }
   },
   "Certificates":{
      "Default":{
         "AllowInvalid": true
       }
   }
}
```

4. Decode and open the *appsettings.Seucrity.json* file, edit the `Kestrel` section, and set the **CertificatePassword** of the host SSL to the cerificate's password.

Example:

```
"Kestrel":{
   "Endpoints":{
      "Https:{
         "Certificate":{
            "Password": "ceritificatepassword"
         }
       }
   }
}
```

5. Regenerate the *JsonSettingsConfig.json* file:

```
JsonSettingsConfigUtility.exe -generateConfigFile
JsongSettingsConfig.json appsettings.Security.json TcSecurity
GoogleConfig Kestrel
```

6. Encode the *appsettings.Security.json* file again:

```
JsonSettingsConfigUtility.exe -encode JsonSettingsConfig.json
```

7. Copy the *.pfx* file to:

   - **Windows**

     *tcgoogleweb_service-version\TcGoogleWeb\certificates*

   - **Linux**

     *tcgoogleweb_service-version/certificates*

   If you copy the *.pfx* file to a different location, enter the location path as the **Certificate Path** value in the *appsettings.json* file located at:

   - **Windows**

     *TC_ROOT\microservices\tcgoogleweb-version\TcGoogleWeb*

   - **Linux**

     *TC_Root/microservices/tcgoogleweb-version*

     The *appsettings.json* file is copied to each running container at startup. Therefore, when the file is updated, restart the containers for the changes to take effect.

8. For Linux, open the *tcgoogleweb_service.yml* file:

   a. Add the **-source** and **target** parameters with values in the **configs** section.

```
configs:
   -source: certificate
   target: /app/certificates/certificate.pfx
```

   b. Add the **certificate** and **file** parameters with values at the top-level **configs** section.

```
configs:
   certificate:
      file: /tc_deployment_path/microservices/service_directory/
certificates/certificate.pfx
```

## Enable single sign-on

If single sign-on (SSO) is set up at your site, then you must enable it. To do so, in the *appsettings.json* file, ensure that the connection node has SSO enabled:

```
"TcConnection": {
   "SsoEnabled": "true"
}
```

The *appsettings.json* file is located at:

- **Windows**

  *TC_ROOT\microservices\tcgoogleweb-version\TcGoogleWeb*

- **Linux**

  *TC_Root/microservices/tcgoogleweb-version*

  The *appsettings.json* file is copied to each running container at startup. Therefore, when the file is updated, restart the containers for the changes to take effect.

# Troubleshooting Teamcenter Google Online

### Document icon displayed in the Active Workspace viewer instead of the file

If the Active Workspace viewer displays a document icon instead of a file, it may be due to some missing components. To resolve this issue, verify that the following values are included in the **AWC_defaultViewerConfig.VIEWERCONFIG** preference:

- **Gg1GDoc.Awp0GoogleViewer=Gg1word**
- **Gg1GSheet.Awp0GoogleViewer=Gg1excel**
- **Gg1GSlide.Awp0GoogleViewer=Gg1powerpoint**

> Note:
>
> If any value is missing, do not add it manually. A missing value indicates that the preference is not set correctly as the **Google Online Client Viewer** feature is not installed. To resolve this, run Teamcenter Environment Manager or Deployment Center to install the feature.

## Teamcenter Google Online service log files

You can refer to the Teamcenter Google Online service log files to troubleshoot issues found while setting up Google Online within Teamcenter. If you have configured logging aggregation, you can view the log files in the log aggregator, which collects logs from distributed microservices on various microservice nodes and forwards the logs to a single location. If you have not configured logging aggregation, the log files are located in *tc_root/microservices/tcgoogleweb-version/TcGoogleWeb/logs* on Windows and are available from the Docker container on Linux.

# 11. Using the sample BMIDE Document Management template

## About the sample BMIDE Document Management template

After you install the **required components** for Document Management, you must configure your deployment set up so that the end users can use Teamcenter to manage documents. You perform these configurations within a template in BMIDE. To help you with the configuration process, a sample BMIDE Document Management template (*sdm0sampledocmgt*) is provided. This sample template contains the following BMIDE Document Management sample objects. You might want to configure them considering your organization's business requirement for managing documents. For example, you can upload your Office file templates in place of example files.

- Item revision definition configuration (IRDC)
- Dispatcher service configuration
- System stamp configuration
- Print configuration
- Conditions
- Tool

To use the sample BMIDE Document Management template, you must:

- Install it through **Deployment Center** by selecting **Teamcenter→Teamcenter→Sample Document Management**. Alternatively, install it through **Teamcenter Environment Manager (TEM)** by selecting the **Extensions→Sample Document Management** feature.

  On installing the sample BMIDE Document Management template, the sample Document Management template (*DMTemplate*) containing the sample objects and files is automatically installed.

  > Example:
  >
  > *DMTemplate* contains the sample template for authoring a *functional specification* in **DMTemplateForSampleFSWordDataset**.

- **Import the sample BMIDE Document Management template** into BMIDE to view the sample objects such as IRDC, Dispatcher Service Config, System Stamp Configuration, and Print Configuration in BMIDE.

- (Optional) **Create a new *dependent* template** from the sample BMIDE Document Management template.

For more information about these files, see *server-install-location\samples\document_management\readme.pdf*.

# Import the sample BMIDE Document Management template

1. In Business Modeler IDE, click **File→Import**.

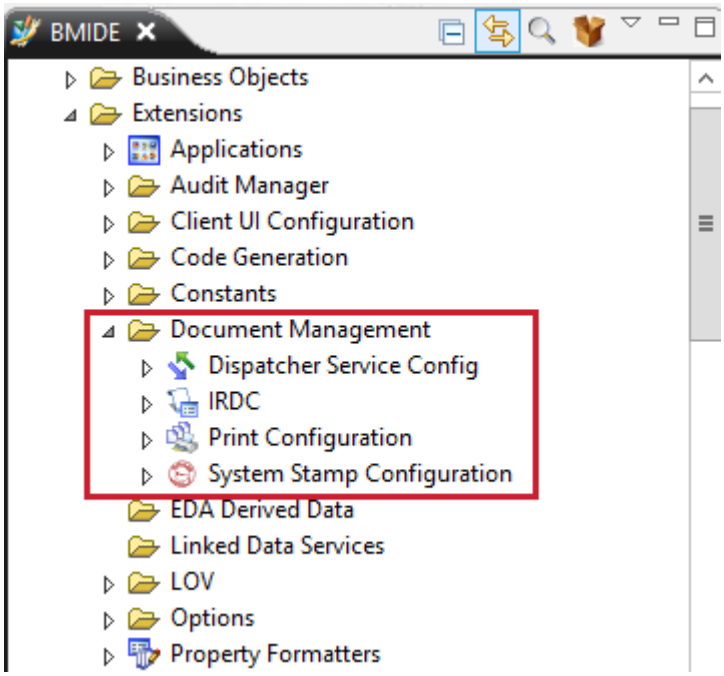
2. In the **Import** dialog box, select **Business Modeler IDE→Import a Business Modeler IDE Template Project** and click **Next**.

3. Click the browse button next to **Project contents** and open the *server-install-location\samples\document_management\bmide_template\sdm0sampledocmgt* folder.

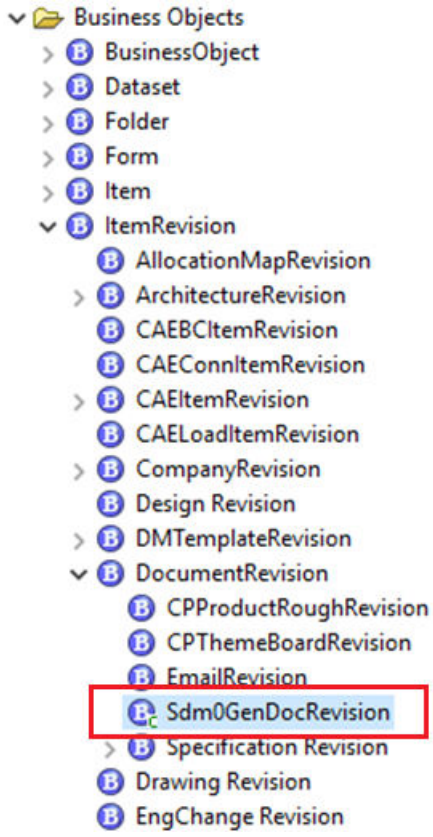
4. Click **Finish**.

5. To verify if you imported the template successfully, perform the following steps:

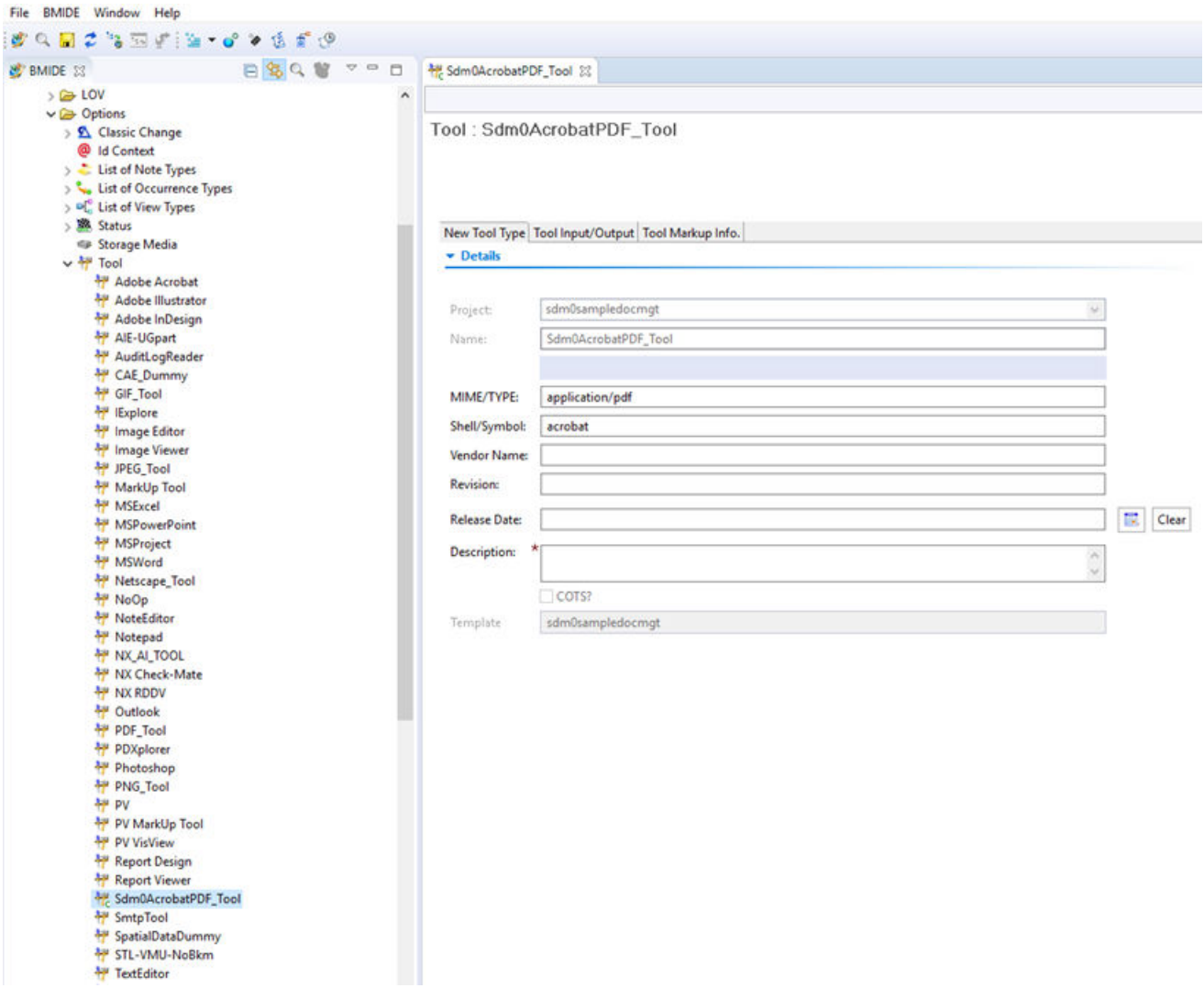
   - Click **Extensions→Document Management** to view the samples available within different folders such as **Dispatcher Service Config**, **IRDC**, **Print Configuration**, and **System Stamp Configuration**.




   - Click **Business Objects→ItemRevision→DocumentRevision** and verify if the document revision for Sample General Document named as **Sdm0GenDocRevision** is created.




- Click **Options→Tool** and then verify if the Sdm0AcrobatPDF_Tool is created as shown in the following image. You can then **enable applications for document markup**.




Next, you can view the sample tool objects under **Extensions→Options→Tool** and the sample conditions under **Extensions→Rules→Conditions**.

You can use these samples as the basis for creating configuration objects as per your site requirements.

6. If you use rich client, modify the TcViewMarkupApplicationPref preference for defining the order of the applications that can be used for viewing and marking up the documents.

To verify if the Document Management template file (*DMTemplate*) is imported successfully:

1. In My Teamcenter, click **File→New→Item**.

2. In the **New Item** dialog box, select **Document** and click **Next**.

3. Enter an **ID** and a **Name** for the document and click **Next** twice in succession.

4. In **Document Subject**, enter the type of the document. For example, Functional Specification.

   If you have configured different templates in the BMIDE for the Sample Document Management template, select the required template. For example, suppose that you configured a template for Functional Specification. In such a case, select **Functional Specification** as the subject of the document.

5. Click **Finish**.

6. Expand the newly created document revision and click the **Viewer** tab to verify that the correct functional specification document is attached to it.

## Import and relate logical objects for sample BMIDE Document Management template

After the sample BMIDE Document Management template is imported, you can import the logical objects and relate them.

### Procedure

1. Open Teamcenter command prompt.

2. Execute the following command:

   ```
   admin_data_import -adminDataTypes=LogicalObjects
   -mergeOption=LogicalObjects:keep_target -u=<username> -p=<password>
   -g=dba -inputPackage=DMSampleLOTc116.zip
   ```

   The import action is completed in a few minutes.

   > Note:
   >
   > The Document Subject attribute of the **Sdm0GenDocSubject** business object in BMIDE is tied to a LOV. The LOV selects the appropriate IRDC rule and DM template to use for each example in the sample Document Management template.

3. Log on rich client as an administrator.

4. Search for the Folder name **DMTemplates** and copy it in the My Teamcenter area.

5. Expand the folder and locate the Microsoft Office datasets in the DMTemplate object.

6. Edit the properties of the datasets and set the required Logical Objects for attribute exchange.

By default, the samples work with the **DMSampleLODocAttach** type and **TC_Attaches** relation for the **Logical Object Type Relation** field and **DMSampleLOPageTypeRel** type and **Document Page** relation for the dataset.

## Create a new BMIDE template from the sample BMIDE Document Management template (sdm0sampledocmgt)

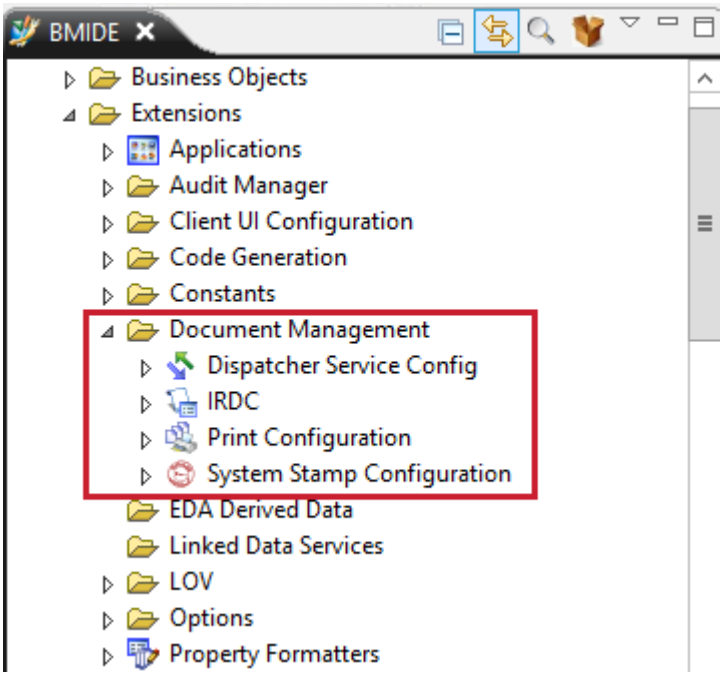
You can create a new BMIDE template from the sample BMIDE Document Management template (sdm0sampledocmgt) that is provided by default. To do so:

1. In Business Modeler IDE, click **File→New→New Business Modeler IDE Template Project**.

2. In the **New BMIDE Template Project** dialog box, enter the required information and click **Next**.

3. Select **Sample Document Management** as the dependent template and click **Finish**.

4. To verify that you have created the template successfully, click **Extensions→Document Management** to view the samples available within different folders such as **Dispatcher Service Config** and **IRDC**.



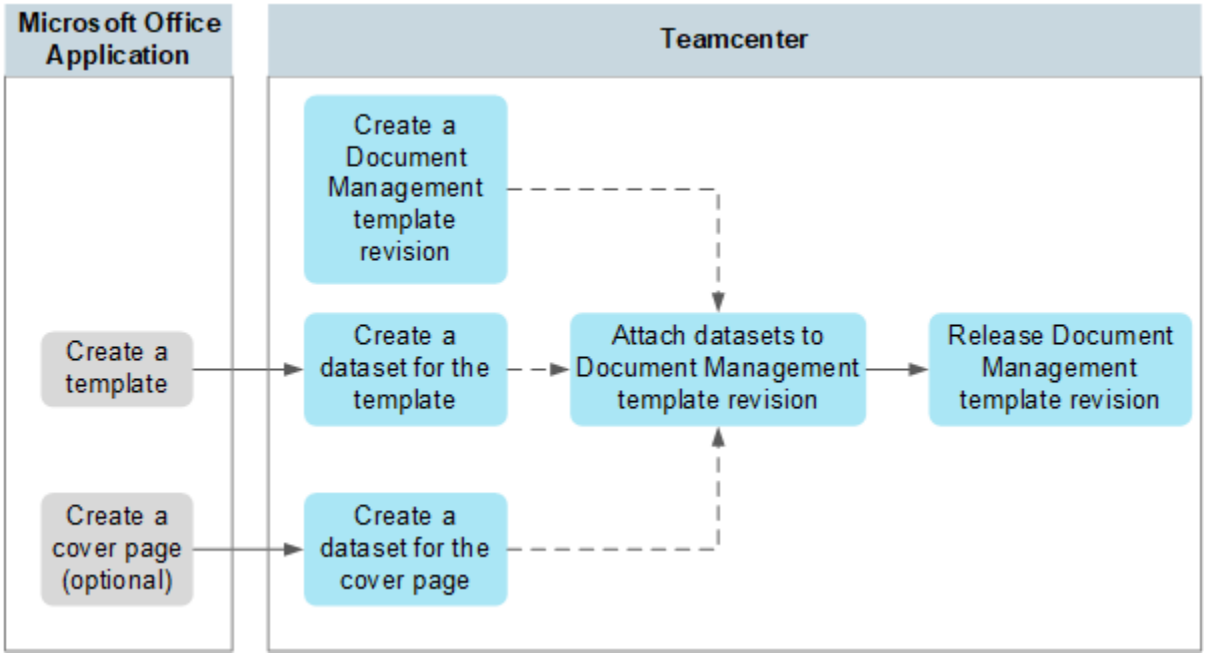

Next, you can view the sample tool objects under **Extensions→Options→Tool** and the sample conditions under **Extensions→Rules→Conditions**.

You can use these samples as the basis for creating configuration objects as per your site requirements.

# 12. Setting up Microsoft Office templates in Teamcenter

You can style Microsoft Office documents by using templates. Templates ensure that a consistent style is followed in documents authored by users. After you design templates in Office applications, you must save the templates in Teamcenter so that the user can starting using them.

### Task flow to set up Microsoft Office templates in Teamcenter




### Create templates and cover pages in Microsoft Office applications

Create templates and cover pages (optional) in different Microsoft Office applications such as Microsoft Word or Excel, as per your requirements.

To create a template file, you may refer to the sample Document Management templates that your imported by importing the *DMTemplate.xml* sample file.

### Create datasets for Microsoft Office templates and cover pages

You create a *dataset* to hold a Microsoft Office template file. You can also create a dataset to hold the cover page, if required.

To create a dataset:

1. Click **File→New→Dataset**.

2. In the **New Dataset** dialog box, enter a name and description for the dataset.

3. Select a dataset type from the **Type** bar, for example, **MSWordX**.

   If you do not see the type you are looking for, click **More** to display all defined dataset types.

4. Click the browse button next to **Import** to import the document template file.

5. For a template file, set **Relation** as **Attaches** and for a cover page, set it as **Document Page**.

6. Click **OK**.

### Create a Document Management template revision

1. Select the folder where you want to create the Document Management template revision and click **File→New→Item**.

2. In the **New Item** dialog box, select **Document Management Template** from the **Business Object Type** list.

3. Click **Next**.

4. Enter the required information and click **Finish**.

### Attach template files to a Document Management template revision

1. Right-click the template file (dataset) you wish to attach and click **Copy**.

2. Right-click the Document Management template revision to which you wish to attach the dataset and click **Paste**.

### Attach cover pages to a Document Management template revision

1. Right-click the cover page (dataset) you wish to attach and click **Copy**.

2. Right-click the Document Management template revision to which you wish to attach the dataset and click **Paste**.

3. Right-click the cover page attached to the Document Management template revision and click **Properties on Relation**.

4. In the **Properties** dialog box, set the **Page Type** as **Cover Page** and click **OK**.

## Release a Document Management template revision

You must release the Document Management template revision and its attached datasets so that authors can start using it. To release the template and its attached datasets:

1. Select the Document Management template revision.

2. Choose **File→New→Workflow Process**.

3. In the **New Process** dialog box, select **TCM Release Process** from the **Process Template** list.

4. In the **Attachments** tab, expand the document template revision and select the datasets to be released.

5. Click **OK**.

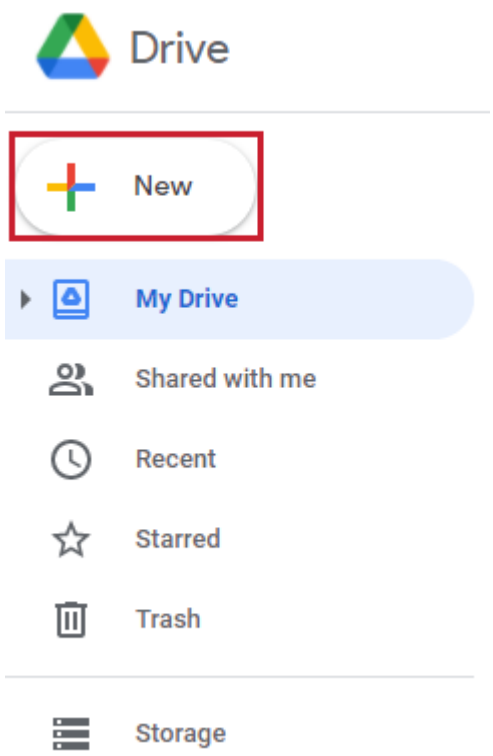
After releasing the Document Management template revision, you can use it to **standardize the behavior of a document** at different stages of its life cycle in Teamcenter.

# 13. Set up Google document templates

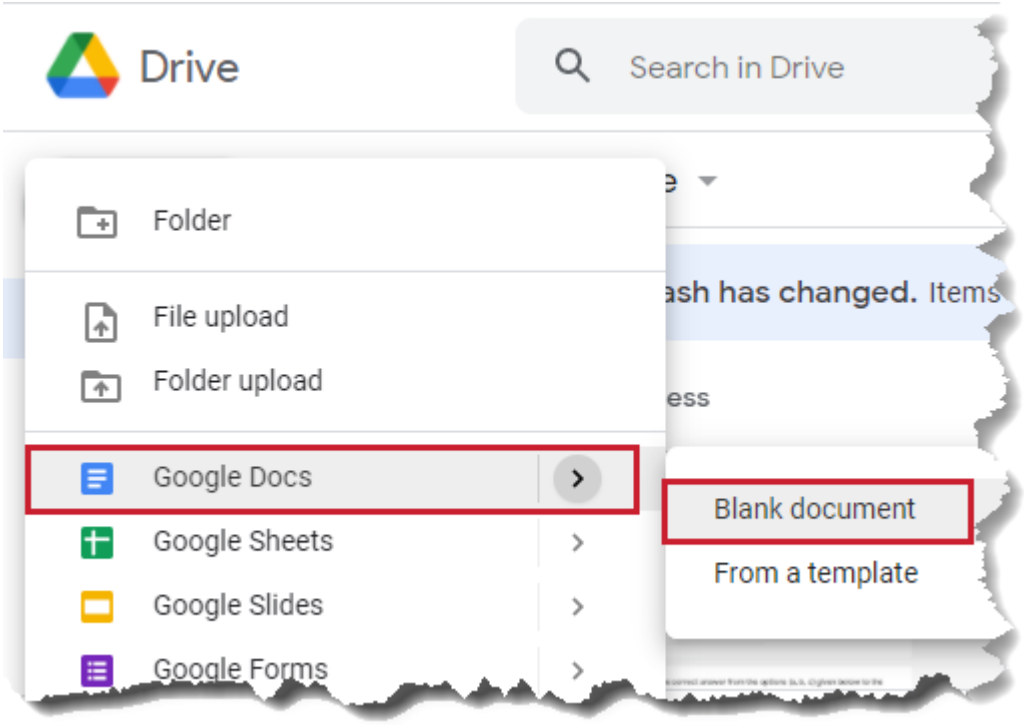
You can style Google documents by using templates. Templates ensure that a consistent style is followed in files authored by users. To set up a Google document template:

1. Open your Google Drive, https://drive.google.com/drive/my-drive.

2. Click **New**.



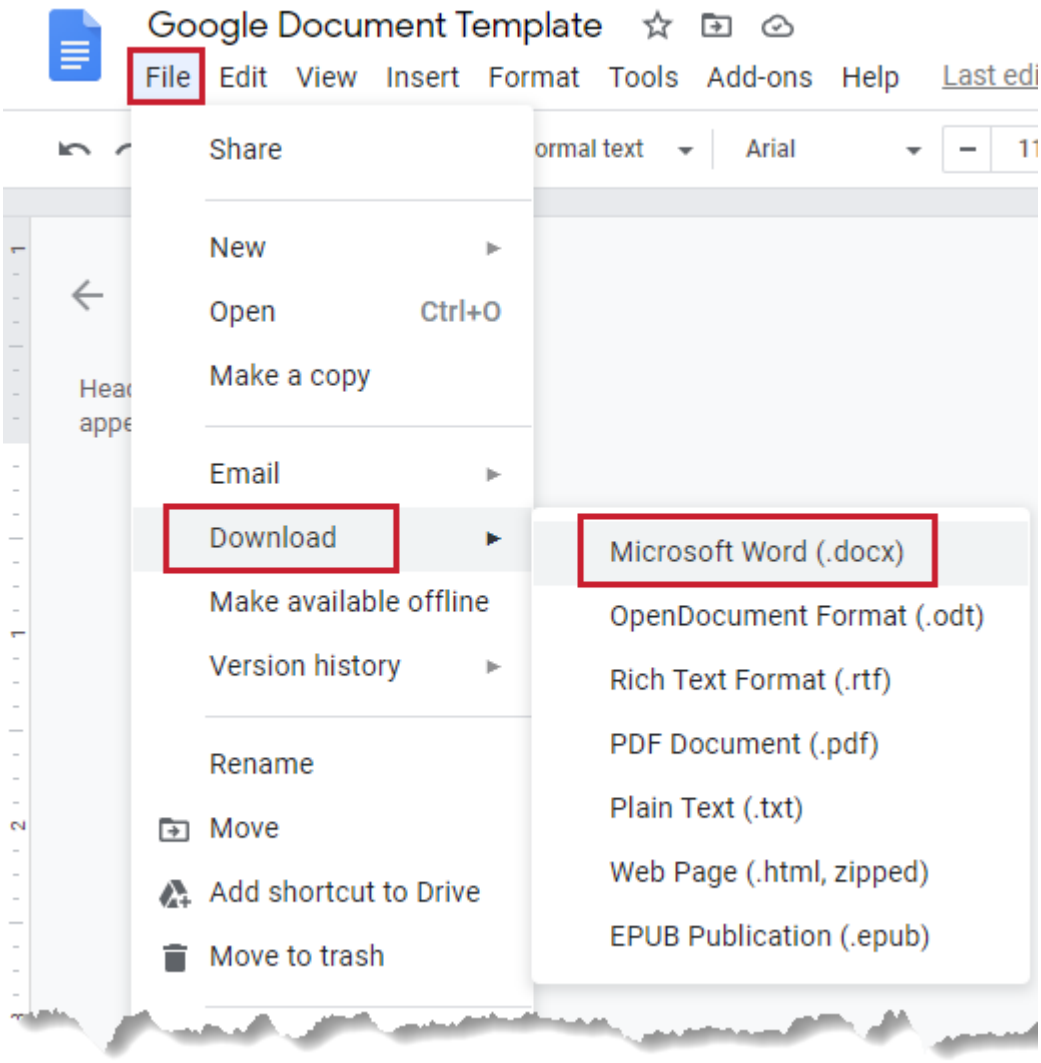

3. Click **Google Docs** > **Blank Document**.




4. In the new file page, click **File** > **Rename** to name the new document.

5. Author the required content.

6. Click **File** > **Download** > **Microsoft Word (.docx)**.




7. The file is downloaded in the **Downloads** folder.

8. Add this downloaded file as the template file while create the **item revision definition configuration (IRDC)** to standardize the behavior of a document.

# 14. Synchronizing attributes between Teamcenter and other applications

By synchronizing attributes between Teamcenter and other applications, you ensure that up-to-date information is available at both instances: in the Teamcenter environment and in other application files.

Logical objects-based attribute exchange can follow any path you can define in a logical object to gather data, but you cannot write from **File to Teamcenter** in Excel or PPTX. The Microsoft Office Client attribute exchange can write bi-directionally; however, you cannot follow paths to other objects to collect data. You can only exchange with the item and item revision.

You can set up the attribute synchronization:

- **Between Teamcenter and Microsoft Office** (Word, Excel, and PowerPoint) files by using **logical objects**.

  > Warning:
  >
  > After an attribute's value is synchronized between Teamcenter and Microsoft Office, and later it is deleted in Teamcenter, the null value cannot be synchronized to Microsoft Office, so the attribute continues to have the previous value.

- **Between Teamcenter and Microsoft Office** (Word, Excel, and PowerPoint) files by using Teamcenter Client for Microsoft Office. Teamcenter Client for Microsoft Office supports bidirectional attribute exchange between Teamcenter and Microsoft Office.

  > Warning:
  >
  > After an attribute's value is synchronized between Teamcenter and Microsoft Office, and later it is deleted in Teamcenter, the null value cannot be synchronized to Microsoft Office, so the attribute continues to have the previous value.

- **From Teamcenter to Adobe PDF files** by using logical objects.

Along with attributes, you can also synchronize logos, distribution statements, and workflow signoff tables. Workflow signoff table is the table the author sees in the **Workflow Signoff** dialog box while sending a document revision through a review workflow.

# 15. Synchronizing attributes between Teamcenter and Microsoft Office files by using logical objects

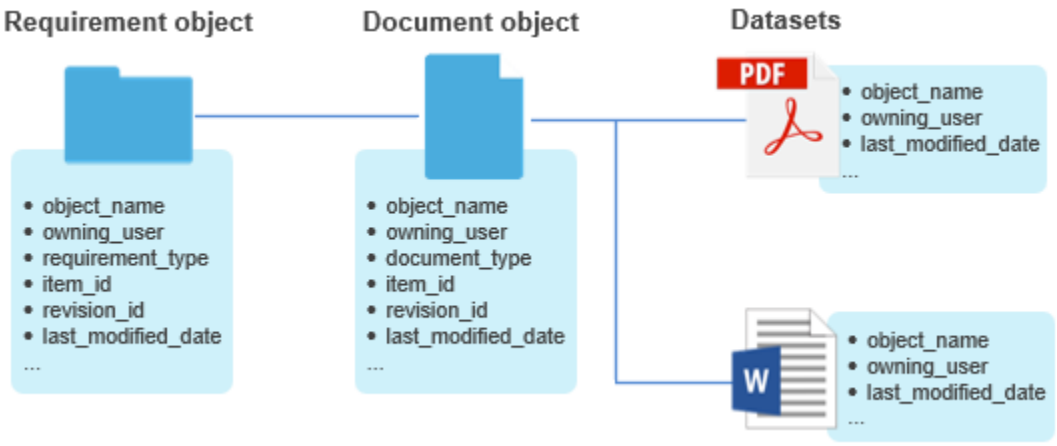
## About logical objects

A *logical object* is a container that consolidates attributes (properties) from different related business objects. For example, a document (item) and its related datasets are different objects in Teamcenter. The attributes of the document can be its ID, name, or description. The different attributes of a dataset can be author, created date, or reviewed by.

The following example shows how you can synchronize attributes, consolidated by the logical object, from Teamcenter to a Microsoft Word document and to a PDF file.

**Example**

Consider that there is a requirement object in Teamcenter which has a reference relation to a document object. The document object has two attached files (datasets), one Word and one PDF.

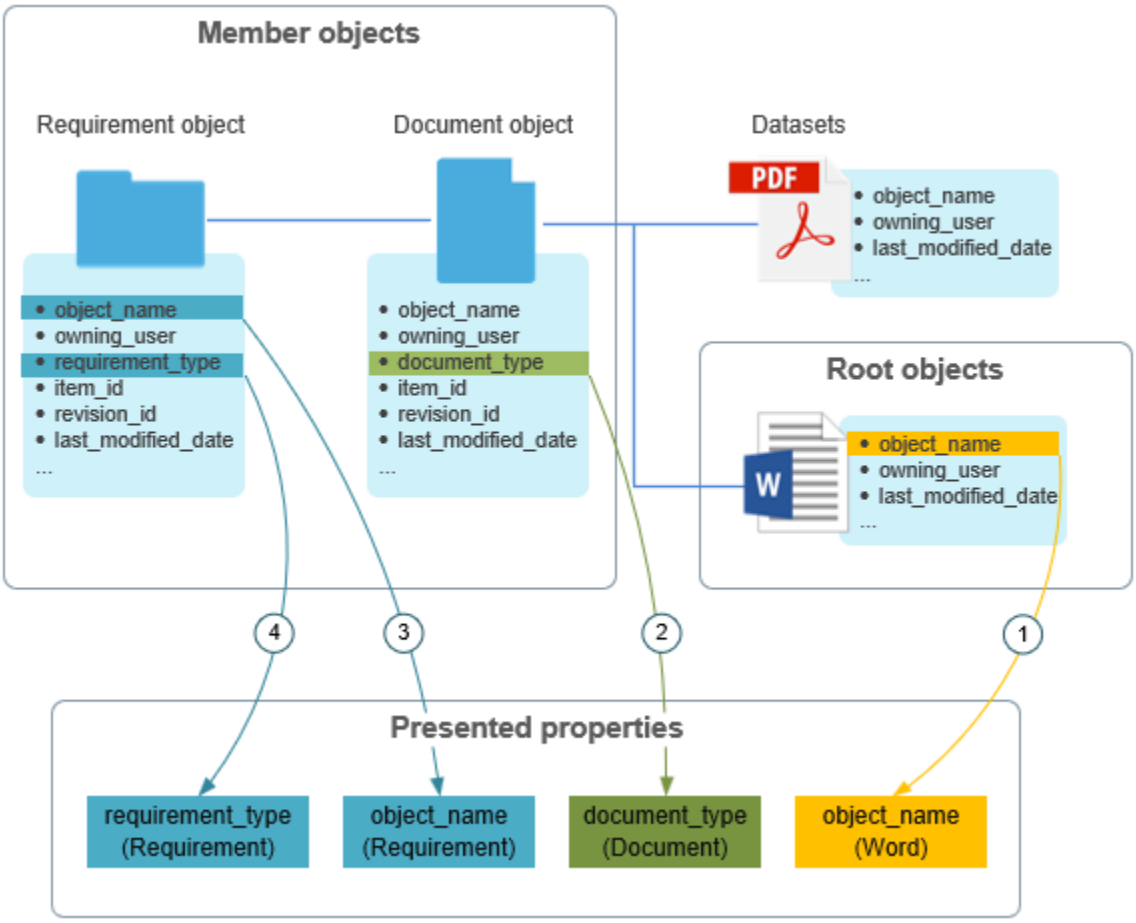



You want to consolidate four attributes from these objects:

- **object_name** and **requirement_type** from the requirement object.
- **document_type** from the document object.
- **object_name** from the Word or PDF file attached to the document.

For consolidating these attributes, you define a logical object. A logical object is composed of *root object*, *member objects*, and *presented properties*. You begin consolidating the properties from the

dataset. This makes the dataset the root object. You then traverse backward tracing its relation to the document object and then to the requirement object. This makes the document and requirement objects the member objects. While traversing from the dataset to the document object and then to the requirement object, you collect the required four attributes in the logical object. These four attributes are the presented properties.




After consolidating the attributes in a logical object, you use the logical object to synchronize attributes between Teamcenter and Microsoft Office files and PDF files. When you synchronize attributes between Teamcenter and Microsoft Office files, you cannot synchronize attributes from Teamcenter to Office files and from Office files to Teamcenter at the same time.

> Note:
>
> Even if you used attribute-level security while implementing Teamcenter, the properties that are used for attribute exchange are still synchronized in the document. The attribute exchange overrides the attribute-level security.

## Task flow to synchronize attributes by using logical objects

1. **Define a logical object**.

2. **Define when to add the Teamcenter attributes in template files**.

3. To synchronize the attributes between Teamcenter and Microsoft Office files, **specify the logos, distribution statements, and workflow signoff tables** to be included in the Office files in the Document Management configuration file.

4. To synchronize the attributes between Teamcenter and Microsoft Office files, define the placement of attributes, logos, distribution statements, and workflow signoff tables in the **Word**, **Excel**, and **PowerPoint** files, if required.

   To synchronize the attributes between Teamcenter and PDF files, define the placement of attributes, logos, distribution statements, and workflow signoff tables in the PDF files, if required.

5. Include the logos, distribution statements, and workflow signoff tables in the synchronized files by setting up a **system stamp configuration**.

6. **Relate the logical object with a file (dataset)**.

When you synchronize attributes between Teamcenter and Microsoft Office files, the Teamcenter attributes along with logos, distribution statements, and workflow signoff tables are included in a file synchronously when a user creates or renders a file. They are included in a file asynchronously through Dispatcher when a user checks in, saves as, or revises a document revision. For more information about workflow signoff table, see *server-install-location\samples\document_management\readme.pdf*. To get the latest information in an Office file, a user can also send the file through an attribute exchange workflow process. For this, you must set up a workflow task for the users by using the **DOCMGT-update-docprop-logicalobject** action handler.

When you synchronize attributes between Teamcenter and PDF files, The Teamcenter attributes along with logos, distribution statements, and workflow signoff tables are included in a PDF file when a user sends the file through an attribute exchange workflow process. For this, you must set up a workflow task for the users by using the **DOCMGTAPP-apply-pdf-control** action handler.

Finally, **you may verify if you have set up the attribute exchange correctly**.

## Define logical objects

You can define logical objects for attribute synchronization. Note that the logical objects do not support the runtime properties for attribute synchronization.

1. In Active Workspace, log on with DBA credentials and select **Default** as the **Workspace**.

2. Click the **LOGICAL OBJECTS** tile on the home page.

3. On the **Logical Object Configuration** page, choose **More Commands** ••• > **New** ✳ > **Define logical object** ⊕.

4. In the **Add** panel:

   a. Enter **Internal Name**, **Name**, and **Description** for the logical object.

   b. In the **Root Object** list, click a root object. For example, click **Dataset**.

   c. Click the **Retrieve Classification Data** checkbox if you want to retrieve the classification data.

   d. In the **Parent Logical Object** list, click a parent logical object. For example, click **Logical Object**.

   e. Click **Add**.

5. In the **Overview** tab, click **Add** ⊕ to add new members to the logical object.

6. In the **Add Member** panel:

   a. Enter **Member ID** and **Display Name**.

   b. Select **Backward** in **Segment 1**.

   c. Select **Business Object**, for example, **DocumentRevision**.

   d. Select **Relation or Reference**, for example, **Tc_Attaches**.

   e. Click **Add Segment** ⊕ in **Segment 2**.

   f. Select **Forward**.

   g. Select **Relation or Reference**, for example, **items_tag**.

   h. Select **Business Object**, for example, **Document**.

   i. Click **Add**.

7. Click **Add** ⊕ again to add another member to the logical object to relate the business object with the dataset by using a different relation.

8. For each member, add member properties that you want to consider for document rendering. To do so:

   a. In the **Presented Properties** section, click **Add** ⊕.

   b. In the **Add** panel, click a property of the member in the Root Or Member ID list. For example, **fnd0Root**.

c. In the **Member Properties** list, click the required property, for example, **last_mod_date**.

d. Click **Add**.

You can add as many properties as required for each member.

9. Go to the **Exchange Configuration** tab to optionally set the attribute synchronization direction (**Teamcenter to File** or **File to Teamcenter**). If you do not specify the exchange direction, attributes are exchanged from **Teamcenter to File** by default.

10. In the **Exchange Configuration** tab, click **Refresh** to load the exchange configurations corresponding to the presented properties that you added earlier.

Default direction for attribute exchange is **Teamcenter to File**. If **File to Teamcenter** direction for attribute exchange is required, you must set **Is Writable** property to **TRUE** in the **Overview** tab.

> Note:
>
> Attribute exchange cannot be displayed in Microsoft Office Online Excel files. This is a limitation of Microsoft Office Online.

> Tip:
>
> To delete a logical object, you must first delete all the presented properties available in the **Exchange Configuration** tab.
>
> Refresh the presented properties in the **Exchange Direction** tab after deleting presented properties in the **Overview** tab.

## Define when to add the Teamcenter attributes in template files

1. In the existing BMIDE template project, click **Business Objects→ItemRevision**.

2. Double-click **ItemRevision** and in the **Main→Business Object Constants** tab, locate **Fnd0RelToDatasetForLOAttrExch** and click **Edit**.

3. In the **Business Object Constant** dialog box, enter the relations that must be considered including attributes in datasets attached to a Document Management template revision. You can specify more than one relation by using commas, for example, **TC_Attaches**, **IMAN_specification**.

4. Click **Finish**.

5. In the **Main→Business Object Constants** tab, locate **Fnd0TriggerLOAttrExch** and click **Edit**.

6. In the **Business Object Constant** dialog box, select one of the following in **Value** and click **Finish**:

- **Disabled**

  This value indicates that the Teamcenter attributes will not be added to template files automatically. This is the default value.

- **Configured**

  This value indicates that the Teamcenter attributes will be added to template files automatically. The attributes will also be added to the datasets attached to the Document Management template revision with relations specified in the **Fnd0RelToDatasetForLOAttrExch** business constant.

- **Enabled**

  This value indicates that the Teamcenter attributes will be added to template files and to all the attached datasets automatically.

## Specify the logos, distribution statements, and workflow signoff tables to be displayed in the stamped files

You specify information about logos, distribution statements, and workflow signoff tables in the Document Management configuration XML file. You may refer to the Document Management configuration schema file (**DocMgmtConfig.xsd**) located at *TC_ROOT\sample\document_management\schema* to create the configuration XML file. In this file, you can specify the logos to be used, when to use a logo, and where to place the logo in a PDF file. You can also specify similar information for distribution statements, workflow signoff tables, and Teamcenter attributes. After specifying the required information, you must **attach the configuration XML file** to a Document Management template revision for stamps.

For more information about workflow signoff table, see *server-install-location\samples\document_management\readme.pdf*.

### A snippet of a list of columns in the workflow signoff table

You can define the columns to be displayed in a workflow signoff table. To do so, add the column names in the Document Management configuration XML file:

```
...
<!--workflow signoff table column names to display, can be removed or
reordered
These names listed below are internal names, not the display names-->

<dmcfg:WorkflowSignOffColumnsToDisplay>
   <dmcfg:WorkflowSignoffColumnName>processName</
dmcfg:WorkflowSignoffColumnName>
   <dmcfg:WorkflowSignoffColumnName>group</dmcfg:WorkflowSignoffColumnName>
```

```
   <dmcfg:WorkflowSignoffColumnName>role</dmcfg:WorkflowSignoffColumnName>
   <dmcfg:WorkflowSignoffColumnName>userName</
dmcfg:WorkflowSignoffColumnName>
   <dmcfg:WorkflowSignoffColumnName>userId</dmcfg:WorkflowSignoffColumnName>
   <dmcfg:WorkflowSignoffColumnName>decisionStringDisplay
   </dmcfg:WorkflowSignoffColumnName>
   <dmcfg:WorkflowSignoffColumnName>decisionDateString
   </dmcfg:WorkflowSignoffColumnName>
   <dmcfg:WorkflowSignoffColumnName>comments</
dmcfg:WorkflowSignoffColumnName>
</dmcfg:WorkflowSignOffColumnsToDisplay>
...
```

By default, all columns are displayed in the workflow signoff table. You can add, remove, or reorder the columns as required.

### A snippet of the logo and distribution statement information

```
...
<dmcfg:logoList>
   <dmcfg:logoDataset datasetName="Logo" ID="1">
      <dmcfg:enableWhen>
         <dmcfg:property key="fnd0CurrentLocationCode" value="Shoreview" />
      </dmcfg:enableWhen>

      <dmcfg:MSOfficePlacement>
         <dmcfg:tag value="Logo_1" />
      </dmcfg:MSOfficePlacement>
   </dmcfg:logoDataset>
</dmcfg:logoList>

<dmcfg:distStatementList>
   <dmcfg:distStatementDataset datasetName="ExportControlledITAR" ID="1" />
      <dmcfg:enableWhen>
         <dmcfg:property key="fnd0CurrentLocationCode" value="Shoreview" />
      </dmcfg:enableWhen>

      <dmcfg:MSOfficePlacement>
         <dmcfg:tag value="Statement_1" />
      </dmcfg:MSOfficePlacement>


   <dmcfg:distStatementDataset datasetName="NonExportControlledITAR"
ID="2" />
      <dmcfg:enableWhen>
         <dmcfg:property key="fnd0CurrentLocationCode" value="Cypress" />
      </dmcfg:enableWhen>
```

```
      <dmcfg:MSOfficePlacement>
         <dmcfg:tag value="Statement_2" />
      </dmcfg:MSOfficePlacement>

   </dmcfg:distStatementDataset>
</dmcfg:distStatementList>
...
```

## Attach files to a Document Management template revision for stamps

To attach the logo, distribution statement, Document Management configuration XML file, PDF command file, and MetaData Stamp file to a Document Management template revision that you created for stamps:

1. In My Teamcenter, select the required system stamp Document Management template revision.

2. Click **File→New→Dataset**.

3. In the **New Dataset** dialog box:

   a. Enter the **Name** of the dataset as the name of the logo or distribution statement specified in the Document Management configuration XML file. For example, the **datasetName** of a logo in the configuration XML file is **Logo_1**. Therefore, the name of the dataset for the logo must also be **Logo_1**.

   b. Import the logo image file, distribution statement text file, or Document Management configuration XML file.

   c. In **Relation**, select **Document Page** while attaching the logo, distribution statement, or Document Management configuration XML file.

      For PDF command and MetaData Stamp files, select **Tc_Attaches**.

   d. Click **OK**.

4. Right-click the newly created dataset and click **Properties On Relation**.

5. In the **Properties** dialog box:

   a. Click **Show Empty Properties** to view all properties.

   b. In **Page Type**, select **Logo**, **Distribution Statement**, or **Document Configuration** based on the entity you are creating the dataset for.

   c. Click **OK**.

6. Select the dataset and click **File→New→Workflow Process** to start a workflow process to release the dataset.

7. To release the dataset, in the **New Process Dialog** dialog box:

   a. Select **TCM Release Process** in **Process Template**.

   > Note:
   >
   > The default value of the release status for Document Management template (*DMTemplate*) is **TCM Released**. This value is set to *MaturityStatuses* constant in BMIDE. You can customize it by adding a status type. You can also understand what are global constants, business objects constants, and how to run data model reports to understand the *MaturityStatuses* constant.

   b. In the **Attachments** tab, expand **Targets** to select the logo, distribution statement, or Document Management configuration schema file that is attached to the dataset.

   c. Click **OK**.

   A release flag is displayed next to the released dataset.

## Define the placement of attributes in Microsoft Word documents

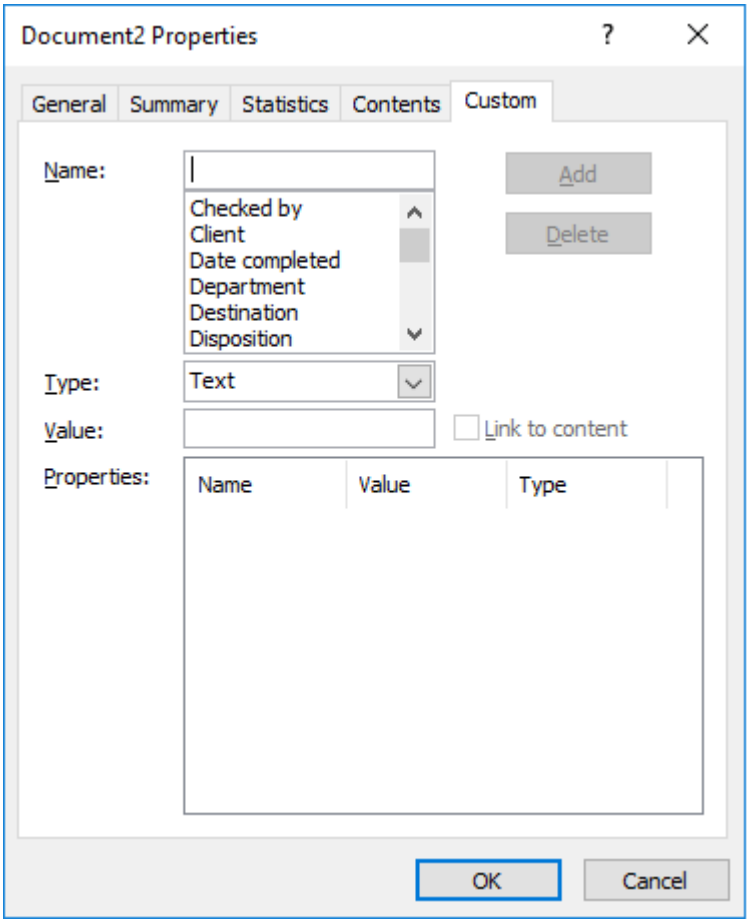
1. Check out the required template file.

2. Open a Word file in which you want to include information.

3. You must add each property that you defined for a logical object member as a custom property. To add each property:

   a. Click **File→Info→Properties→Advanced Properties**.

   b. In the **Properties** dialog box, click **Custom**.

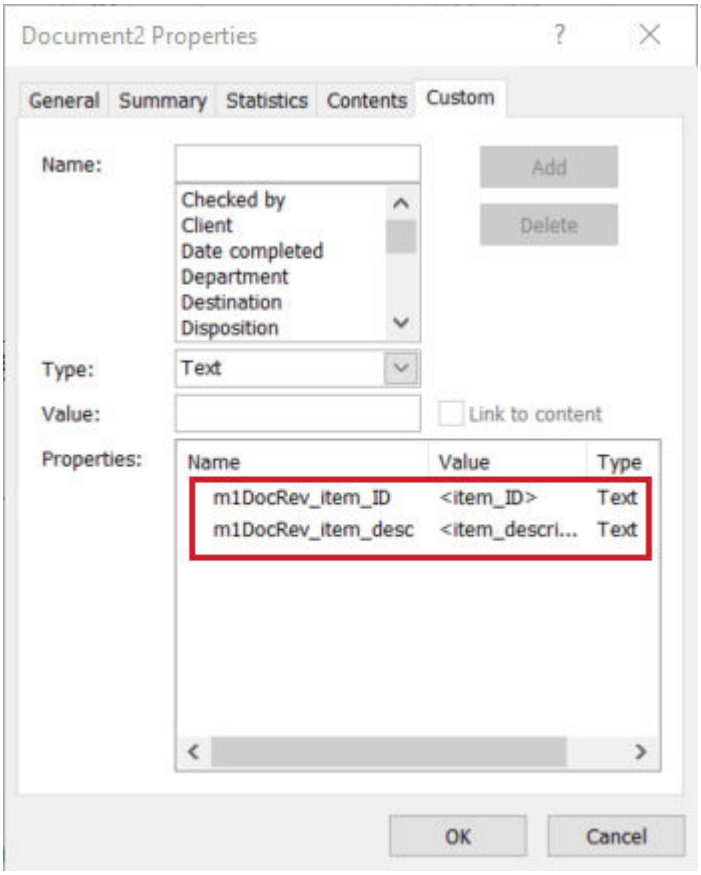



c. Enter the ID of the member property in **Name**, for example, **m1DocRev_item_ID**.

d. Select **Type**.

e. Enter the information in **Value**. As a default value is required, enter either a space or information that must be displayed in the document.

f. Click **Add**.

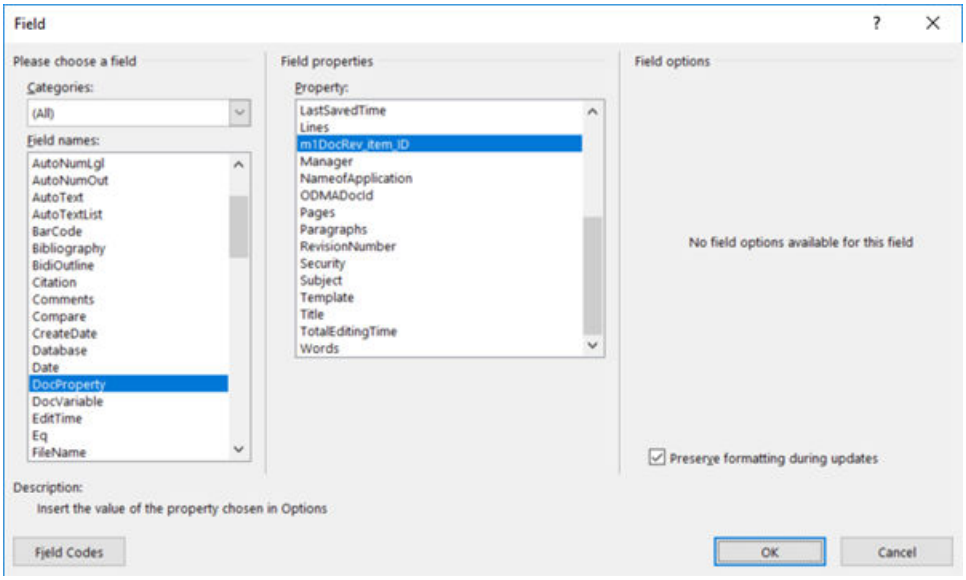



g. Add other required properties and finally click **OK**.

h. To insert the newly created custom properties:

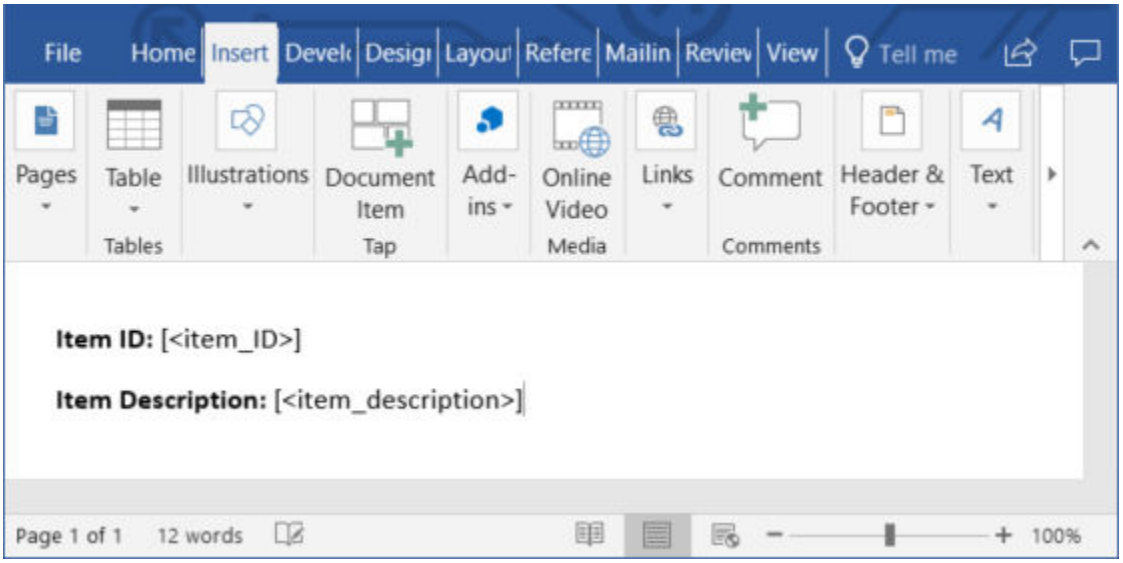
   A. In Word, place the cursor at the location where you want to enter a custom property and click **Insert→Quick Parts→Field**.

   B. In the **Field** dialog box, select **DocProperty** in **Field names** and select a custom property that you created in **Property**.




C. Click **OK**.

The value of the custom property is inserted at the selected location.




To see the value of a custom property later, right click the inserted property and click **Toggle Field Codes**.

4. To insert distributions statements:

   a. Click **FILE→Options**.

   b. In the **Word Options** dialog box, click **Customize Ribbon**.

c. In **Customize the Ribbon**, select **Main Tabs**.

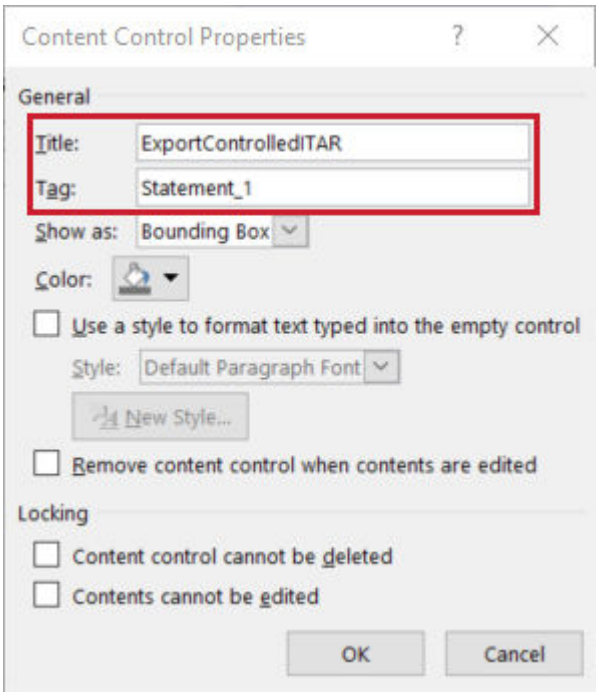
d. In **Main Tabs**, select the **Developer** check box and click **OK**.

e. Go to the **DEVELOPER** tab.

f. Place the cursor where you want to position the distribution statement and click one of the **Aa** icons to insert either rich text content or plain text content.

g. Select **Click here to enter text** and click **Properties**.

h. In the **Content Control Properties** dialog box, enter **Title** and **Tag**. The value of the tag must be same as that specified in the **MSOfficePlacement** tag value in the **Document Management configuration file**.




The value of the **datasetName** that matches the title value you specified in Word is placed as the distribution statement.

i. Click **OK**.

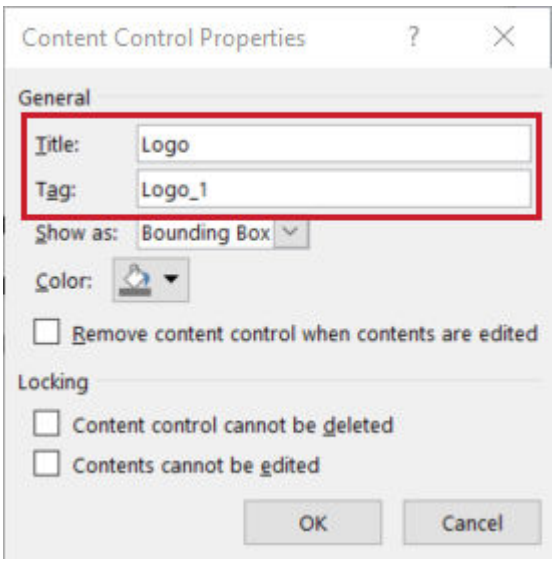
5. To insert logos:

a. Go to the **DEVELOPER** tab.

b. Click the **Picture Content Control** icon.

c. Select the image and click **Properties**.

d. In the **Content Control Properties** dialog box, enter **Title** and **Tag**. The value of the tag must be same as that specified in the **logoList** tag value in the **Document Management configuration file**.




The value of the **datasetName** that matches the title value you specified in Word is placed as the logo.

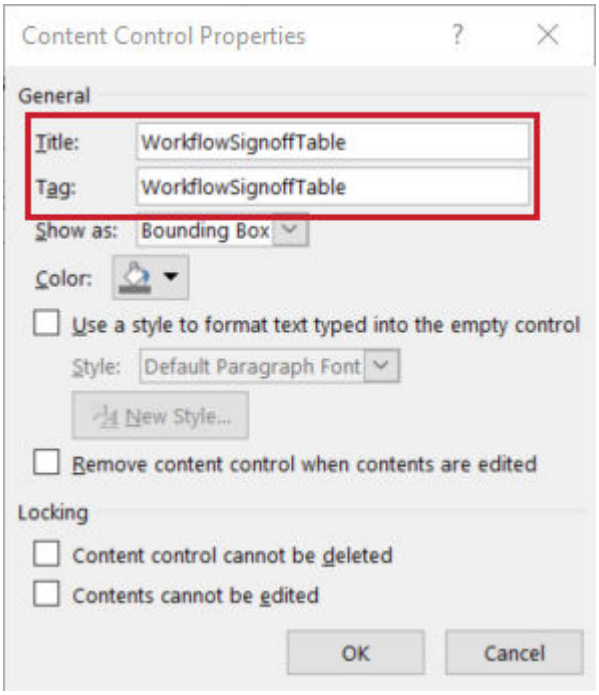
6. To insert the workflow signoff table:

a. Go to the **DEVELOPER** tab.

b. Place the cursor where you want to position the workflow signoff table and click one of the **Aa** icons to insert either rich text content or plain text content.

c. Select **Click here to enter text** and click **Properties**.

d. In the **Content Control Properties** dialog box, enter **Title** and **Tag** as **WorkflowSignoffTable**.




e. Click **OK**.

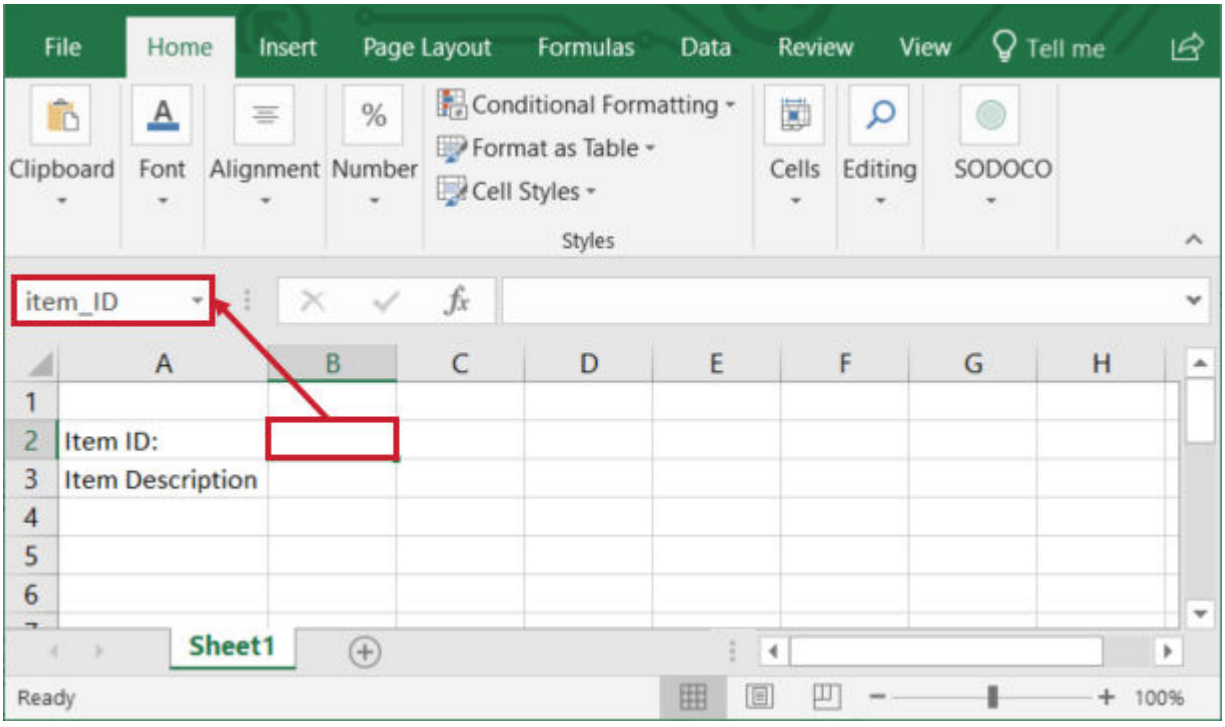
7. After creating the Word document, **attach it to the required document template revision item** and check in the template.

## Define the attributes of information in Microsoft Excel files

1. Check out the required template file.

2. Open an Excel file in which you want to include attributes.

3. Place the cursor at the cell where you want to enter a custom property and change the cell name to the property value that you specified while defining the logical object, for example, **item_ID**.




4. To add properties:

   a. Click **File→Info→Properties→Advanced Properties**.

   b. In the **Properties** dialog box, click **Custom**.

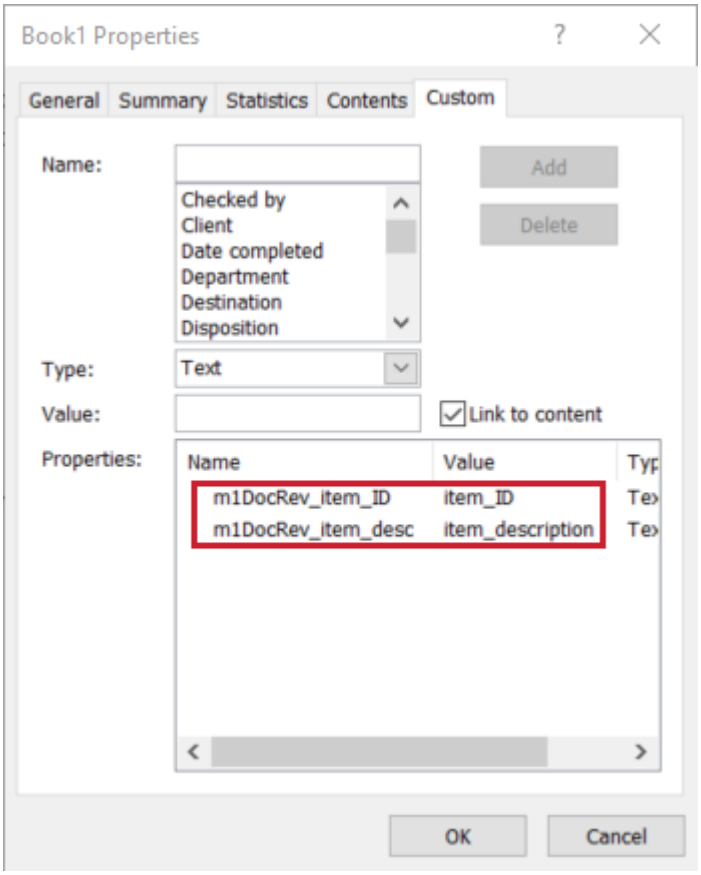
   c. Enter the ID of the member property in **Name**, for example, **item_ID**.

   d. Select **Type**.

   e. Enter the information in **Value**. As a default value is required, enter either a space or information that must be displayed in the file.

   f. Select the **Link to content** check box.

   g. Click **Add**.

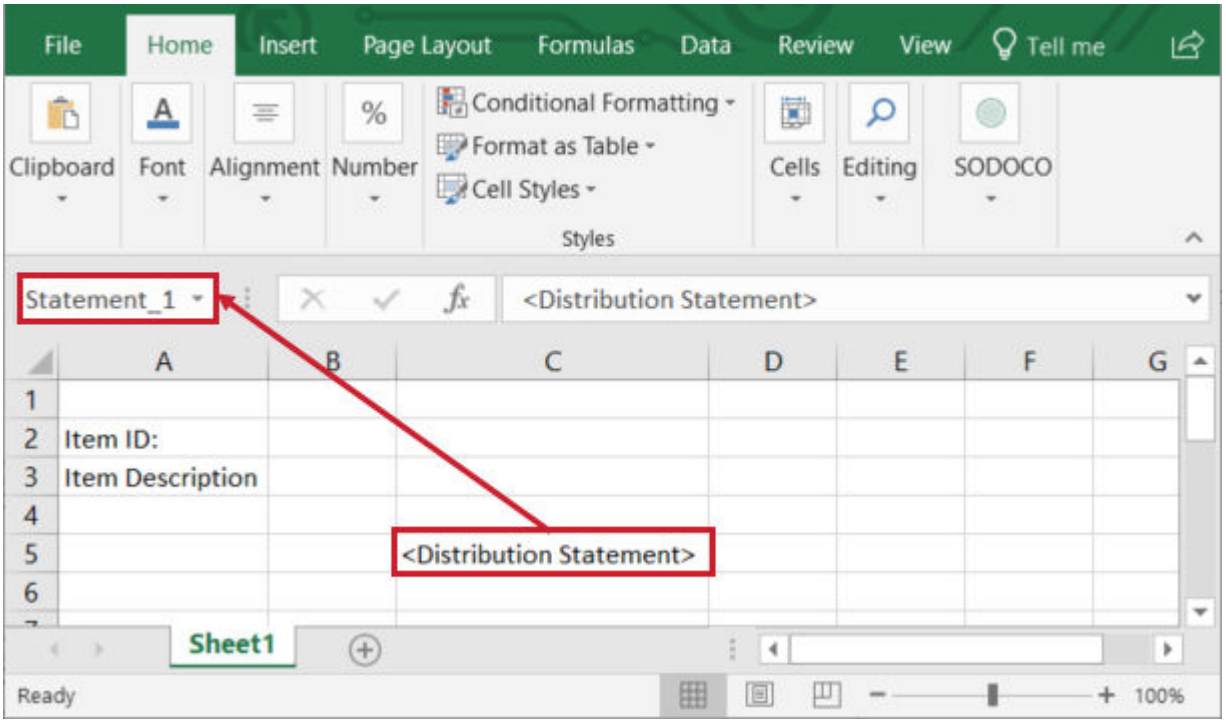



h. Add other required properties and finally click **OK**.

5. To insert distribution statements:

   a. Select a cell in which you want to insert the distribution statement.

   b. In the **Name Box**, rename the cell to the name of the distribution statement that you specified in the **MSOfficePlacement** tag value in the **Document Management configuration file**.

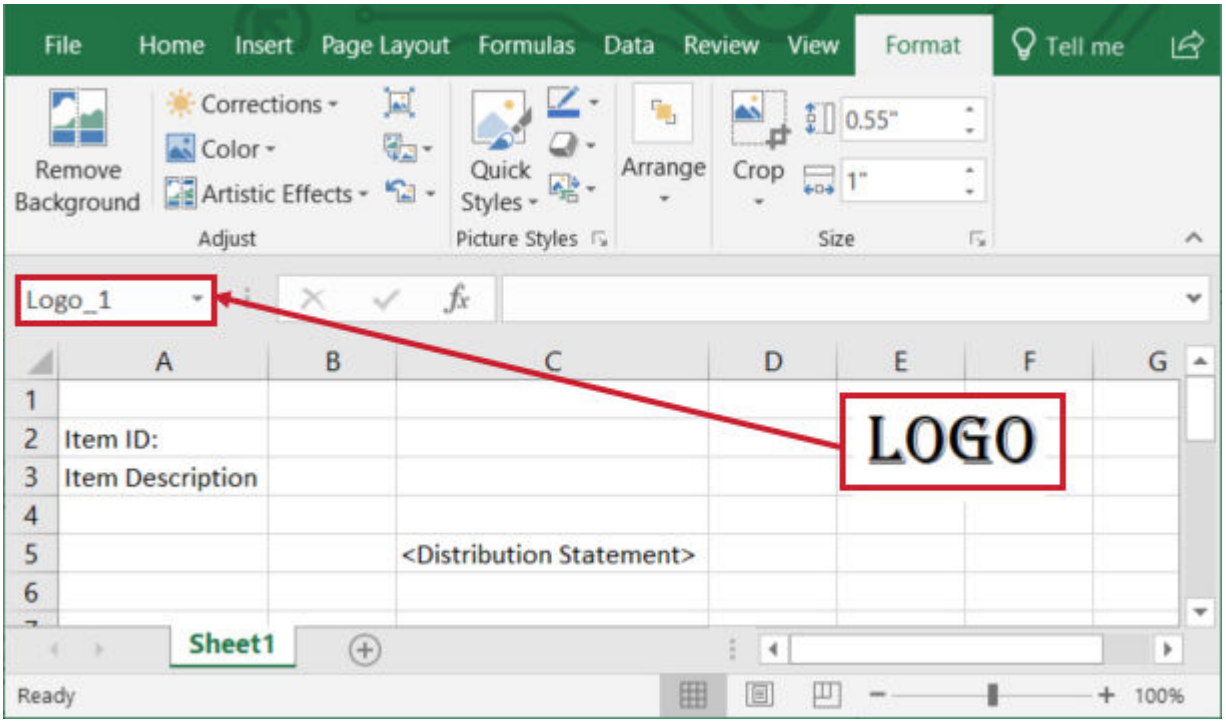



6. To insert logos:

   a. Click **Insert→Pictures**.

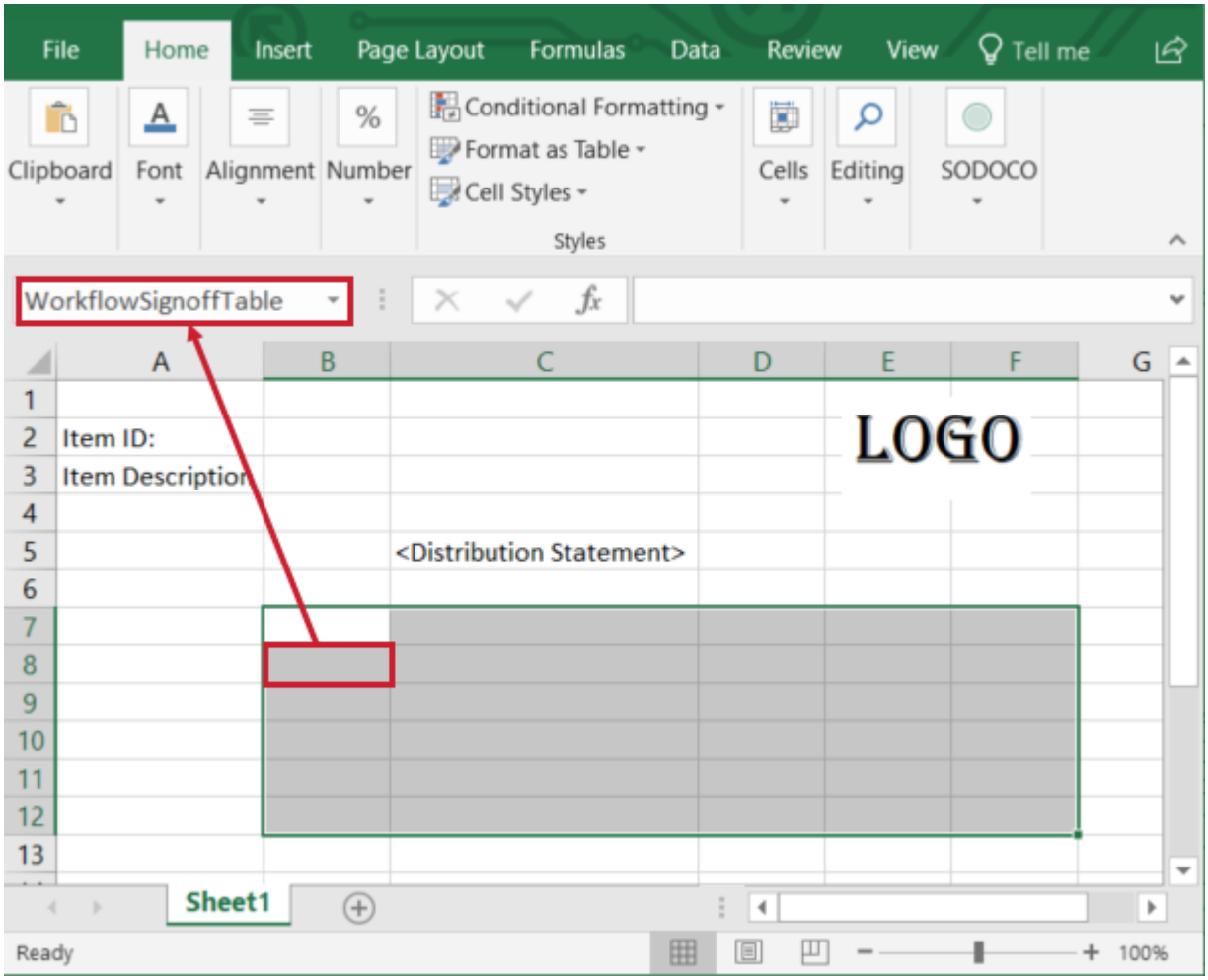
   b. Select the required logo and click **Insert**.

      Ensure that the name of the logo is the same as that specified in the **Document Management configuration file**.




7. To insert the workflow signoff table:

   - Select a cell where you want to position the workflow signoff table and change the name of the cell to **WorkflowSignoffTable**. For example, if you want the workflow signoff table to appear from cells **B7** to **F12**, you must select the cell **B8** and rename it to **WorkflowSignoffTable**. This is because the header of the workflow signoff table is placed in cells **B7** to **F12**.




8. After creating the Excel file, **attach it to the required document template revision item** and check in the template.

## Define the placement of attributes in Microsoft PowerPoint files

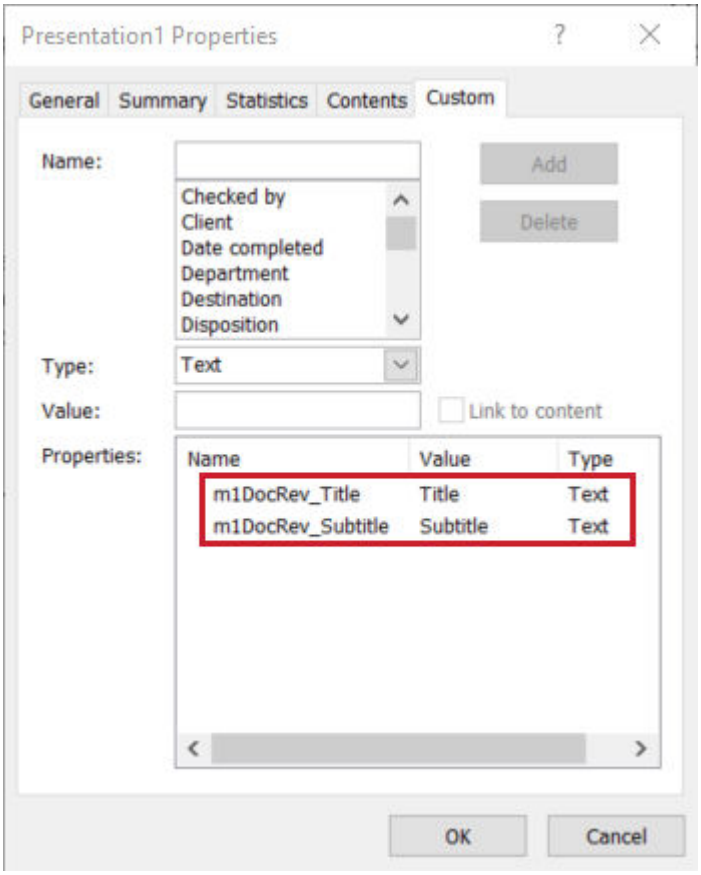
1. Check out the required template file.
2. Open a PowerPoint file in which you want to include attributes.
3. You must add each property that you defined for a logical object member as a custom property. To add each property:
   a. Click **File→Info→Properties→Advanced Properties**.
   b. In the **Properties** dialog box, click **Custom**.
   c. Enter the ID of the member property in **Name**.

d. Select **Type**.

e. Enter the information in **Value**. As a default value is required, enter either a space or information that must be displayed in the document.

f. Click **Add**.

g. Add other required properties and finally click **OK**.




4. To insert the newly created custom properties:

   a. Insert a text box for each property. You can also use the existing text boxes, if any.

   b. Click **Home→Arrange→Selection Pane**.

   c. In the **Selection** pane, rename each text box. The name of the text box must be same as the name of the property you added earlier.

      For example, rename the text box in which you the title to appear as **m1DocRev_Title**.

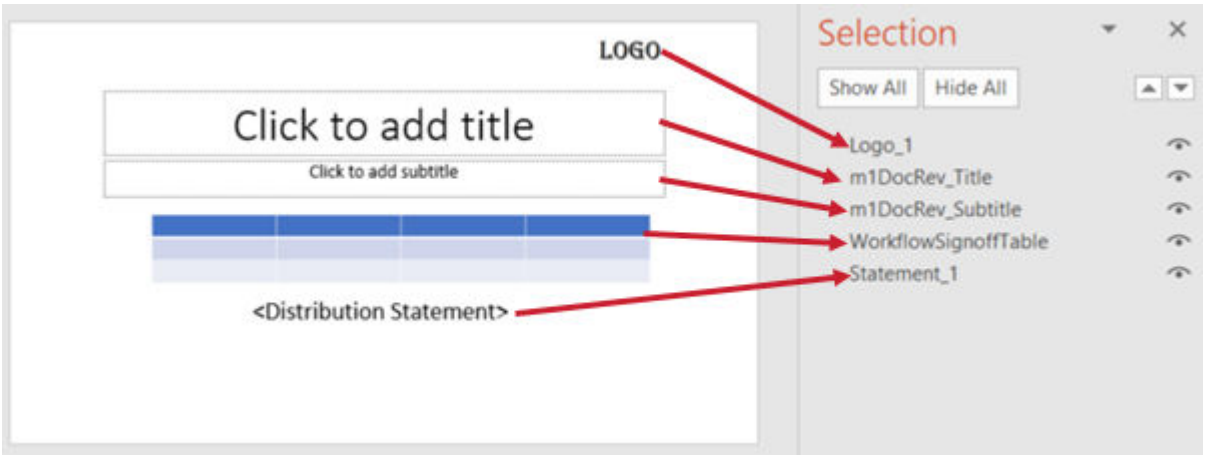
5. To place a logo, click **Insert→Pictures** and select an image.

Place the image where you wish the logo to be placed. In addition, rename the image to the value of the logo specified in the **Document Management configuration file**.

6. To place a distribution statement, insert a text box at the appropriate place and rename the text box so that its name is same as the tag value of the distribution statement in the **Document Management configuration file**.

7. To include the workflow signoff table, click **Insert→Table**.

   Place the table where you wish the workflow signoff table to be displayed and rename the table as **WorkflowSignOffTable**.




8. After creating the PowerPoint file, **attach it to the required document template revision item** and check in the template.

## Set up system stamps

1. In the Business Modeler Integrated Development Environment (BMIDE), click **BMIDE** > **New Model Element**.

2. In the **Add New Model Element** dialog box, select **System Stamp Configuration** in **Wizards** and click **Next**.

3. Enter the following information for the system stamp configuration:

| Field | Action |
|---|---|
| **Business Object** | Select **ItemRevision**. |
| **Condition** | Select the condition under which this stamp is applicable. |
| **Applies To** | Choose one of the following: |

| Field | Action |
| --- | --- |
| | • **Print**: To include the system stamps in a printed file.<br>• **Render**: To include the system stamps in a rendered file.<br>• **PDF_Control**: To include the system stamps in an existing PDF file when it is sent to a stamp workflow.<br>• **PrintAndRender**: To include the system stamps in both printed and rendered files. |
| **Include User Name?** | Select this check box to include the print requester's name in the printed files. |
| **Include Date And Time?** | Select this check box to include the print date and time in the printed files. |

4. Click **Next** and enter the following stamp information:

| Field | Action |
| --- | --- |
| **Properties** | Specify the properties that must be included in the printed or rendered files. You can optionally specify a prefix that must precede a property, for example, **Document Name**: <**object_name**>. |
| **User Stamp** | Type the text that you want to appear on the document, such as *Internal Distribution*. |
| **Watermark** | Type the text that you want to appear as a watermark, such as *Confidential*. |
| **MDS Template** | Select the MetaData Stamp template that defines how the stamps must be applied in the printed and rendered files.<br>To apply stamps in existing PDF files, select the PDF control command XML file.<br>To apply stamps during attribute synchronization, select the Document Management template revision for stamps. |

5. Click **Finish**.

The newly created system stamp configuration is listed under **Extensions**>**Document Management** > **System Stamp Configuration**.

## Relate logical objects with datasets

You can relate one or more logical objects to a dataset. To do so:

1. If you are an administrator, in Active Workspace, open and check out the document template revision containing the required dataset.

   Note:

   To apply the **Logical Object Type Relation** property to a document revision other than the document template, ensure that the style sheet of that document revision contains the *Fnd0LogicalObjectTypeRel* property added to it. If the property is already not added to the document revision, edit the style sheet for adding the property and then apply it.

2. In the **Files** section, change the view to **Table**.

3. In the **Files** section, select the dataset and double-click in the **Logical Object Type Relation** column of the table.

4. Select the logical object that you created for the document revision in **Logical Object Type Relation**.

5. Click **Edit > Check In**.

Note:

Modifications to a *.docx*, *.pptx*, or *.xlsx* file's body text (irrespective of the values of the mapped attributes) can get lost if a logical object is attached to it, that is, if the **Logical Object Type Relation** is defined for the MS WordX, MS PowerPointX, or MS ExcelX dataset. This can happen because the logical object attribute exchange is asynchronous by design. If you update the dataset file during this time, both processes (dispatcher and user) update two different versions of the same datasets.

To work around this problem, set the **Create Derived Visualization Data** IRDC to perform an **Optional render after checkin** (so that check-in is not stopped). Before the system sends the MS WordX, MS PowerPointX, or MS ExcelX dataset to the dispatcher for rendering, it triggers the logical object attribute exchange first synchronously.

## Set up a workflow to include attributes and stamps in Microsoft Office files

By default, the Teamcenter attributes, logos, distribution statements, and workflow signoff tables are included in Microsoft Word, Excel, and PowerPoint files when a user creates, checks in, revises, saves, or renders a document revision. However, a user can also send a file through an attribute exchange workflow to include the information. For this, you must set up a workflow task for the users by using the **DOCMGT-update-docprop-logicalobject** action handler.

To set up a workflow task:

1. Create a workflow process template.

a. In Workflow Designer, click **File→New Root Template**.

b. In the **New Root Template** dialog box, enter a name in **New Root Template Name**.

c. Select **Empty Template** in **Based On Root Template** and **Process** in **Template Type**.

d. Click **OK** to create a workflow process template.

2. Add a task to the workflow process template.

a. On the toolbar, click **Edit Mode** and then click **Task**.

b. In the process flow pane, double-click where you want to place the new task.

A new task appears with the default name **New Task #**. In the **Name** box, type a name for the task.

3. Link the task to its predecessor and successor tasks.

a. Click the task node you want to specify as the predecessor task.

> Caution:
>
> Do not click the title bar of the task node as this action drags the task node to a different location.

b. Drag your cursor to the task node that you want to specify as the successor task.

A link arrow follows the cursor as you drag. When your cursor moves over a task node, the node is highlighted.

c. Release the mouse button.

A link arrow connects the predecessor and successor nodes.

4. Configure the task.

a. Right-click the task and click **Task Properties**.

b. Click the **Task Handler** pane.

c. In **Task Action**, select **Complete**.

d. In **Action Handler**, select **DOCMGT-update-docprop-logicalobject** to create a workflow task to include stamps in Microsoft Word, Excel, and PowerPoint files during attribute exchange through logical objects.

e. Click **Create**.

## Understand the DOCMGT-update-docprop-logicalobject action handler

**Description** Updates the datasets (for example, MSWordX with a **.docx** extension or MSExcelX with a **.xlsx** extension) associated with the target item revisions with the latest attribute exchange data. Attribute exchange data can include Teamcenter properties, logos, distribution statements, and workflow sign off tables, if the target object is in a review task. Attributes are exchanged between Teamcenter and the files.

> Note:
>
> - The generic (logical object) attribute exchange currently supports Microsoft Word, Excel, and PowerPoint datasets only.
> - The Microsoft Word, Excel, and PowerPoint datasets must be **related to the logical objects** for the generic attribute exchange to occur.
> - Target item revisions must be valid and checked in.
>
>  The attribute exchange process from this workflow action handler bypasses the **Fnd0TriggerLOAttrExch** business object constant configuration.
> - Logos and distribution statements must be enabled based on their document configuration setting.
> - System stamp must be enabled for a business object revision and logical objects must be defined for its datasets.

> Caution:
>
> Document Management does not support multiple references in a dataset. Attempts to use such dataset can produce unpredictable results.

The update is synchronous.

**Syntax** **DOCMGT-update-docprop-logicalobject**

**Arguments** **-ignore_errors**

(Optional) Specifies that the current task can be continued on to the next task even if the current task fails. If this optional argument is not specified, the current task might stop on failures (including password protected file or dataset is checked out, or object cannot be saved), and might not continue to the next task.

**Placement** Place on the **Start** action of a **Task**.

Caution:

Do not place this handler on the **perform** action of the **perform-signoffs** task. Otherwise, this handler runs multiple times.

**Restrictions** Item revisions with attached datasets such as Microsoft Word, Excel, or PowerPoint must be included as targets of the workflow process.

## Export files along with their attributes to another site

You can export files attached to a document revision from one site to another site. Along with the files, their attributes also get exported. Before exporting the files,you must first **set up logical objects**. Document authors later attach files as datasets to a document revision by using **Logical Object Type Relation**. In this relation, they select the logical object that you set.

### Exporting files in a multisite environment

The export of the **Fnd0LogicalObjectTypeRel** relation is not supported by the **objio** export mechanism.

1. Import the logical object that you created at the source and the target sites.

   **admin_data_import -adminDataTypes=LogicalObjects -u=***username* **-p=***password* **-g=***group_name* **-inputPackage=***logical_object_path*

   For example:

   **admin_data_import -adminDataTypes=LogicalObjects -u=***Tc-admin-user* **-p=***password* **-g=***group* **-inputPackage=***C:\Development\Export\LO\LogicalObjectsExport106.zip*

2. Export the document revision from the source site.

   **data_share -u=***username* **-p=***password* **-f=offline_export -item_id=***document_rev_item_id* **-optionset -dir=***folder_name*

3. Import the document revision at the target site.

   **data_share -u=***username* **-p=***password* **-f=offline_export -optionset -dir=***folder_name*

### Exporting files in a site consolidated environment

1. Import the logical object that you created at the source and the target sites.

   **admin_data_import -adminDataTypes=LogicalObjects -u=***username* **-p=***password* **-g=***group_name* **-inputPackage=***logical_object_path*

For example:

**admin_data_import -adminDataTypes=LogicalObjects -u=***Tc-admin-user* **-p=***password* **-g=***group* **-inputPackage=***C:\Development\Export\LO\LogicalObjectsExport106.zip*

2. Export the document revision.

   **tcxml_export -u=***username* **-p=***password* **-f=***Site_Consolidation_xml_output* **-item=***document_rev_item_id* **-low_level -force_retraverse**

## Verify if attributes and stamps are included in template files automatically

To verify if you have configured generic attribute exchange correctly, you can perform one of the following:

- Start a workflow process by using the **DOCMGT-update-docprop-logicalobject** workflow action handler.

- Create, check in, revise, save as, or render a document revision with an attached MSWordX, MSExcelX, or MSPowerPointX file. Verify that attributes are automatically filled in the attached dataset. Also verify that other information such as logos, distribution statements, and workflow signoff tables are inserted into the Word document during the attribute exchange.

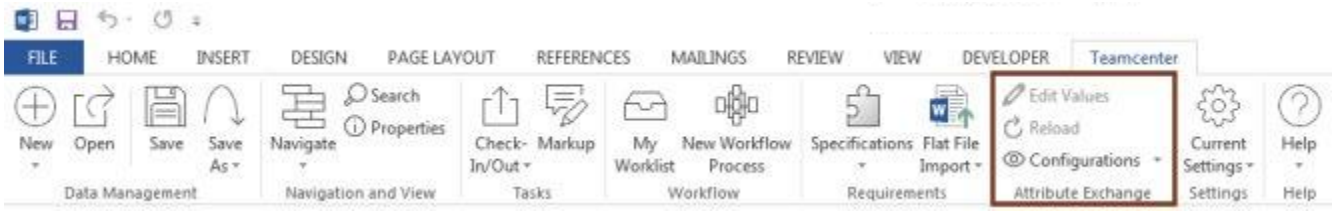
  Currently, on creating a document revision, the attributes are synchronized only if the attribute exchange direction is set as **Teamcenter to File**.

> Note:
>
> Logical object attribute exchange does not support **Teamcenter to File** with the display values of the properties. In stead, use the internal values of the properties. The properties include list of values (LOV).

# 16. Synchronizing attributes between Teamcenter and Microsoft Office file by using Client for Office

## Exchanging attributes between Office files and Teamcenter datasets

With Client for Office attribute exchange, you can synchronize property values between Teamcenter datasets and Microsoft Office files. In Word, Excel, and PowerPoint, these features are available through the **Attribute Exchange** button group on the **Teamcenter** tab of the Office ribbon.




Note:

- Attribute exchange is not available in Outlook.
- The property values are synchronized between Teamcenter and Microsoft PowerPoint using **Attribute Exchange**. However, PowerPoint does not allow you to use the synchronized properties in the content of the slides.
- Even if the attribute-level security is used while implementing Teamcenter, the properties that are used for attribute exchange are shown in the document. The attribute exchange overrides the attribute-level security.

You can map attributes on individual datasets and on item revision definition configurations (IRDCs).

- The mappings on a single dataset do not apply to any other datasets.
- When a Teamcenter item with a related dataset is created using an IRDC template with attribute mapping, the mapping is applied to the new dataset.

  If an item and its related dataset are created based on an IRDC template, and then the **Save As** command is used to create a new dataset from the original, the attribute mappings associated with the IRDC template can be applied to the duplicate dataset.

Mapped properties are updated automatically when you do any of the following:

- Open and check out a Teamcenter dataset.
- Save changes to the same dataset.
- Save changes as a new dataset.

You can map the following Teamcenter property types:

**Boolean**
**Date**
**Float**
**Integer**
**Text**

Note:

The following Teamcenter properties are not supported:

- System properties, such as **Date Created** and **Item ID**
- Properties for which *property constants* are defined in the Business Modeler IDE
- Properties with dynamic and cascading lists of values (LOVs)

## Setting up Attribute Exchange for Microsoft Word and Microsoft Excel on an Item Revision

Attribute Exchange is a way to link properties in a Microsoft Office application to Teamcenter attributes. To create the mapping, open the Microsoft Office dataset in its native application and create mappings as follows:

- File to Teamcenter (unidirectional) - Mapped property values go from the Microsoft Office application to Teamcenter.
- Teamcenter to File (unidirectional) - Mapped Teamcenter attribute values go to the Microsoft Office application property values.
- Two way (bidirectional) - Mapped Teamcenter attribute values go to and come from the mapped Microsoft Office application property values depending on where they were last changed.

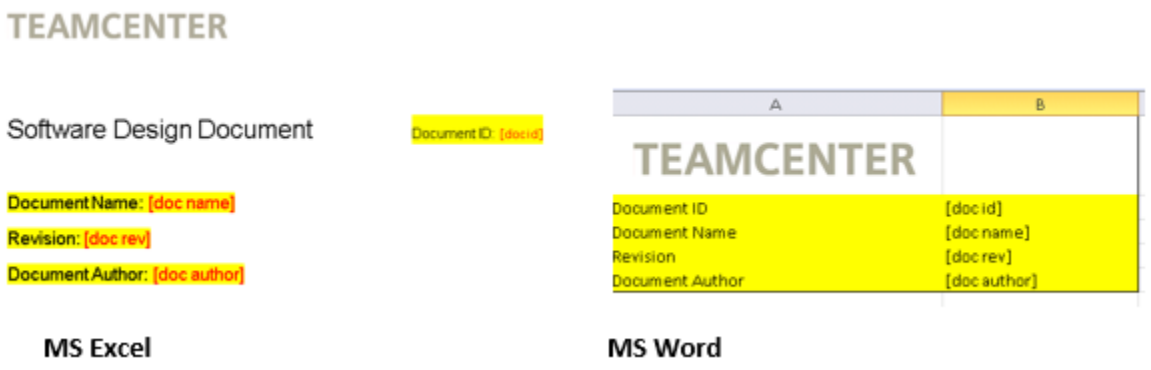
Because datasets are Teamcenter objects and can have attributes, the mapping can be directly to the Microsoft Office dataset attributes. However, this is impractical for most users as they want to pull existing Teamcenter attributes which exist on item revisions. When you set up the mapping to an item revision, the Microsoft Office dataset is usually attached to the item revision as the mapping is dependent on the relation path to the revision. Additionally, you want to be able to create a custom

item and have the mapped Microsoft Office dataset attached so that you do not have to map to the dataset each time. To do this, you must use the Item Revision Definition Configuration (IRDC).

In this example, we set up attribute exchange with a Microsoft Word dataset or a Microsoft Excel dataset to an item or item revision or their child objects such as a document or document revision, or any custom item or custom item revision. The datasets are created from an existing Word document or an Excel workbook. The task is to automatically populate the following fields with values from the document or document revision or any custom item or custom item revision. The office document properties and the Teamcenter properties they map to are as follows:

| Office document properties | Mapped to Teamcenter properties |
|---|---|
| Document ID | Document's item_id (display name as **ID**) |
| Document Name | DocumentRevision's object_name (display name as **Name**) |
| Revision | DocumentRevision's item_revision_id (display name as **Revision**) |
| Document Author | User's owning_user (display name as **Name**) |

- Document ID: [doc id]
- Document Name: [doc name]
- Revision: [doc rev]
- Document Author: [doc author]

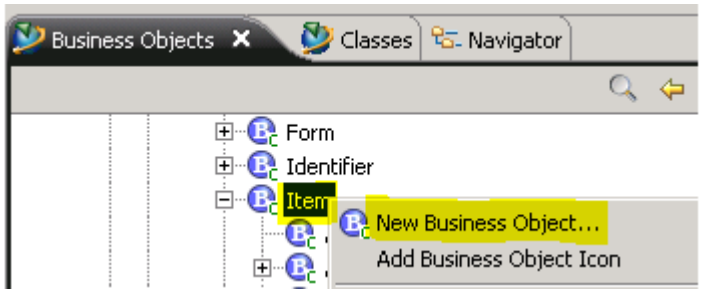



### Step 1: Create the custom item or custom item revision and the properties on the revision

To map to custom properties on a custom item or custom item revision, you must first create the business objects. If you use COTS document or documentRevision as example above, skip the steps below.

1. Open BMIDE and search for the item.

2. Right-click the item and select **New Business Object**.




3. Set the **Name** to <*prefix*>**CustomItem**.

4. Set the **Display Name** to **CustomItem**.

5. Click **Next**.

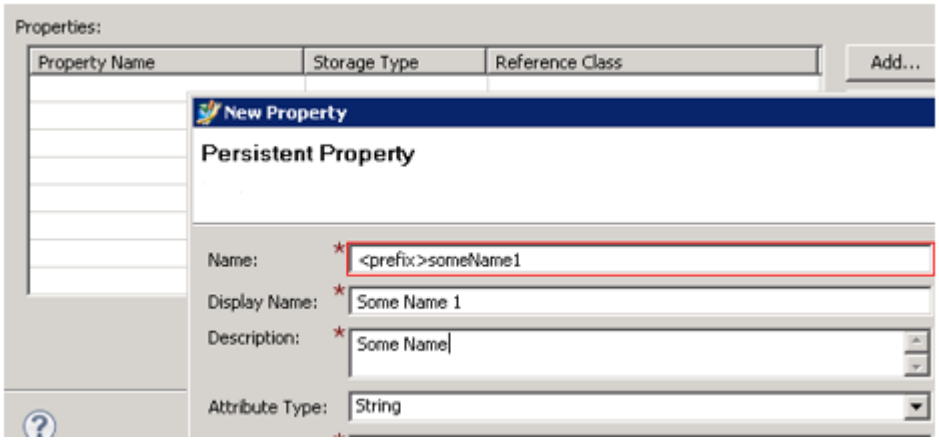
6. Set the **Display Name** to **CustomItem Revision**.


7. Click the **Add** button in the **Properties** section.

8. Create a property named <*prefix*>**someName1**.

9. Set the **Display Name** to **Some Name 1**.

10. Click **Finish**.



11. Repeat the property creation for the following:

    - Create a property named <*prefix*>**someName2**. Set its **Display Name** as **Some Name 2**.
    - Create a property named <*prefix*>**someName3**. Set its **Display Name** as **Some Name 3**.

    > Note:
    >
    > Similarly, you can create other properties as required.

12. Save and deploy the BMIDE template. See the Teamcenter guide on the deployment process.

### Step 2: Create a sample of the document or document revision (or customItem or customRevision) and attach the respective dataset

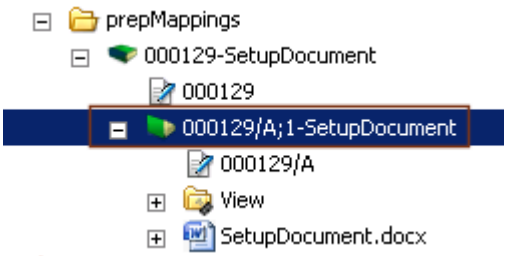
To map to properties on a document or document revision or custom item or custom item revision, let us create sample objects to attach the Microsoft Office dataset. This shows how to map the Microsoft Office properties to Teamcenter attributes.

1. In the rich client, in My Teamcenter, create a folder named **prepMappings** under the **Newstuff** folder.

2. Select the **prepMappings** folder.


3. Create a document named **SetupDocument** (or create the CustomItem). Expand the item so that the item revision is displayed.

4. Drag and drop the Microsoft Office file and onto the respective item revision to create the Microsoft Office dataset (MSWordX or MSExcelX).




   The MSWordX dataset is related to the documentRevision with the **Attaches** (**TC_Attaches**) relation. (other relation can be used instead also).

If you use COTS document/documentRevision in the example above, skip the following steps.

5. If you use customItem or customItemRevision, select the customItemRevision. Right-click the revision and select **Edit Properties**. Checkout if necessary. Update the properties (Some Name 1, Some Name 2, ..., Some Name N) with some values.

6. Click **Save** and then click **Save and Check-In**.

### Step 3: Open the dataset in the Teamcenter Client for Microsoft Office application and set the mappings

The mapping configuration is created in the Client for Office Ribbon on the Attribute Exchange pane. Microsoft Word uses fields to display the mapped attributes, while Microsoft Excel uses named references to display the mapped attributes.

In Microsoft Word:

1. In Microsoft Word, select the Teamcenter ribbon (installed with the Teamcenter Client for Microsoft Office).

2. Select **Current Settings →Login**.

3. Authenticate using your Teamcenter credentials.

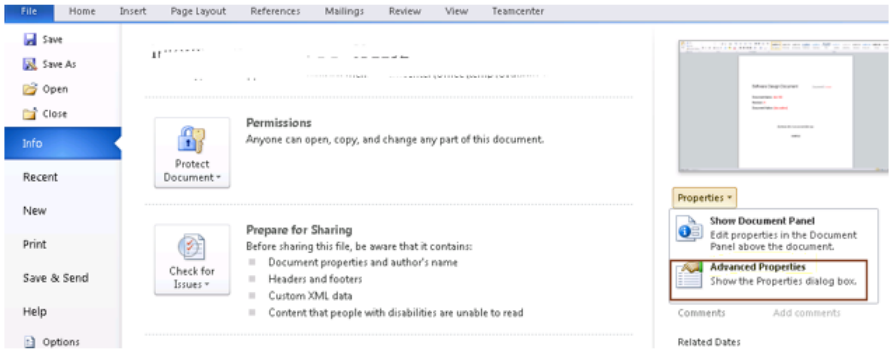
4. Click **Navigate→Browse**.

5. In the navigation pane, browse to the Microsoft Word dataset attached to the SetupDocument Revision (or create the CustomItem Revision).

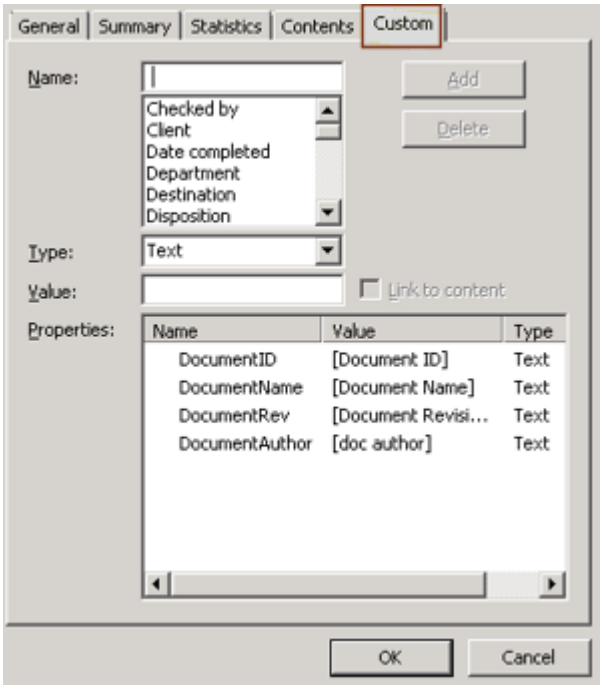
6. Right-click the dataset and select **Open** and **Check-Out File**.

7. Configure the Office Document Properties on the MSWord file.

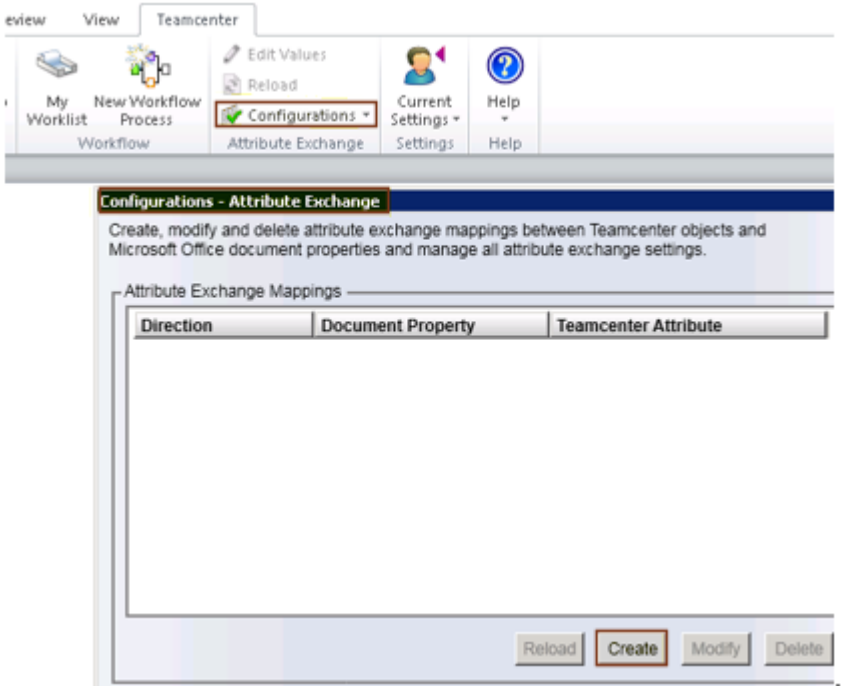
8. Choose **File→Info→Properties→Advanced Properties**.



9. In the Advanced Properties, select **Custom** tab and add the office properties such as DocumentID, DocumentName, SomeName1, and SomeName2. Click **OK**.

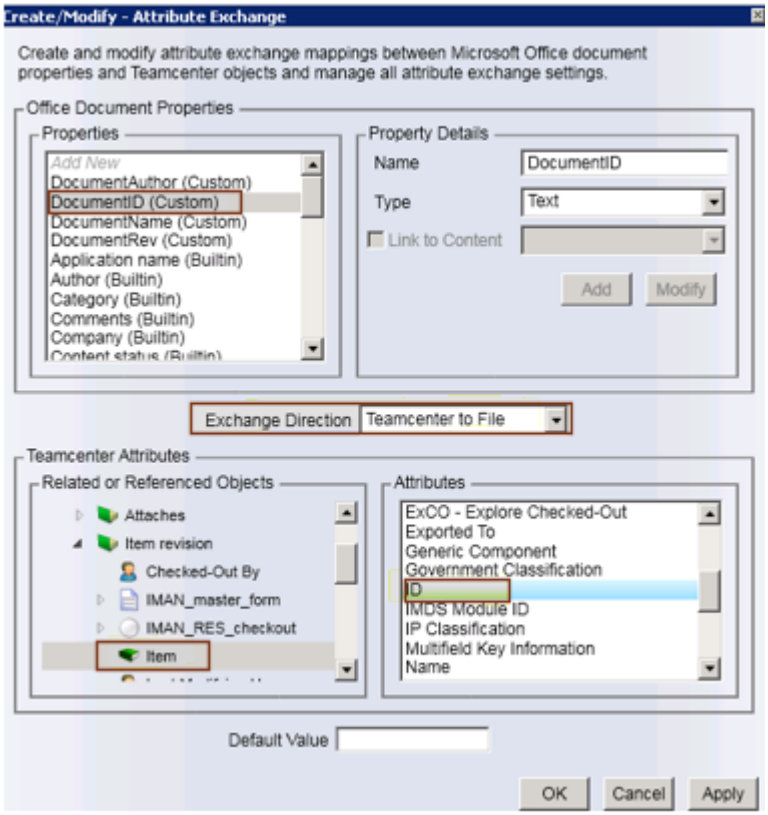



10. To configure the attribute exchange mapping, on the Teamcenter ribbon, select **Configurations→Create** in the **Attribute Exchange** pane.



11. To pull Teamcenter attributes into Microsoft Word, set the **Exchange Direction** to **Teamcenter to File**.

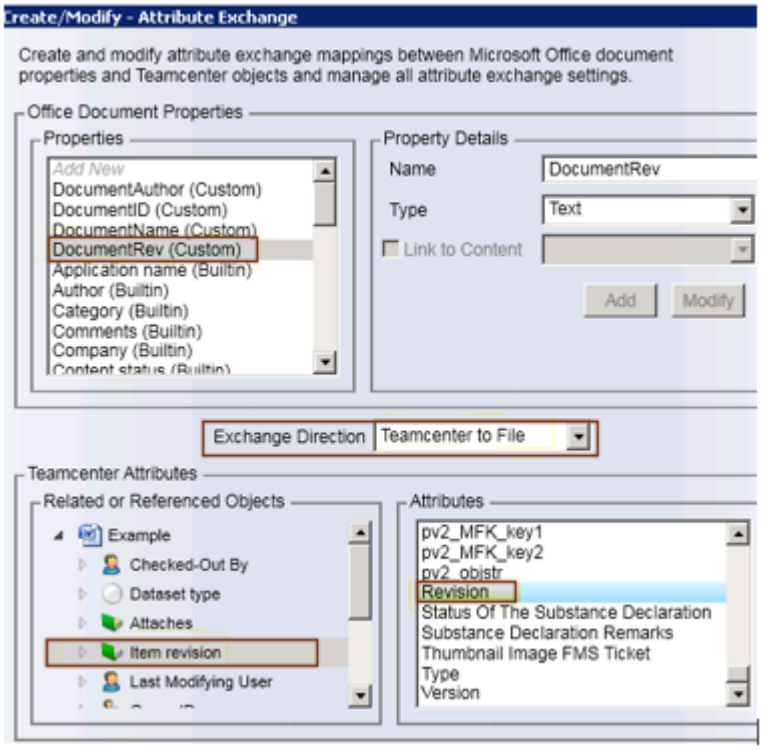
12. In the top section, configure the Microsoft Office document properties, and in the bottom section, configure the Teamcenter attributes.




13. In the Office Document Properties, select the property for the attribute exchange with Teamcenter.

14. Now to link the Teamcenter Attribute, notice that the document ID property is not in the list of Attributes initially and this is because the **Related or Referenced Objects** section has the Word Dataset selected. In example above, expand the Item revision and select the Item. Scroll through the **Attributes** list and select the attribute (for example, ID).

15. To view the Teamcenter Attributes on the SetupDocument Revision, select item revision in the **Related or Referenced Objects** section.




16. Click **Apply**. Repeat step 13 to 16 for linking other attributes.

17. After all of the mappings are created, click **OK**.

18. To verify the Teamcenter attribute mapping, select **Configurations→All Configurations** in the **Attribute Exchange** pane and click the **Close** button.

19. In the **Attribute Exchange** pane, click **Reload**.

    The attributes that are refreshed to the Microsoft Word dataset are displayed.

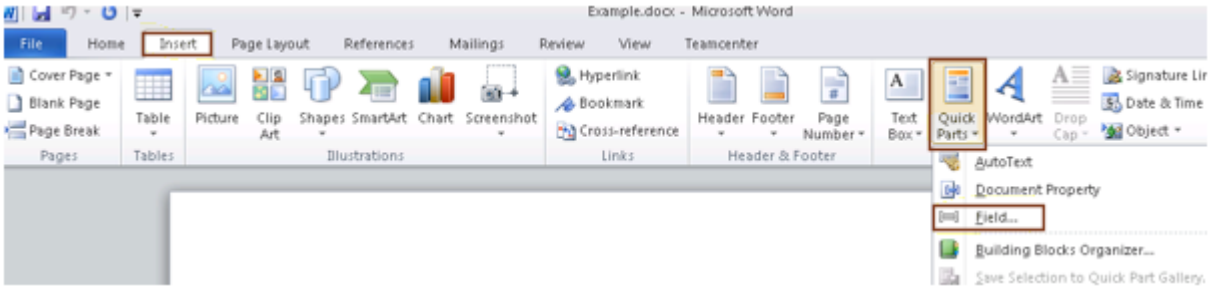
20. Click the **OK** button.

21. Verify that the Microsoft properties now have the same values by viewing the Microsoft Word Advanced Properties on the **Custom** tab. (**File** >**Info** >**Properties** > **Advanced Properties** >**Custom** tab)

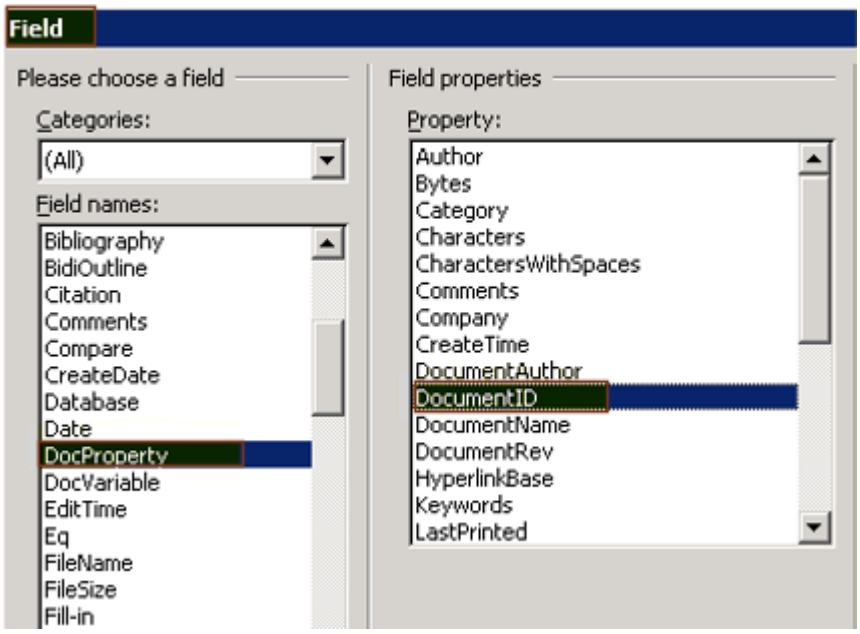
22. Click the **OK** button.

23. To insert the value of an office property in the Word document, highlight the location in the Word document where you want the Document ID attribute to be displayed for example.

24. Select the **Insert** ribbon then **Quick Parts→Field**.




25. Select the **Field Name** as **DocProperty** and **Property** as **DocumentID**.




26. Click **OK**.

    The value inserted because of the mapping is displayed.

27. Similarly repeat steps 24 to 26 to create other properties.

28. On the Teamcenter ribbon, click the **Save** button in the **Data Management** pane.

29. When presented with the **Confirm Check-In** dialog box, click **Yes**.

30. Close Microsoft Word.

In Microsoft Excel:

1. Open Microsoft Excel and select the Teamcenter ribbon (installed with the Teamcenter Client for Microsoft Office).

2. Select **Current Settings→Login**.

3. Authenticate using your Teamcenter credentials.

4. Click **Navigate→Browse**.

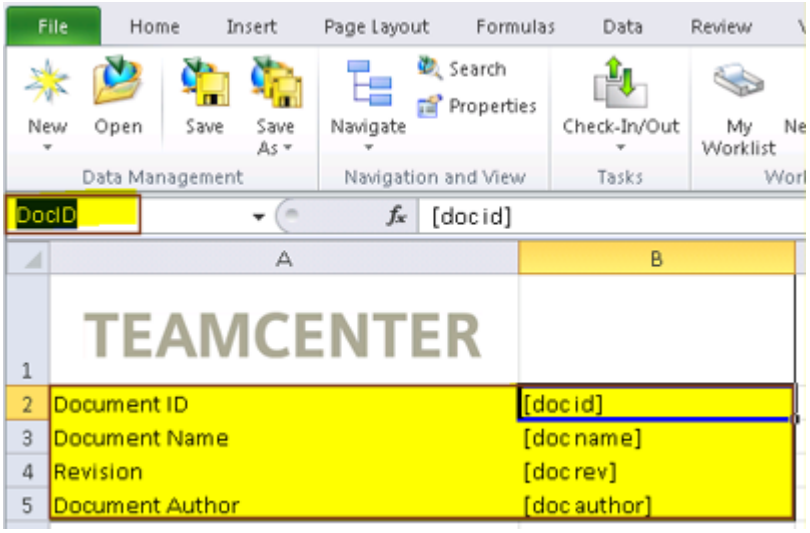
5. In the navigation pane, browse to the Microsoft Excel dataset attached to the SetupDocument Revision (or create the SetupCustomItem Revision).

6. Right-click the dataset and select **Open and Check-Out File**.

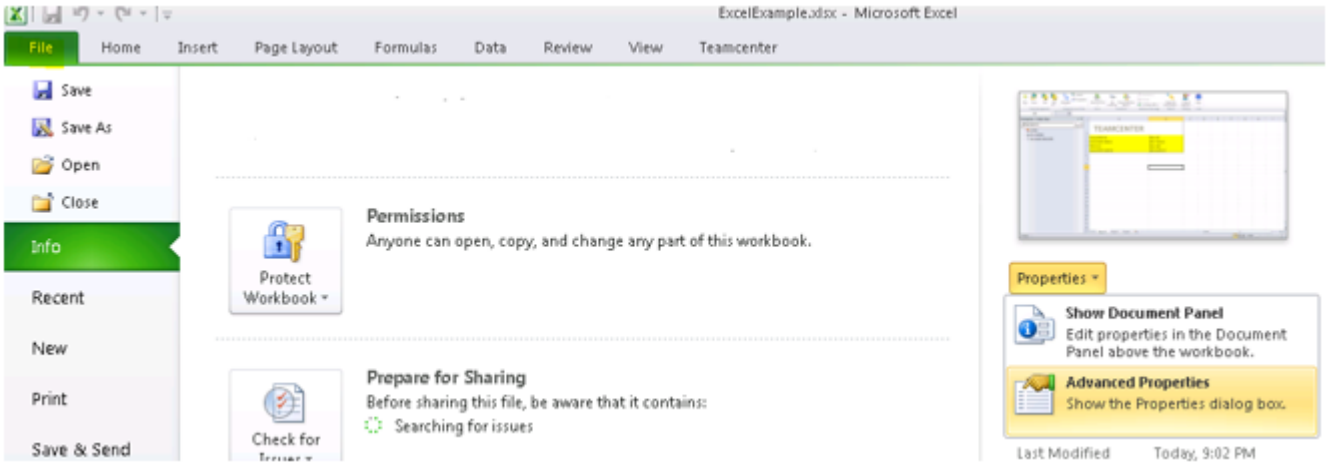
7. Configure the Office Document Properties on the MSWord file.

8. To link an office property to a cell on the spreadsheet, you must rename that cell.

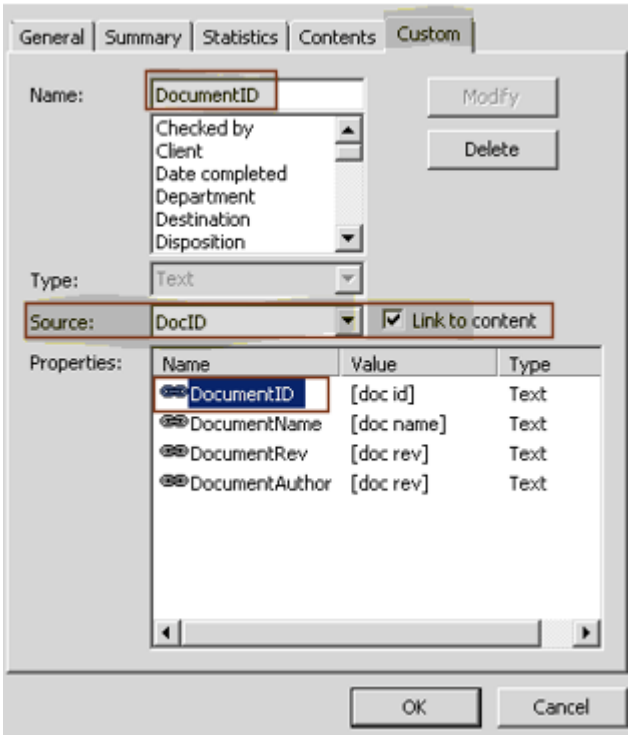
   For example, the B2 cell is renamed to DocID. Similarly, repeat the same for the other properties.




9. Select **File→Info→Properties→Advanced Properties** from the menu.

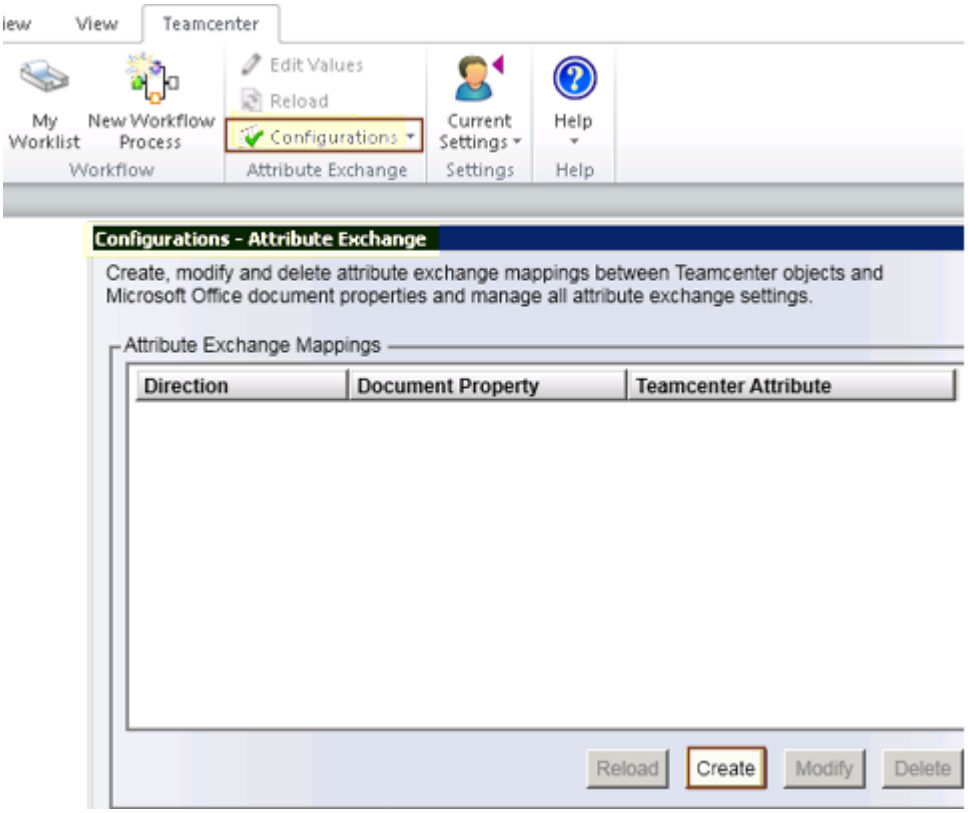



10. In the Advanced Properties, select **Custom** tab and add the office properties such as DocumentID, DocumentName, SomeName1, and SomeName2.



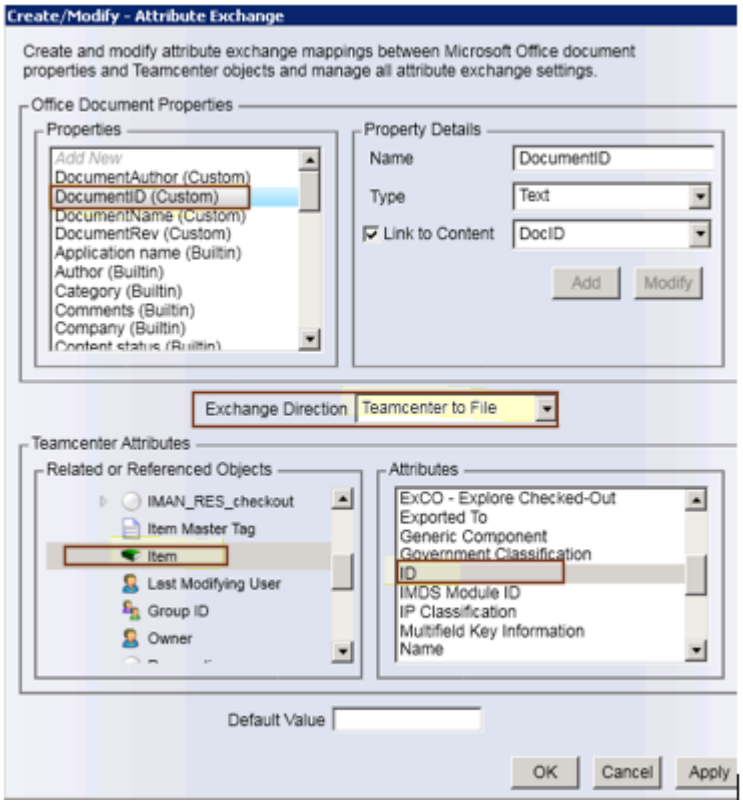

For example, add **DocumentID** in the **Name** field. Check the box **Link to content**. Scroll the **Source** field and select **DocID** (that was renamed in step 9). Repeat the same for other properties. Click **OK**.

11. To configure the attribute exchange mapping, on the Teamcenter ribbon, select **Configurations→Create** in the **Attribute Exchange** pane.




12. To pull Teamcenter Attributes into Microsoft Excel, set the **Exchange Direction** to **Teamcenter to File**.

13. The top section is to configure the Microsoft Office document properties and the bottom one is for the Teamcenter attributes.




14. In the Office Document Properties, select the property for the attribute exchange with Teamcenter.

15. Now to link the Teamcenter Attribute, notice that the document ID property is not in the list of **Attributes** initially and this is because the **Related or Referenced Objects** section has the Excel Dataset selected. In example above, expand the item revision and select the Item. Scroll through the **Attributes** list and select the attribute (for example, ID).

16. To view the Teamcenter Attributes on the SetupDocument Revision, select item revision in the **Related or Referenced Objects** section.

17. Click the **Apply** button. Repeat step 13 - 17 for linking other attributes such as DocumentName and DocumentRev.

18. To verify the Teamcenter Attribute mapping, select **Configurations→All Configurations** in the **Attribute Exchange** pane and click the **Close** button.

19. In the **Attribute Exchange** pane, click **Reload**.

    The Teamcenter attribute values are immediately updated in the Excel Workbook.

20. Click the **OK** button.

21. Verify that the Microsoft properties now have the same values by viewing the Microsoft Excel Advanced Properties on the **Custom** tab. (**File** >**Info** >**Properties** >**Advanced Properties** >**Custom** tab)

    Click **OK**.

22. On the Teamcenter ribbon, click the **Save** button in the **Data Management** pane.

23. When presented with the **Confirm Check-In** dialog, select **Yes**.

24. Close Microsoft Excel.

### Step 4: Create document templates to store the mapped datasets

To create the document or custom items with the configured datasets, you need to configure IRDC in BMIDE. First, in the Teamcenter rich client you must create document templates to store the configured datasets. This creation of document templates is necessary for the IRDC configuration.

To create the document templates:

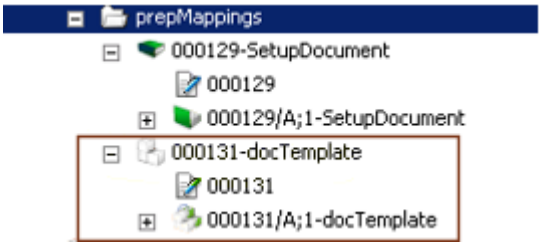
1. Select the **prepMappings** folder.

2. Select **File →New→Item** from the menu.

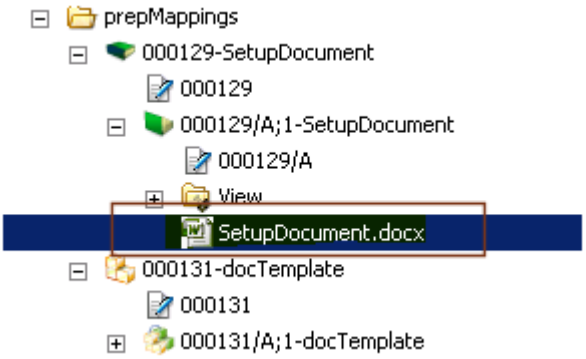
3. Select the document template and click **Next**.

4. Name the document template as **docTemplate** and click **Finish**.

5. Expand the **docTemplate** so that you can see the revision.




6. Select the MSWordX dataset attached to the SetupDocument revision and cut it by selecting **Edit→Cut**.




7. Select the docItemTemplate revision and select **Edit→Paste**.

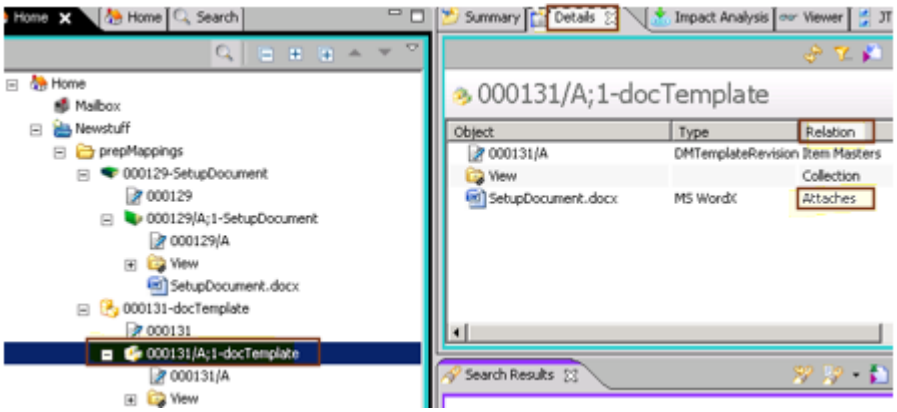
8. Select the **SetupDocument**.

9. Select **Edit→Delete** to remove it from Teamcenter.

To configure the document templates and datasets:

The document template revision and attached dataset must be in a released status. Also the dataset must be connected via a relation Attaches (TC_Attaches) in this example. Select the docTemplate revision then select the **Details** tab.

1. Select the docTemplate revision and select the **Details** tab.




2. If the Relation for the MSWordX Dataset is not Attaches, cut the MSWordX dataset and select the docTemplate Revision and select **Edit→ Paste Special**. Scroll down to select the **Attaches** relation.

3. Select docITemplate revision and the MSWordX dataset (or MSExcel dataset). You can press the CTRL key and click to select.

4. Select **File →New→Workflow Process**.

5. Select **TCM Release Process** as the **Process Template**.

6. Exit the Teamcenter rich client.

### Step 5: Create IRDC in BMIDE

The final step is to create an IRDC for the custom items and map them to their respective document templates in the Teamcenter rich client. After deploying the new template they should be available for use.

1. Go to the **Extension** section in BMIDE.

2. Expand your project then expand Document Management.

To set up the IRDC for the docTemplate revision:

1. Right-click **IRDC** and select **New IRDC**.

2. Name the IRDC docTemplateIRDC (or customTemplateIRDC) for example.

3. Set the **Applies to Business Object** to **DocumentRevision**.

4. If you use your CustomRevision, then select CustomRevision.

5. Ensure that the **Condition** is set to **isTrue**.

   > Note:
   >
   > You can create a new condition for DocumentRevision (or CustomRevision) and use it.

6. Set the **Create Template** to **docTemplate**. (This selection is done via BMIDE connecting to Teamcenter and presenting choices.)

7. Click **Next**.

8. On the **IRDC Dataset Criteria** page, click **Add** in the **Source Dataset** area.

9. Set the **Source Dataset** to **MSWordX** or **MSExcelX**.

10. Set the **Item Revision Relation** to **TC_Attaches**.

11. Click **Finish**.

12. Click **Next**.

    The IRDC Dataset Naming page is displayed.

13. Click **Next**.

The IRDC Checkin page is displayed.

14. Click **Next**.

    The IRDC Rules page is displayed.

15. Set **Delete Data file** to **No**.

16. If you need to add any deep copy rules, you can add them here.

17. Click **Finish**.

18. Click **Finish**.

19. Click **Next**.

    The **Enter Markup Information** page is displayed.

20. Click **Finish**.

21. Save and deploy the BMIDE template.

### Step 6: Verify

Test by creating a new document (or new customItem). If testing with your CustItem, populate the custom properties. Go to Client for Office. Browse to and open or checkout the MSWordX dataset (or MSExcelX dataset) attached to the DocumentRevision. The reload should occur automatically to display the objects.


**MS Word**

OR


**MS Excel**

## Directional settings for mapping attributes

The attribute map for a dataset includes a setting for the direction of the exchange.

- **File to Teamcenter**

  The value of the Office **Comments** property is copied to the Teamcenter **Description** attribute.

- **Teamcenter to File**

  The system copies the Teamcenter **Name** attribute value, where **Name** is the owner of the Teamcenter object, to the **Author** document property. The value is copied to the document.

- **Two Way**

  When you save a document to Teamcenter, the system copies the **Title** document property value to the Teamcenter **Name** attribute, where **Name** is the object name. The value is copied to Teamcenter. When you open and check out a Teamcenter dataset, the system copies the Teamcenter **Name** attribute value, where **Name** is the object name, to the **Title** document property. The value is copied to the document.

Caution:

Properties with dynamic and cascading lists of values (LOVs) are not supported.

You can view the attribute mapping for a dataset and, if needed, modify the mapping direction.

## Updating mapped properties

Several events automatically trigger attribute exchange between Teamcenter dataset objects and Microsoft Office files. The attribute or document properties exchanged depend on the configured exchange direction.

- Open and check out

  When a Teamcenter dataset is opened and checked out, the configured attribute exchange occurs. This exchange happens only on the mappings whose direction is **Teamcenter to File** and **Two Way**. A default value is used for the exchange if the following two items apply:

  - The Teamcenter dataset or its attribute does not exist on the server or the attribute value is null.
  - A default value is specified.

  Note:

  Users are informed if any errors occurred during the exchange.

- Save

  When a Teamcenter dataset is saved, the configured attribute exchange occurs. This exchange happens only on the mappings whose direction is **File to Teamcenter** and **Two Way**.

  If you delete the value for a custom Office document property that is defined for attribute exchange, the default property value is used during the save operation, if a default is specified.

  Note:

  Users are informed if any errors occur during the exchange.

- Save as

  When a Teamcenter dataset is saved using the **Save As** command, dataset creation is initiated. When the new dataset is created, if attribute exchange mappings are configured on the original dataset, they are copied to the new dataset. If the dataset is checked out, attribute exchange is performed both ways: **File to Teamcenter** and **Teamcenter to File**.

Note:

When creating a new dataset from the **Teamcenter** tab using the **Save As** command, you have the options of creating a new item where you want to save the dataset, and checking out the new item revision and the dataset.

If you check out the item revision and the related dataset file, the mappings configured for an IRDC controlled item are *not* associated with the checked out dataset. Therefore, if an item and its related dataset were originally created based on an item revision document control (IRDC) template, and if you want to retain the attribute mapping for the newly created dataset, you must:

1. Click **No** in the **Confirm Check-Out** dialog box.

2. Perform the check out operation separately from the create operation the first time you edit the object.

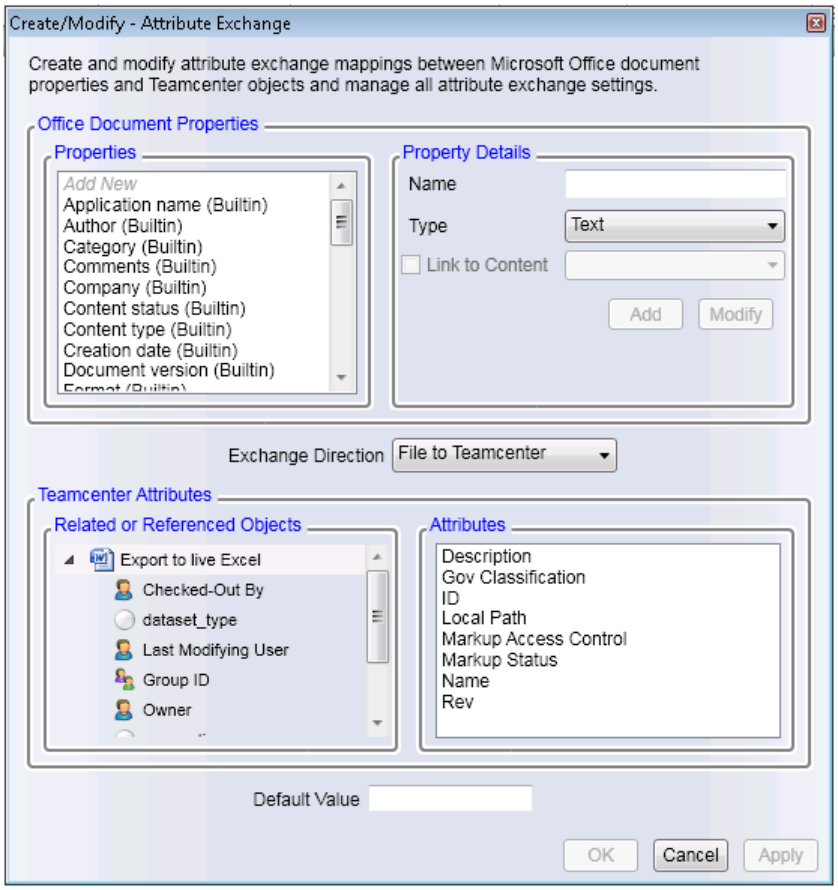
Warning:

After an attribute's value is synchronized between Teamcenter and Microsoft Office, and later it is deleted in Teamcenter, the null value cannot be synchronized to Microsoft Office, so the attribute continues to have the previous value.

## Map Office document properties to Teamcenter

1. With an **MS Office** dataset selected in Teamcenter, choose **Open and Check Out**.

2. On the **Teamcenter** tab in the **Attribute Exchange** button group, click **Configurations→Create** to open the **Create/Modify - Attribute Exchange** dialog box.




3. Click **Create**.

   The **Create/Modify - Attribute Exchange** dialog box opens.

   > Note:
   >
   > You can open the **Create/Modify - Attribute Exchange** dialog box from the **Attribute Exchange** button group by clicking the list button next to **Configurations** and clicking **Create**.

4. In the **Properties** box, select an existing Office document property.

   The values for **Name** and **Type** appear in the **Property Details** pane.

   You can also click the **Link to Content** box and select the linked content from the list.

   > Note:
   >
   > When performing attribute exchange, if you set the direction of the exchange as **File to Teamcenter**, and you select a property to which you have write access, the exchange is

allowed. However, if you select a property that the system uses to control the status of the object, this can alter the status of the target object and may render the object unusable or corrupt. Therefore, when configuring attribute exchange from **File to Teamcenter**, do not select a property of the target Teamcenter object that is used to determine the processing status of the target object.

5. Click **Modify** to apply any changes you made to the property details.

   You can also create a new custom document property by adding a unique name and selecting a type.

6. Click **Add** to create a new custom document property.

7. Select the exchange direction from the list.

   - **File to Teamcenter**

     The document property value is copied to Teamcenter. The **Related or Referenced Objects** list only allows you to navigate one level to find objects attributes.

   - **Teamcenter to File**

     The Teamcenter attribute value is copied to the document. The **Related or Referenced Objects** list allows you to navigate two levels to find objects attributes.

   - **Two Way**

     The mapped property and attribute are copied both ways. The **Related or Referenced Objects** list only allows you to navigate one level to find objects attributes.

   Caution:

   Properties with dynamic and cascading lists of values (LOVs) are not supported.

8. In the **Related or Reference Objects** box, select a Teamcenter object.

   The attributes for the objects appear in the **Attributes** box. Only the attributes that match the property type display.

   Example:

   If the property type is **Date**, only Teamcenter attributes defined as date type appear.

> Note:
>
> When you select a relation and multiple objects of the same type are found by this relation, none of these objects can be selected for exchange. The valid related objects are the ones that are uniquely identified by relation and object type.

You can also enter a default value for the mapping.

9. Click **Apply** to add the attribute exchange mapping and leave the **Create/Modify - Attribute Exchange** dialog box open.

   The mapping appears as a new row in the **Configurations - Attribute Exchange** dialog box.

10. Click **OK** to add the attribute exchange mapping and close the **Create/Modify - Attribute Exchange** dialog box.

    The mapping appears as a new row in the **Configurations - Attribute Exchange** dialog box.

## View mapping configuration details

1. Open and check out a Teamcenter dataset.

2. Click the **Teamcenter** tab.

3. In the **Attribute Exchange** button group, click **Configurations**.

   The system displays the **Configurations - Attribute Exchange** dialog box. Configuration mappings display IRDC template mappings and locally configured mappings.

   > Note:
   >
   > Optionally, you can open the **Configurations - Attribute Exchange** dialog box from the **Attribute Exchange** button group by clicking the list button next to **Configurations** and clicking **All Configurations**.

4. Select from the list to modify, delete, or perform exchange (reload) after necessary validations apply.

   > Caution:
   >
   > Properties with dynamic and cascading lists of values (LOVs) are not supported.

5. Click **Close** to close the **Configurations - Attribute Exchange** dialog box.

# Modify an attribute map

> Note:
>
> You must be in the owning group of the user who mapped the property or a member of the configuration manager group.

1. Open and check out a Teamcenter dataset.
2. Click the **Teamcenter** tab.
3. In the **Attribute Exchange** button group, click **Configurations**.

   The system displays the **Configurations - Attribute Exchange** dialog box.
4. Select a mapping row to modify.
5. Click **Modify**.

   The system displays the **Create/Modify - Attribute Exchange** dialog box.

   > Note:
   >
   > You must be in the mapping's owning group or in the configuration manager group to modify the mapping.
6. Modify the office document properties, exchange direction, and Teamcenter attributes.

   > Caution:
   >
   > Properties with dynamic and cascading lists of values (LOVs) are not supported.
7. Click **Apply** to modify the attribute exchange mapping and leave the **Create/Modify - Attribute Exchange** dialog box open.

   The mapping is modified in the **Configurations - Attribute Exchange** dialog box.
8. Click **OK** to modify the attribute exchange mapping and close the **Create/Modify - Attribute Exchange** dialog box.

   The mapping is modified in the **Configurations - Attribute Exchange** dialog box.

# Set the default locale and the manager group

1. Open and check out a Teamcenter dataset.

2. Click the **Teamcenter** tab.

3. In the **Attribute Exchange** button group, click **Configurations**.

   The system displays the **Configurations - Attribute Exchange** dialog box.

   > Note:
   >
   > Optionally, you can open the **Configurations - Attribute Exchange** dialog box from the **Attribute Exchange** button group by clicking the list button next to **Configurations** and clicking **All Configurations**.

4. In the **General Settings** pane, click the **Locale** box and select a language that is the Teamcenter default locale.

   > Note:
   >
   > This locale is only for attribute exchange. It is not the Office Client session culture. You have to be in the manager group to modify this locale.

5. In the **General Settings** pane, click the **Change** button to the right of **Configuration Manager Group**

   The system displays the **Groups - Attribute Exchange** dialog box.

   > Note:
   >
   > You must be in the configuration manager group to modify the configuration manager group.

6. Select a manager group for the attribute exchange.

   The configuration manager group is the administrator group for the mapping and the attribute exchange. This manager group defaults to the original document author's Teamcenter group. If the document is created under item revision definition configuration (IRDC) control, this is what the IRDC template is set to.

7. Type text in the **Search** box to filter the list.

8. Click **Reload** to display a complete list of Teamcenter groups.

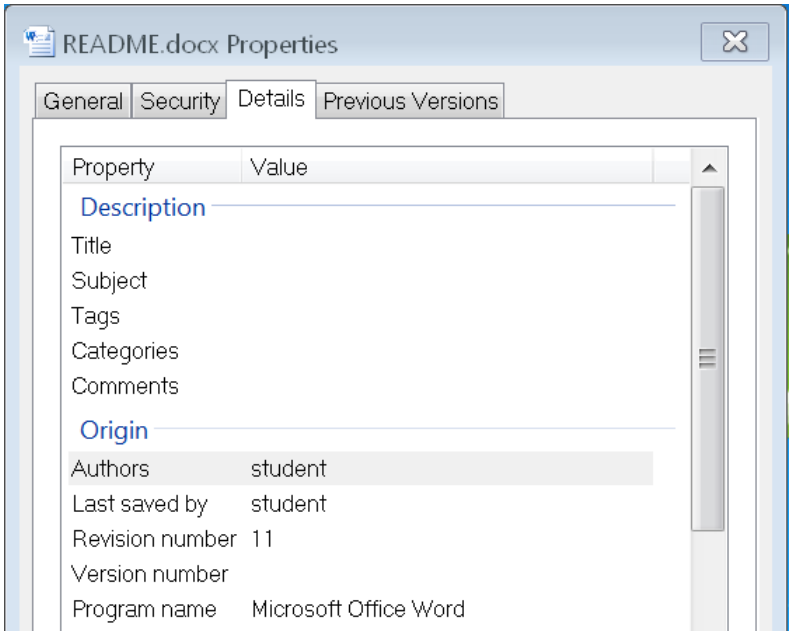
9. Click **OK** to set the configuration manager group and close the **Groups - Attribute Exchange** dialog box.

## View mapped properties in Office files

1. On the Office ribbon, choose **File→Info**.

2. In the right pane, select **Properties→Advanced Properties** to open the **Properties** dialog box.

> Tip:
>
> You can also open the **Properties** dialog box by selecting **Document Properties→Advanced Properties** in the Word Document Panel.




3. Click the tabs in the dialog box to view the mapped properties by category.

## Add a custom property in Word

1. In a Word document that you want to synchronize with Teamcenter, choose **File→Info**.

2. In the right pane, select **Properties→Advanced Properties** to open the **Properties** dialog box.

> Tip:
>
> You can also open the **Properties** dialog box by selecting **Document Properties→Advanced Properties** in the Word Document Panel.

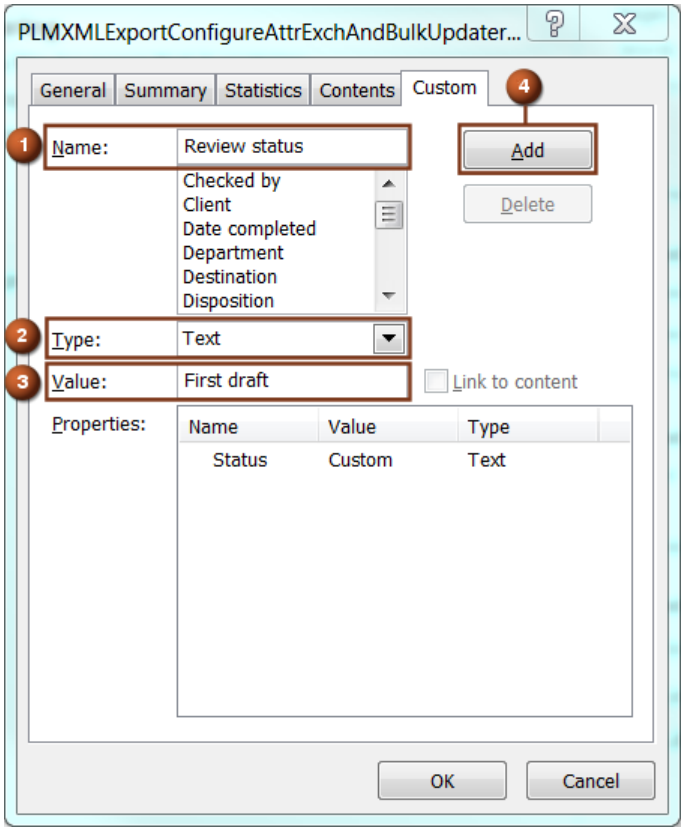
3. Click the **Custom** tab, and then do the following:

   a. Enter the property name in the **Name** box.

   b. Select the property type from the **Type** list.

   c. Enter the initial property value in the **Value** box.

d. Click **Add**.




The new property appears in the **Properties** table in the dialog box.

4. Do one of the following:

   - To add another custom property, repeat this procedure.
   - To commit your additions and close the **Properties** dialog box, click **OK**.

> Note:
>
> When using a custom attribute of the **Date** type, you cannot enter a date prior to Jan 1, 1900 as Teamcenter initializes the start date as Jan 02, 1900.

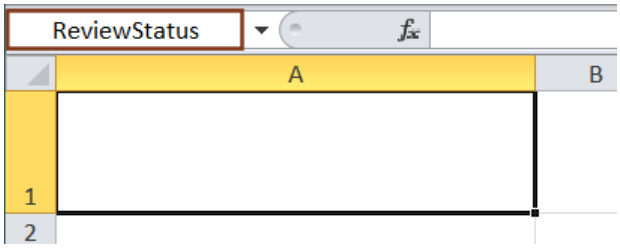
## Add a custom property in Excel

1. In an Excel spreadsheet that you want to synchronize with Teamcenter, rename the cell that you want to contain the property value.

   a. Right-click the cell (for example, **A1**) and choose **Define Name** to open the **New Name** dialog box.

b. Enter the new name in the **Name** box, and then click **OK**.

The new name appears to the left of the **fx** field.




2. Choose **File→Info**.

3. In the right pane, select **Properties→Advanced Properties** to open the **Properties** dialog box.

> Tip:
>
> You can also open the **Properties** dialog box by selecting **Document Properties→Advanced Properties** in the Word Document Panel.

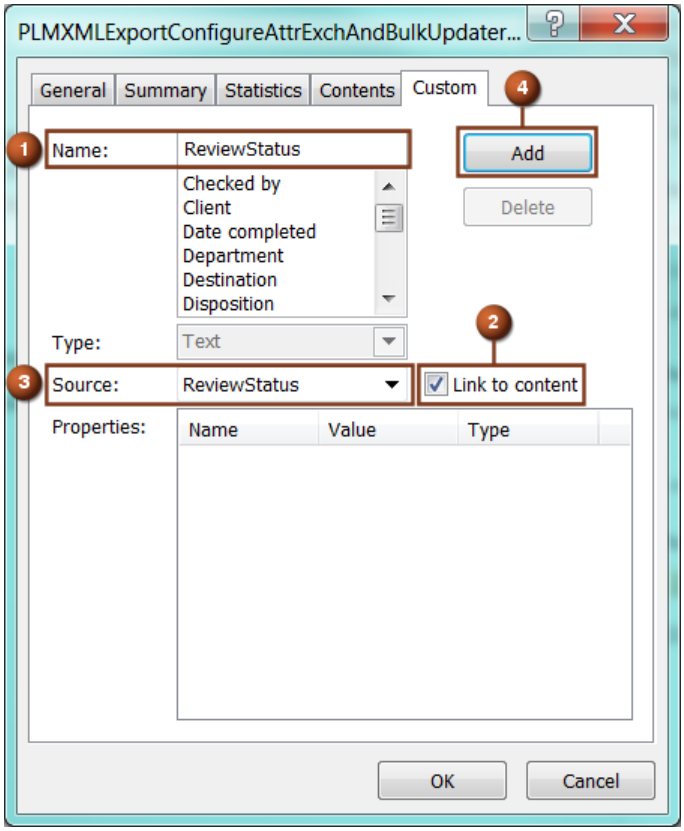
4. Click the **Custom** tab, and then do the following:

a. Enter the property name in the **Name** box.

b. Check the **Link to content** box to display the **Source** list in place of the **Value** box.

c. Select the renamed cell in the **Source** list.

d. Click **Add**.




The new property appears in the **Properties** table in the dialog box.

5. Do one of the following:

   - To add another custom property, repeat this procedure.
   - To commit your additions and close the **Properties** dialog box, click **OK**.

Note:

When using a custom attribute of the **Date** type, you cannot enter a date prior to Jan 1, 1900 as Teamcenter initializes the start date as Jan 02, 1900.

Also, if you enter a date in the months of January or February 1900 in Excel, the value is not displayed in Teamcenter correctly. This is because Microsoft Excel treats the year 1900 as leap year; therefore, the date conversion in Microsoft Excel is off by one day for the months of January and February of the year 1900.

## Edit document property values

1. Open and check out a Teamcenter dataset.
2. Click the **Teamcenter** tab.

3. In the **Attribute Exchange** button group, click **Edit Values**.

   The system displays the **Edit Values - Attribute Exchange** dialog box.

   > Note:
   >
   > Only the mappings for **File to Teamcenter** and **Two Way** allow edits. The mappings for **Teamcenter to File** are not displayed.

4. Select a row in the table to edit.

5. Enter a value in the **Value** box.

   > Tip:
   >
   > You can click **Refresh** to return the original value or click **Reset to Defaults** to return the default value.

6. Click **Apply** to apply the value updates and leave the **Edit Values - Attribute Exchange** dialog box open.

7. Click **OK** to apply the value updates and close the **Edit Values - Attribute Exchange** dialog box.

## Reload Teamcenter values to Office

1. Open and check out a Teamcenter dataset.

2. Click the **Teamcenter** tab.

3. In the **Attribute Exchange** button group, click **Reload**.

   The system displays the **Confirm Reload** dialog box. Any locally modified Office properties are lost with the updated values from Teamcenter.

4. Click **OK** to continue with the reload.

## Delete an attribute map

> Note:
>
> You must be in the owning group of the user who mapped the property or a member of the configuration manager group.

1. Open and check out a Teamcenter dataset.

2. Click the **Teamcenter** tab.

3. In the **Attribute Exchange** button group, click **Configurations**.

   The system displays the **Configurations - Attribute Exchange** dialog box.

4. Select a mapping row to delete.

5. Click **Delete**.

   The system displays the **Delete - Warning** dialog box.

   > Note:
   >
   > You must be in the mapping's owning group or in the configuration manager group to delete the mapping.

6. Click **OK**.

   The selected mapping is removed from the list and deleted when the file is saved back to Teamcenter.

# Converting legacy PropSync datasets for attribute exchange

## Mapping attributes for PropSync datasets

When PropSync forms are attached to legacy datasets, that data is incompatible with Client for Office. For such datasets, attribute mapping conversion involves the following procedures.

> Note:
>
> This only works to convert to Microsoft based attribute exchange, not to logical objects.

1. Set up the **fnd0InstanceAttrExMappings** property for export.

   The **fnd0InstanceAttrExMappings** property contains the attribute mapping data for Teamcenter datasets. You use PLM XML/TC XML Export Import Administration to set up the property for export.

   In this procedure, you first create a *property set* that relates the **fnd0InstanceAttrExMappings** property to the **Dataset** primary object type. Next, you associate the new property set with the **unconfiguredDataFileExport** *transfer mode*.

2. Export the **fnd0InstanceAttrExMappings** data to PLM XML.

Using the **unconfiguredDataFileExport** transfer mode, you export the attribute mapping data to a PLM XML output file. Later in the conversion process, you use this data to replace the legacy data that you export with the **attribute_export** utility.

3. Build an XML file as input to the **attribute_export** utility.

   In the XML input file, you specify property names and values as search criteria for the datasets to be updated. You also specify a temporary value for the **fnd0InstanceAttrExMappings** property. When you run the **attribute_export** utility, you enter this file name in the **-inputfile** argument.

4. Export the legacy data with the **attribute_export** utility.

   The **attribute_export** utility queries for the datasets that meet the criteria you specify in the XML input file. For each found dataset, the utility inserts the temporary value in the **fnd0InstanceAttrExMappings** property.

   The utility outputs the data to a TC XML file. In this file, you replace each temporary value with the corresponding data from the PLM XML output file. Then, you save the changes and import the replacement data to update the datasets.

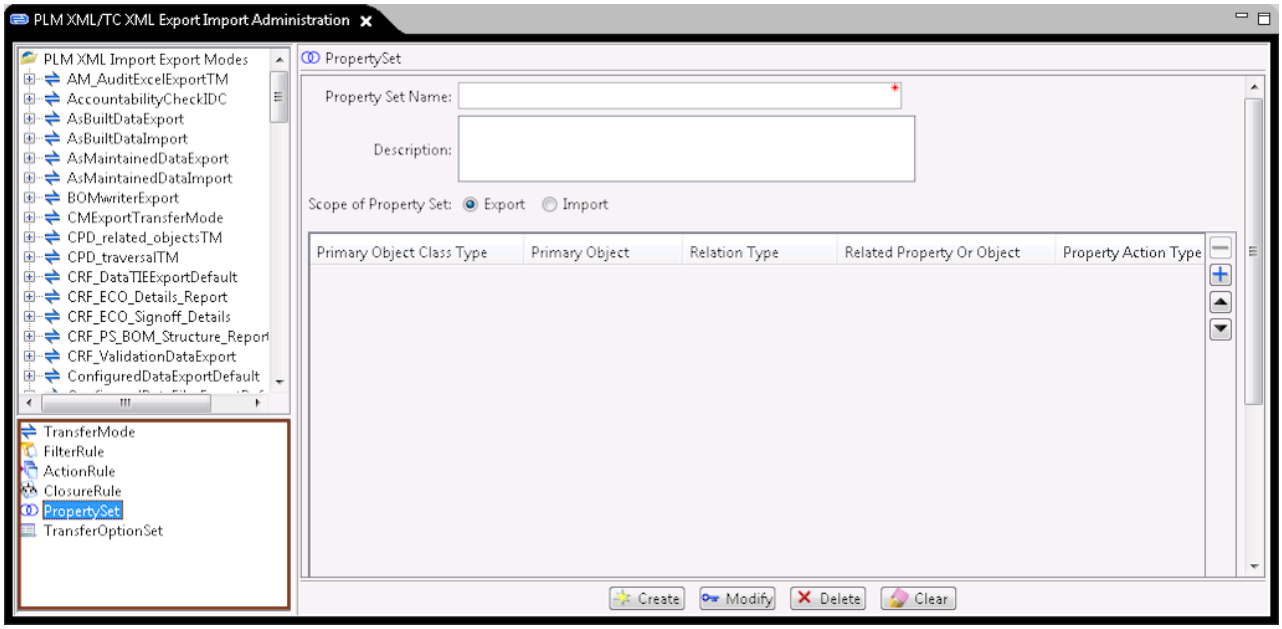
5. Import the replacement data to the legacy datasets.

   You run the **tcxml_import** utility to update the dataset properties with the replacement data from the TC XML output file. For a fast bulk update, you use the **-bulk_load** and **-bypassSiteCheck** arguments.

   > Note:
   >
   > The **-bulk_load** argument requires a Site Consolidation license key value in the **SITCONS_AUTH_KEY** environment variable. The license key is available on the Support Center:
   > **https://support.sw.siemens.com**

## Set up the fnd0InstanceAttrExMappings property for export

1. In PLM XML/TC XML Export Import Administration, select **PropertySet** in the **TransferMode** tree.




2. In the **TransferMode** pane, enter the object name in the **Property Set Name** box.

   You can also enter additional information in the **Description** box.

3. Beside **Scope of Property Set**, select **Export**.

4. Add an ordered clause that specifies how to traverse the data structure.

   a. Click the **Add clause** button + to the right of the clause table.

   b. Under **Primary Object Class Type**, click the empty cell and select **CLASS**.

   c. Under **Primary Object**, double-click the empty cell and type **Dataset**.

   d. Under **Relation Type**, click the empty cell and select **ATTRIBUTE**.

   e. Under **Related Property Or Object**, double-click the empty cell and type **fnd0InstanceAttrExMappings**.

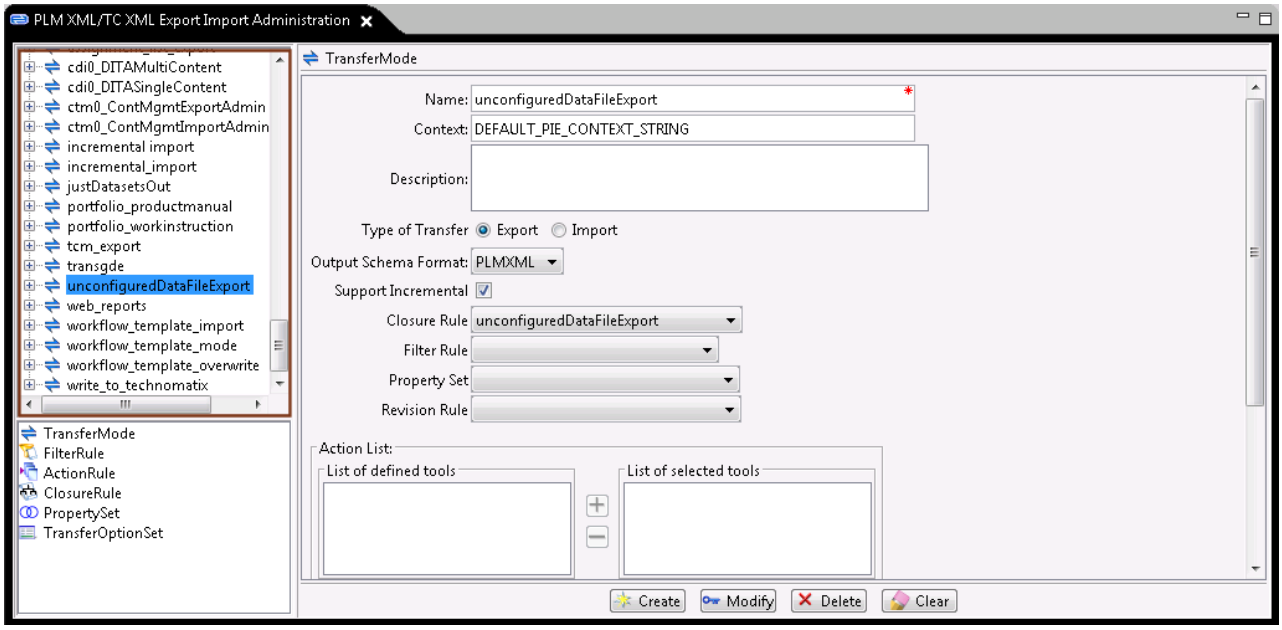
   f. Under **Property Action Type**, click the empty cell and select **DO**.

| Primary Object Class Type | Primary Object | Relation Type | Related Property Or Object | Property Action Type |
|---|---|---|---|---|
| CLASS | Dataset | ATTRIBUTE | fnd0InstanceAttrExMappings | DO |

5. At the bottom of the **TransferMode** pane, click **Create**.

The new property set is added under **PropertySet** in the bottom left pane.

6. In the object tree, select the **unconfiguredDataFileExport** transfer mode.




7. In the **TransferMode** pane, select the new property set in the **Property Set** list, and then click **Modify**.

The property set is added to the **unconfiguredDataFileExport** transfer mode in the object tree.

## Export the fnd0InstanceAttrExMappings data to PLM XML

1. Select the DMTemplate item revision or its dataset.

2. Choose **Tools→Export→To PLMXML** to open the **PLMXML Export** dialog box.

3. In the **Export Directory** box, enter the full path of the directory where you want to store the XML output file.

4. In the **Export Filename** box, enter the name of the XML output file.

5. In the **Transfer Mode** list, select **unconfiguredDataFileExport**.

6. Click **OK** to initiate the export.

7. When the export is complete, open the output file and verify that it contains the **fnd0InstanceAttrExMappings** property data.

The data in an **Item** entry is a string from the string array in the **fnd0InstanceAttrExMappings** property.

```
<DataSet id="id13" name="DMTemplateForSampleSDDWordXDatasetAttrExch"
accessRefs="#id3" releaseStatusRefs="#id18" version="3" memberRefs="#id17"
type="MSWordX"> <Description>Sample SDD</Description>
<ApplicationRef version="gmdFGCjDAABaaA" application="Teamcenter"
label="gmdFGCjDAABaaA"></ApplicationRef>
<UserData id="id15"> <UserValue value=""
title="fnd0InstanceAttrExMappings">
<UserList id="id3" type="list">
<Item value="&lt;?xml version=&quot;1.0&quot;encoding=&quot;utf-8&quot;
?&gt;&lt;AttributeExchangeConfiguration xmlns:xsi=&quot;http://www.w3.org/
2001/
XMLSchema-instance&quot; xmlns:xsd=&quot;http://www.w3.org/2001/
XMLSchema&quot;version=&quot;1&quot;&gt;&lt;AdminGroup&gt;dba&lt;/
AdminGroup&gt;&lt;
Locale&gt;en_US&lt;/Locale&gt;&lt;MappingList&gt;&lt;Mapping source=&quot;
instance&quot;&gt;&lt;Direction&gt;TeamcenterToFile&lt;/Direction&gt;&lt;
OfficeProperty name=&quot;Author&quot; format=&quot;Text&quot;
propertySet=&quot;Builtin&quot;&gt;&lt;LinkToContent&gt;&lt;Linked&gt;false&
lt;
/Linked&gt;&lt;/LinkToContent&gt;&lt;/
OfficeProperty&gt;&lt;TeamcenterMapping;
bOType=&quot;MSWordX&quot;&gt;&lt;MappingPath&gt;MappingObject&lt;/
MappingPath&gt;
&lt;MappingObject bOType=&quot;DocumentRevision&quot; referencedName=&quot;
item_revision&quot; displayReferencedName=&quot;item_revision&quot;&gt;&lt;
MappingPath&gt;Leaf&lt;/MappingPath&gt;&lt;TcAttribute
name=&quot;DocumentSubject&quot; displayName=&quot;Document Subject&quot; /
&gt;&lt;
/MappingObject&gt;&lt;/
TeamcenterMapping&gt;&lt;Permissions&gt;&lt;OwningGroup&gt;
dba&lt;/OwningGroup&gt;&lt;/Permissions&gt;&lt;/Ma"></Item>
<Item value="pping&gt;&lt;Mappingsource=&quot;instance&quot;&gt;&lt;
Direction&gt;TeamcenterToFile&lt;/Direction&gt;&lt;OfficeProperty
name=&quot;Title&quot;
format=&quot;Text&quot;
propertySet=&quot;Builtin&quot;&gt;&lt;LinkToContent&gt;&lt;
Linked&gt;false&lt;/Linked&gt;&lt;/LinkToContent&gt;&lt;/
OfficeProperty&gt;&lt;
TeamcenterMappingbOType=&quot;MSWordX&quot;&gt;&lt;MappingPath&gt;MappingObj
ect&lt;
/MappingPath&gt;&lt;MappingObject bOType=&quot;DocumentRevision&quot;
referencedName=&quot;item_revision&quot;displayReferencedName=&quot;
item_revision&quot; &gt;&lt;MappingPath&gt;Leaf&lt; /MappingPath&gt;&lt;
TcAttribute name=&quot;DocumentTitle&quot; displayName=&quot;Document
Title&quot;
/&gt;&lt;/MappingObject&gt;&lt;/TeamcenterMapping&gt;&lt;Permissions&gt;&lt;
```

```
OwningGroup&gt;dba&lt;/OwningGroup&gt;&lt;/Permissions&gt;&lt;/
Mapping&gt;&lt;
/MappingList&gt;&lt;/AttributeExchangeConfiguration&gt;"></Item>
</UserList></UserValue></UserData></DataSet>
```

## Build an XML file as input to the attribute_export utility

In a new XML file, enter the following code:

```
<?xml version="1.0" encoding="utf-8"?>
<BulkUpdate>
    <UpdateSets>
        <UpdateSet>
            <type name="Dataset"/>
            <where>
                <cond_prop attrName="object_type" attrValue="MSWordX"/>
                <cond_prop attrName="fnd0InstanceAttrExMappings"
attrValue=""/>
                <cond_prop attrName="last_mod_date" attrValue="18-
Jul-2024 14:56"
                    cond_operator="GE"/>
            </where>
            <update>
                <update_prop attrName="fnd0InstanceAttrExMappings"
                    attrValue="temp_val"/>
            </update>
        </UpdateSet>
    </UpdateSets>
</BulkUpdate>
```

Replace *temp_val* with a temporary value. The **attribute_export** utility inserts this value in the **fnd0InstanceAttrExMappings** property for each exported dataset.

## Export the legacy data with the attribute_export utility

1. To run the **attribute_export** utility, enter the following at the command prompt:

   **attribute_export -u=***userid* **-p=***password* **-g=***group* **-inputfile=***xml-file-name* **-outdir=***path*

| | |
|---|---|
| *userid* | Your Teamcenter user identifier. |
| *password* | Your Teamcenter password. |
| *group* | The Teamcenter user group to which your user identifier is assigned. You must be a member of a group that has **dba** privileges. |

- For the **-inputfile** argument, replace *xml-file-name* with the name of the XML file that you built as input to the **attribute_export** utility.

- For the **-outdir** argument, replace *path* with the full path of the output file directory.

The utility generates a TC XML output file, which contains data for each exported dataset. This data includes the temporary value of the **fnd0InstanceAttrExMappings** property. For example:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<TCXML xmlns="http://www.tcxml.org/Schemas/TCXMLSchema" format="low_level">
       <Dataset elemId="id1" island_id="1" puid="ADYNcOCF6hSNuC"
         lsd="2014-07-18T22:01:56Z" last_mod_date="2014-07-18T22:01:56Z"
fnd0InstanceAttrExMappings="temp_val"/>
       <Dataset elemId="id2" island_id="2" puid="AHSNcOCF6hSNuC"
         lsd="2014-07-18T22:01:56Z" last_mod_date="2014-07-18T22:01:56Z"
fnd0InstanceAttrExMappings="temp_val"/>
       <Dataset elemId="id3" island_id="3" puid="AHXNcOCF6hSNuC"
         lsd="2014-07-18T22:01:56Z" last_mod_date="2014-07-18T22:01:56Z"
fnd0InstanceAttrExMappings="temp_val"/>
...
```

2. For each **fnd0InstanceAttrExMappings** property in the TC XML output file, replace the temporary value with the corresponding data from the PLM XML file.

   In the PLM XML file, the replacement data is the **Item** value string that corresponds to the dataset. For example:

```
<Item value="&lt;?xml version=&quot;1.0&quot;encoding=&quot;
utf-8&quot;?&gt;&lt;
AttributeExchangeConfiguration xmlns:xsi=&quot;http://www.w3.org/
2001/
XMLSchema-instance&quot; xmlns:xsd=&quot;http://www.w3.org/2001/
XMLSchema&quot;
version=&quot;1&quot;&gt;&lt;AdminGroup&gt;dba&lt;/
AdminGroup&gt;&lt;Locale&gt;
en_US&lt;/Locale&gt;&lt;MappingList&gt;&lt;Mapping source=&quot;
instance&quot;&gt;&lt;Direction&gt;TeamcenterToFile&lt;/
Direction&gt;&lt;
OfficeProperty name=&quot;Author&quot; format=&quot;Text&quot;
propertySet=&quot;Builtin&quot;&gt;&lt;LinkToContent&gt;&lt;Linked&gt
;false&lt;
/Linked&gt;&lt;/LinkToContent&gt;&lt;/
OfficeProperty&gt;&lt;TeamcenterMapping;
bOType=&quot;MSWordX&quot;&gt;&lt;MappingPath&gt;
MappingObject&lt;/MappingPath&gt;
&lt;MappingObject bOType=&quot;DocumentRevision&quot;
referencedName=&quot;
item_revision&quot;
displayReferencedName=&quot;item_revision&quot;&gt;&lt;
MappingPath&gt;Leaf&lt;/MappingPath&gt;&lt;TcAttribute
```

```
name=&quot;DocumentSubject&quot;
displayName=&quot;Document Subject&quot; /&gt;&lt;
/MappingObject&gt;&lt;/TeamcenterMapping&gt;&lt;Permissions&gt;&lt;
OwningGroup&gt; dba&lt;/OwningGroup&gt;&lt;/Permissions&gt;&lt;
;/Mapping&gt;&lt;/MappingList&gt;&lt;
/AttributeExchangeConfiguration&gt;"></Item>
```

3. Save the changes.

## Import the replacement data to the legacy datasets

> Note:
>
> In the **tcxml_import** utility, the **-bulk_load** argument requires a Site Consolidation license key value in the **SITCONS_AUTH_KEY** environment variable. The license key is available on the Support Center:
> **https://support.sw.siemens.com**

1. To run the **tcxml_import** utility, enter the following at the command prompt:

   **tcxml_import -u=***userid* **-p=***password* **-g=***group* **-bulk_load -file=***xml-file-name* **-bypassSiteCheck**

| | |
|---|---|
| *userid* | Your Teamcenter user identifier. |
| *password* | Your Teamcenter password. |
| *group* | The Teamcenter user group to which your user identifier is assigned. You must be a member of a group that has **dba** privileges. |

   - The **-bulk_load** argument performs a fast bulk update of legacy data from a TC XML file.
   - For the **-file** argument, replace *xml-file-name* with the name of the TC XML file that contains the input data.
   - The **-bypassSiteCheck** argument bypasses the check that prevents the utility from updating objects at the same site and the check that prevents the update of replica objects.

   All other **tcxml_import** arguments are ignored.

2. Verify the data in Teamcenter or in Client for Office.
   - In Teamcenter, you can select a dataset and view its properties in the **Summary** view.

- In Client for Office, you can open a dataset and view mapped properties in the **Properties** dialog box.

# 17. Synchronizing attributes from Teamcenter to Adobe PDF files by using logical objects

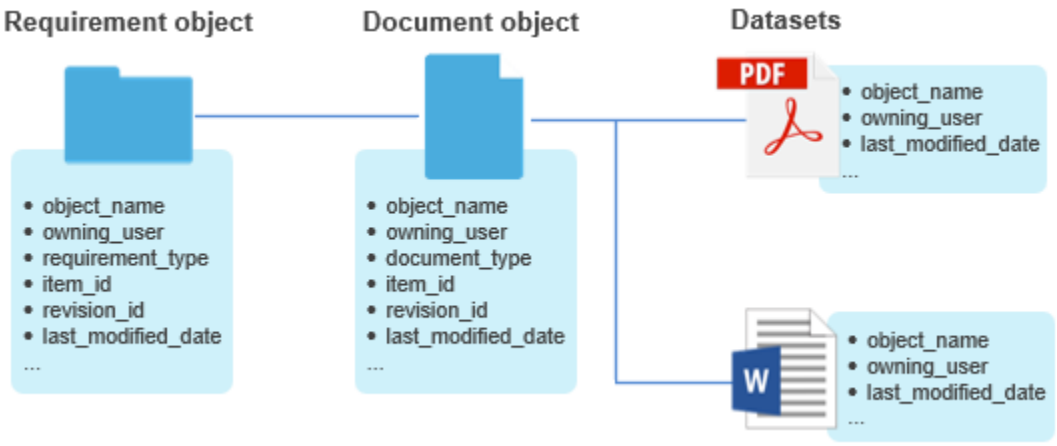
## About logical objects

A *logical object* is a container that consolidates attributes (properties) from different related business objects. For example, a document (item) and its related datasets are different objects in Teamcenter. The attributes of the document can be its ID, name, or description. The different attributes of a dataset can be author, created date, or reviewed by.

The following example shows how you can synchronize attributes, consolidated by the logical object, from Teamcenter to a Microsoft Word document and to a PDF file.

**Example**

Consider that there is a requirement object in Teamcenter which has a reference relation to a document object. The document object has two attached files (datasets), one Word and one PDF.

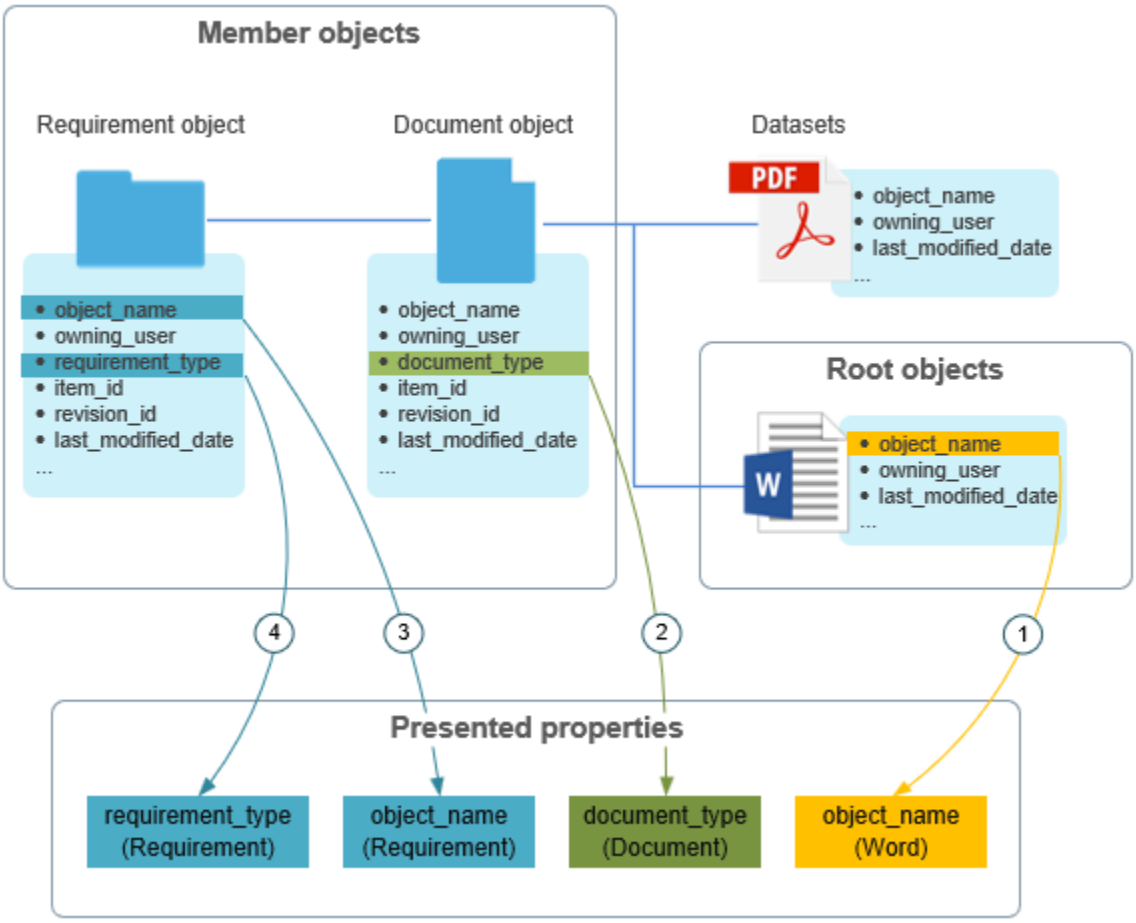



You want to consolidate four attributes from these objects:

- **object_name** and **requirement_type** from the requirement object.
- **document_type** from the document object.
- **object_name** from the Word or PDF file attached to the document.

For consolidating these attributes, you define a logical object. A logical object is composed of *root object*, *member objects*, and *presented properties*. You begin consolidating the properties from the

dataset. This makes the dataset the root object. You then traverse backward tracing its relation to the document object and then to the requirement object. This makes the document and requirement objects the member objects. While traversing from the dataset to the document object and then to the requirement object, you collect the required four attributes in the logical object. These four attributes are the presented properties.




After consolidating the attributes in a logical object, you use the logical object to synchronize attributes between Teamcenter and Microsoft Office files and PDF files. When you synchronize attributes between Teamcenter and Microsoft Office files, you cannot synchronize attributes from Teamcenter to Office files and from Office files to Teamcenter at the same time.

> Note:
>
> Even if you used attribute-level security while implementing Teamcenter, the properties that are used for attribute exchange are still synchronized in the document. The attribute exchange overrides the attribute-level security.

## Task flow to synchronize attributes by using logical objects

1. **Define a logical object**.

2. **Define when to add the Teamcenter attributes in template files**.

3. To synchronize the attributes between Teamcenter and Microsoft Office files, **specify the logos, distribution statements, and workflow signoff tables** to be included in the Office files in the Document Management configuration file.

4. To synchronize the attributes between Teamcenter and Microsoft Office files, define the placement of attributes, logos, distribution statements, and workflow signoff tables in the **Word**, **Excel**, and **PowerPoint** files, if required.

   To synchronize the attributes between Teamcenter and PDF files, define the placement of attributes, logos, distribution statements, and workflow signoff tables in the PDF files, if required.

5. Include the logos, distribution statements, and workflow signoff tables in the synchronized files by setting up a **system stamp configuration**.

6. **Relate the logical object with a file (dataset)**.

When you synchronize attributes between Teamcenter and Microsoft Office files, the Teamcenter attributes along with logos, distribution statements, and workflow signoff tables are included in a file synchronously when a user creates or renders a file. They are included in a file asynchronously through Dispatcher when a user checks in, saves as, or revises a document revision. For more information about workflow signoff table, see *server-install-location\samples\document_management\readme.pdf*. To get the latest information in an Office file, a user can also send the file through an attribute exchange workflow process. For this, you must set up a workflow task for the users by using the **DOCMGT-update-docprop-logicalobject** action handler.

When you synchronize attributes between Teamcenter and PDF files, The Teamcenter attributes along with logos, distribution statements, and workflow signoff tables are included in a PDF file when a user sends the file through an attribute exchange workflow process. For this, you must set up a workflow task for the users by using the **DOCMGTAPP-apply-pdf-control** action handler.

Finally, **you may verify if you have set up the attribute exchange correctly**.

## Define logical objects

You can define logical objects for attribute synchronization. Note that the logical objects do not support the runtime properties for attribute synchronization.

1. In Active Workspace, log on with DBA credentials and select **Default** as the **Workspace**.

2. Click the **LOGICAL OBJECTS** tile on the home page.

3. On the **Logical Object Configuration** page, choose **More Commands** ••• > **New** ✳ > **Define logical object** ⊕.

4. In the **Add** panel:

   a. Enter **Internal Name**, **Name**, and **Description** for the logical object.

   b. In the **Root Object** list, click a root object. For example, click **Dataset**.

   c. Click the **Retrieve Classification Data** checkbox if you want to retrieve the classification data.

   d. In the **Parent Logical Object** list, click a parent logical object. For example, click **Logical Object**.

   e. Click **Add**.

5. In the **Overview** tab, click **Add** ⊕ to add new members to the logical object.

6. In the **Add Member** panel:

   a. Enter **Member ID** and **Display Name**.

   b. Select **Backward** in **Segment 1**.

   c. Select **Business Object**, for example, **DocumentRevision**.

   d. Select **Relation or Reference**, for example, **Tc_Attaches**.

   e. Click **Add Segment** ⊕ in **Segment 2**.

   f. Select **Forward**.

   g. Select **Relation or Reference**, for example, **items_tag**.

   h. Select **Business Object**, for example, **Document**.

   i. Click **Add**.

7. Click **Add** ⊕ again to add another member to the logical object to relate the business object with the dataset by using a different relation.

8. For each member, add member properties that you want to consider for document rendering. To do so:

   a. In the **Presented Properties** section, click **Add** ⊕.

   b. In the **Add** panel, click a property of the member in the Root Or Member ID list. For example, **fnd0Root**.

c. In the **Member Properties** list, click the required property, for example, **last_mod_date**.

d. Click **Add**.

You can add as many properties as required for each member.

9. Go to the **Exchange Configuration** tab to optionally set the attribute synchronization direction (**Teamcenter to File** or **File to Teamcenter**). If you do not specify the exchange direction, attributes are exchanged from **Teamcenter to File** by default.

10. In the **Exchange Configuration** tab, click **Refresh** to load the exchange configurations corresponding to the presented properties that you added earlier.

Default direction for attribute exchange is **Teamcenter to File**. If **File to Teamcenter** direction for attribute exchange is required, you must set **Is Writable** property to **TRUE** in the **Overview** tab.

> Note:
>
> Attribute exchange cannot be displayed in Microsoft Office Online Excel files. This is a limitation of Microsoft Office Online.

> Tip:
>
> To delete a logical object, you must first delete all the presented properties available in the **Exchange Configuration** tab.
>
> Refresh the presented properties in the **Exchange Direction** tab after deleting presented properties in the **Overview** tab.

## Define when to add the Teamcenter attributes in template files

1. In the existing BMIDE template project, click **Business Objects→ItemRevision**.

2. Double-click **ItemRevision** and in the **Main→Business Object Constants** tab, locate **Fnd0RelToDatasetForLOAttrExch** and click **Edit**.

3. In the **Business Object Constant** dialog box, enter the relations that must be considered including attributes in datasets attached to a Document Management template revision. You can specify more than one relation by using commas, for example, **TC_Attaches**, **IMAN_specification**.

4. Click **Finish**.

5. In the **Main→Business Object Constants** tab, locate **Fnd0TriggerLOAttrExch** and click **Edit**.

6. In the **Business Object Constant** dialog box, select one of the following in **Value** and click **Finish**:

- **Disabled**

  This value indicates that the Teamcenter attributes will not be added to template files automatically. This is the default value.

- **Configured**

  This value indicates that the Teamcenter attributes will be added to template files automatically. The attributes will also be added to the datasets attached to the Document Management template revision with relations specified in the **Fnd0RelToDatasetForLOAttrExch** business constant.

- **Enabled**

  This value indicates that the Teamcenter attributes will be added to template files and to all the attached datasets automatically.

## Define the placement of attributes in Adobe PDF files

To define the placement of Teamcenter attributes, workflow signoff tables, logos,and distribution statements in Adobe PDF files, you update the:

- **Meta Data Stamp file** for rendered or printed PDFs.
- **PDF command file** for existing PDFs.

After defining the placement information, you attach these files to the system stamp Document Management template revision.

## Set up system stamps

1. In the Business Modeler Integrated Development Environment (BMIDE), click **BMIDE** > **New Model Element**.
2. In the **Add New Model Element** dialog box, select **System Stamp Configuration** in **Wizards** and click **Next**.
3. Enter the following information for the system stamp configuration:

| Field | Action |
|---|---|
| **Business Object** | Select **ItemRevision**. |
| **Condition** | Select the condition under which this stamp is applicable. |

| Field | Action |
| --- | --- |
| **Applies To** | Choose one of the following:<br>• **Print**: To include the system stamps in a printed file.<br>• **Render**: To include the system stamps in a rendered file.<br>• **PDF_Control**: To include the system stamps in an existing PDF file when it is sent to a stamp workflow.<br>• **PrintAndRender**: To include the system stamps in both printed and rendered files. |
| **Include User Name?** | Select this check box to include the print requester's name in the printed files. |
| **Include Date And Time?** | Select this check box to include the print date and time in the printed files. |

4. Click **Next** and enter the following stamp information:

| Field | Action |
| --- | --- |
| **Properties** | Specify the properties that must be included in the printed or rendered files. You can optionally specify a prefix that must precede a property, for example, **Document Name**: <**object_name**>. |
| **User Stamp** | Type the text that you want to appear on the document, such as *Internal Distribution*. |
| **Watermark** | Type the text that you want to appear as a watermark, such as *Confidential*. |
| **MDS Template** | Select the MetaData Stamp template that defines how the stamps must be applied in the printed and rendered files.<br>To apply stamps in existing PDF files, select the PDF control command XML file.<br>To apply stamps during attribute synchronization, select the Document Management template revision for stamps. |

5. Click **Finish**.

The newly created system stamp configuration is listed under **Extensions**>**Document Management** > **System Stamp Configuration**.

## Relate logical objects with datasets

You can relate one or more logical objects to a dataset. To do so:

1. If you are an administrator, in Active Workspace, open and check out the document template revision containing the required dataset.

   > Note:
   >
   > To apply the **Logical Object Type Relation** property to a document revision other than the document template, ensure that the style sheet of that document revision contains the *Fnd0LogicalObjectTypeRel* property added to it. If the property is already not added to the document revision, edit the style sheet for adding the property and then apply it.

2. In the **Files** section, change the view to **Table**.

3. In the **Files** section, select the dataset and double-click in the **Logical Object Type Relation** column of the table.

4. Select the logical object that you created for the document revision in **Logical Object Type Relation**.

5. Click **Edit > Check In**.

> Note:
>
> Modifications to a *.docx*, *.pptx*, or *.xlsx* file's body text (irrespective of the values of the mapped attributes) can get lost if a logical object is attached to it, that is, if the **Logical Object Type Relation** is defined for the MS WordX, MS PowerPointX, or MS ExcelX dataset. This can happen because the logical object attribute exchange is asynchronous by design. If you update the dataset file during this time, both processes (dispatcher and user) update two different versions of the same datasets.
>
> To work around this problem, set the **Create Derived Visualization Data** IRDC to perform an **Optional render after checkin** (so that check-in is not stopped). Before the system sends the MS WordX, MS PowerPointX, or MS ExcelX dataset to the dispatcher for rendering, it triggers the logical object attribute exchange first synchronously.

## Set up a workflow to include stamp information in PDF files

1. Create a workflow process template.

   a. In Workflow Designer, click **File→New Root Template**.

   b. In the **New Root Template** dialog box, enter a name in **New Root Template Name**.

c. Select **Empty Template** in **Based On Root Template** and **Process** in **Template Type**.

d. Click **OK** to create a workflow process template.

2. Add a task to the workflow process template.

a. On the toolbar, click **Edit Mode** and then click **Task**.

b. In the process flow pane, double-click where you want to place the new task.

A new task appears with the default name **New Task #**. In the **Name** box, type a name for the task.

3. Link the task to its predecessor and successor tasks.

a. Click the task node you want to specify as the predecessor task.

> Caution:
>
> Do not click the title bar of the task node as this action drags the task node to a different location.

b. Drag your cursor to the task node you want to specify as the successor task.

A link arrow follows the cursor as you drag. When your cursor moves over a task node, the node is highlighted.

c. Release the mouse button.

A link arrow connects the predecessor and successor nodes.

4. Configure the task.

a. Right-click the task and click **Task Properties**.

b. Click the **Task Handler** pane.

c. In **Task Action**, select **Complete**.

d. In **Action Handler**, select **DOCMGTAPP-apply-pdf-control** to create a workflow task to include stamps in the existing PDF files.

e. Click **Create**.

## DOCMGTAPP-apply-pdf-control

**Description** Applies a system stamp, watermark, logo, distribution statement text, workflow signoff table (if the target object is in a review task), and Teamcenter attributes when the logical object is related to the attached PDF dataset. A target object can be an item, an item revision or its subtype, or the PDF dataset itself.

The system stamp is an imprint comprising data such as a watermark and optional boilerplate text. In Business Modeler IDE, the data model administrator creates a system stamp configuration, associating the configuration with the XML command file that defines the watermark and text. The system stamp configuration also contains *Document Configuration*, *Logo*, and *Distribution Statement* datasets.

For this handler to apply the stamp and watermark, the following conditions are required:

- The PDF dataset must be related to the item revision or its subtype.
- The system stamp configuration must be enabled for the item revision or its subtype. The **Applies To** attribute of the system stamp configuration must be set to **PDF_Control**.
- The **PDF Control** access privilege must be granted.
- Logos and distribution statements must be enabled based on their document configuration setting.

**Syntax** **DOCMGTAPP-apply-pdf-control -user_stamp=***text string*

**Arguments** **-user_stamp**

(Optional) Specifies any string for the text portion of the stamp.

**-ignore_error**

(Optional) Specifies that the current task can be continued on to the next task even if the current task fails. If this optional argument is not specified, the current task might stop on failures (including password protected file or dataset is checked out, or object cannot be saved), and might not continue to the next task.

**Placement** Place on the **Start** action or the **Complete** action.

**Restrictions** None

# 18. Synchronize attributes between Teamcenter and Adobe Creative Cloud applications by using logical objects

Use logical objects to exchange attributes-related information between Teamcenter and Adobe Creative Cloud (CC) applications—Illustrator, InDesign, and Photoshop.

To synchronize the attributes between Teamcenter and Adobe CC applications:

1. **Define Teamcenter logical objects.**

2. Identify the text fields of the Adobe CC document that need to be modified.

3. Initialize the text fields of the Adobe CC document.

   Each text field of the Adobe CC document must be `[Name of the Property]`. The `[Name of the Property]` is the name of the property on the Teamcenter logical object.

   After synchronizing the attributes, whenever you open the Adobe CC document, the text field of Adobe CC document fetches latest information from Teamcenter and displays it in the Adobe CC document.

> Note:
>
> Even if you used attribute-level security while implementing Teamcenter, the properties that are used for attribute exchange are still synchronized in the document. The attribute exchange overrides the attribute-level security.

# 19. Setting up the rendering of documents into different file formats

You can render a document into different file formats. For example, you can render a Word document to a PDF so that the user can send the PDF for review. You can set a naming convention for the rendered file. You can also choose to include stamps such as a watermark, logo, distribution statement, and user name in the rendered file.

You can set up documents to render into different file formats when a user:

- Clicks the **Render Document** option in the rich client or the **Generate PDF** command in Active Workspace.
- Sends a document through a workflow process. For this, you must set up a workflow process task by using the **DOCMGT-render-document-revision** workflow action handler.

To set up document rendering, you must:

- Ensure that you have **installed Dispatcher with RenderMgtTranslator and PreviewService translators**. Also, ensure that you have **installed Lifecycle Visualization Convert and Print**.
- **Specify the source and derived file formats** in item revision definition configuration (IRDC). IRDC manages the definition of a document revision in Teamcenter.

  > Note:
  >
  > IRDC and dispatcher service configuration are not required for generating PDFs in Active Workspace using the **Generate PDF** command.

- **Specify the naming convention** for the rendered documents in IRDC, if required.

  > Note:
  >
  > IRDC and dispatcher service configuration are not required for generating PDFs in Active Workspace using the **Generate PDF** command.

- **Set up system stamps**, if required.
- **Set up the file translation** by creating a dispatcher service configuration.
- **Edit the preferences so that the documents can be rendered automatically.**

## Supported source and derived formats for document rendering

| Source format | Derived format |
|---|---|
| Microsoft Office (Word, Excel, and PowerPoint), PostScript | PDF |
| Microsoft Office (Word, Excel, and PowerPoint) | PostScript |
| Microsoft Office (Word, Excel, and PowerPoint) | TIF |
| DXF, HPGL, Text, PostScript, and PDF | Thumbnail (JPEG) |
| Microsoft Office (Word, Excel, PowerPoint, and Project), PostScript, Encapsulated PostScript (***.eps**), Adobe Photoshop (***.psd**), WordPerfect, Rich Text, Bitmap. GIF, JPEG, TIFF, and multipage TIF | PDF |

## Edit preferences for rendering the documents automatically

For rendering a document that is associated with a Document object automatically after the Document object is released, you must first set the following preferences:

- To render a document that is associated with a Document object to PDF when the existing rendered dataset is not present, set the value of the **Fnd0DM_AllowRenderReleasedItemOrSrcDataset** preference to **true**.
- To render a document that is associated with a Document object and override the existing rendered dataset (PDF), set the value of the **Fnd0DM_AllowRenderAllReleasedItemsOrSrcDatasets** preference to **true**. The source dataset must be related to the rendered PDF Dataset using **TC_Derived** relation.

You can set these preferences in Active Workspace or in the rich client.

# 20. Configuring document markup in Teamcenter

## Setting up applications for document markup

As a part of a review process, document reviewers mark up documents by applying geometric and text markups. *Threaded conversations* and *disposition reports* help users jointly discuss document changes. The markup can be retained for retrospective reviews or audit purposes. It is organized by user, timestamp, and disposition of the markup comments. Dispositions provide a way to track the status of and comment on reviewers' markup comments. For example, a reviewer can add a disposition to a comment with the status *Rejected* and add a disposition comment such as *Out of the scope of this project*.

To set up applications for document markups:

- **Specify the markup information** in an item revision definition configuration (IRDC). IRDC defines how a document revision is handled in Teamcenter.
- **Enable applications for markups** by modifying the associated tool objects. A *tool* refers to a software application such as Microsoft Word or Adobe Acrobat.
- **Specify the applications to be launched** for marking up Microsoft Office documents. For example, you can specify that an Excel file can be opened in Word for marking it up.
- **Turn off Adobe protected mode** for marking up PDFs.

You can also **provide access privileges to users for marking up documents** by using access control lists (ACLs).

## Enable applications for document markups

To enable an application for markups, you modify the associated *tool* object, which represents a software application such as Microsoft Word or Adobe Acrobat. To do so:

1. In BMIDE, select the project in which you are performing the Document Management configurations.
2. Go to **Extensions→Options→Tool** and double-click the required application, for example, **MSWord**.
3. Click the **Tool Markup Info** tab and select the **Markup Capable?** check box.

## Specify Microsoft applications to be launched for markups

You can specify which Microsoft applications can be launched when a Microsoft document is opened for review. For example, you can specify that an Excel document can be opened in Word for marking it up.

To do so, you set the following preferences:

- **Excel_ViewMarkup_Launchable_TcTypes**
- **Outlook_ViewMarkup_Launchable_TcTypes**
- **PowerPoint_ViewMarkup_Launchable_TcTypes**
- **Word_ViewMarkup_Launchable_TcTypes**

## Turn off Adobe protected mode for markups

Acrobat Reader version 11.0.09 or later is set up in protected mode on the Windows platform. Teamcenter administrators must turn this mode off to facilitate PDF view and markup for users. Perform the steps specified at **https://www.adobe.com/devnet-docs/acrobatetk/tools/AppSec/sandboxprotections.html#configuration** if you have installed Teamcenter Acrobat/Reader Plugin.

> Example:
>
> To turn off the Adobe protected mode, in the Adobe Acrobat/Reader DC, click **Edit→Preferences** and click **Security (Enhanced)** category. Clear the **Enable Protected Mode at Startup** checkbox.

## Provide access privileges for marking up documents

You can provide access privileges to users for marking up documents by using the following access control lists (ACLs) in Access Manager:

| ACL | Path | Description |
|---|---|---|
| **Markup Item** | **Has Class(POM_application_object) → Working→Has Class (Item)** | View and markup action privileges for items. |
| **Markup Item Revision** | **Has Class(POM_application_object) → Working→Has Class (ItemRevision)** | View and markup action privileges for item revisions |
| **Markup Dataset** | **Has Class(POM_application_object) → Working→Has Class (Dataset)** | View and markup action privileges for datasets |
| **Private Markup** | **Has Class(POM_application_object) → Working→Has** | Read, write, and delete privileges for datasets having private markups |

| ACL | Path | Description |
|---|---|---|
| | **Attribute(Dataset:markup_acl=PrivateMarkup)** | |
| **General Markup** | **Has Class(POM_application_object) → Working→Has Attribute(Dataset:markup_acl=GeneralMarkup)** | Read, write, and delete privileges for datasets having general markups |
| **Markup Official** | **Has Class(POM_application_object) → Working→Has Attribute(Dataset:markup_official=1)** | Read, write, checkin, checkout, and delete privileges for datasets having official markups |

To let users mark up rendered PDFs, you can do one of the following:

- Grant the **Markup** access privilege for the **World** accessor type for all markup ACLs.
- Edit the Dispatcher client configuration file *dispatcher_root***DispatcherClient\conf\Service.properties** to set **Service.DataSetOwner=CAD**.

# Providing markup privileges to specific groups and users

## About providing markup privileges to specific groups and users

When a user creates a document in Teamcenter, all users from the same group can mark up that document by default. However, users from other groups who do not have markup privileges for that document by default might be required to provide suggestions and comments. In such a case, as an administrator, you can use one of the following approaches to provide the markup privileges to those users.

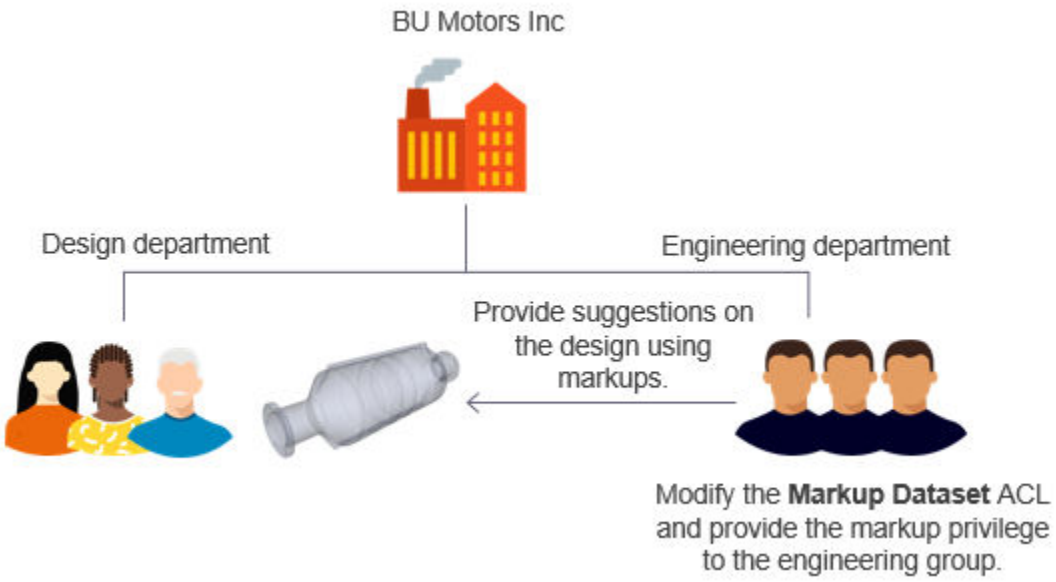
- Provide the markup privilege to all users who are a part of another group by **modifying the Markup Dataset access control list (ACL)**.
- Provide the markup privilege to all users who are a part of a workflow using **Active Workspace** or **rich client**. The users can be from different groups.

Apart from these approaches, a user can directly provide the markup privilege to another user for a document in Rich client.

### Example: providing the markup privilege to all users of another group

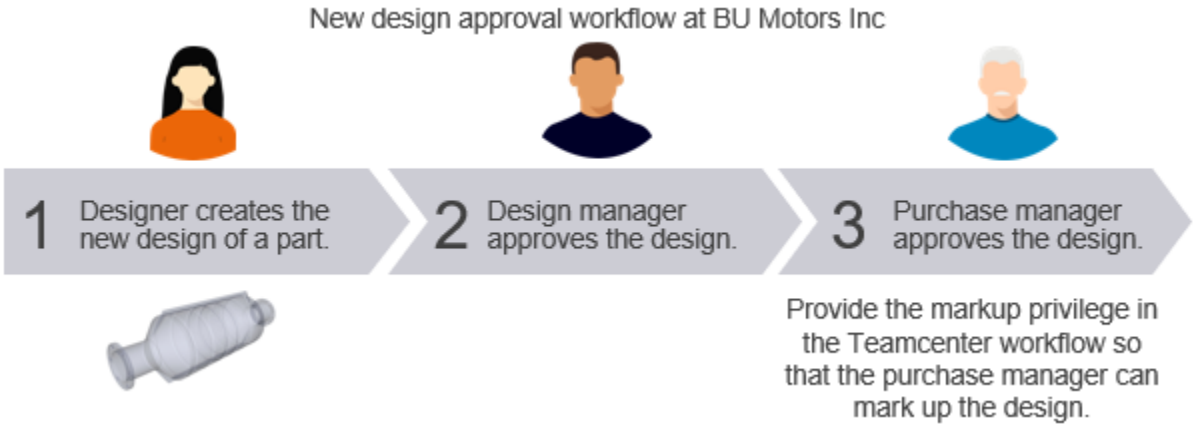
In BU Motors Inc., the designers create the 3D drawings and designs, and then they seek suggestions and comments from the engineering department for better collaboration. Because the engineers are from another group, they cannot provide suggestions and comments on the designs. In such a scenario, as an administrator, you can modify the **Markup Dataset** access control list (ACL), and then you can

provide the markup privilege to the engineering group. This way, all engineers can markup the 3D drawings and designs that the designers create in Teamcenter.



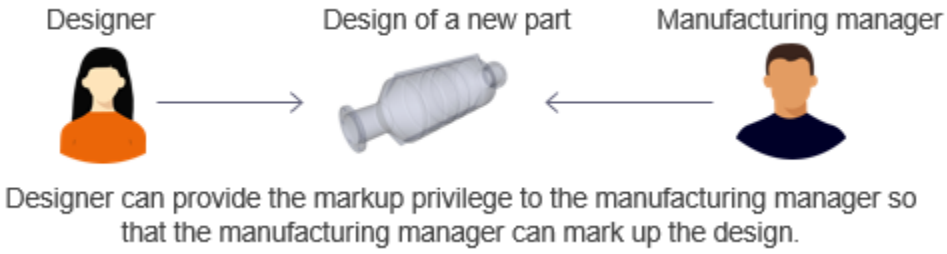

### Example: providing the markup privilege to all users who are a part of a workflow

In BU Motors Inc., the purchasing manager always provides suggestions and also approves the new design of the part. By using this company-specific process, BU Motors Inc. ensures that the raw material required for manufacturing the new part can be procured within a reasonable time frame. Because the purchasing manager is from another group, as an administrator, you can modify the required Teamcenter workflow and provide the markup privilege to the purchasing manager. This way, all users involved in the Teamcenter workflow, including the purchase manager, can markup the documents.



### Example: providing the markup privilege to a specific user for a document

Suppose that a designer in the BU Motors Inc. has designed a new part and wants the manufacturing manager to provide the suggestions on the new design. In such a scenario, the designer can provide the markup privilege to the manufacturing manager for this part design only.



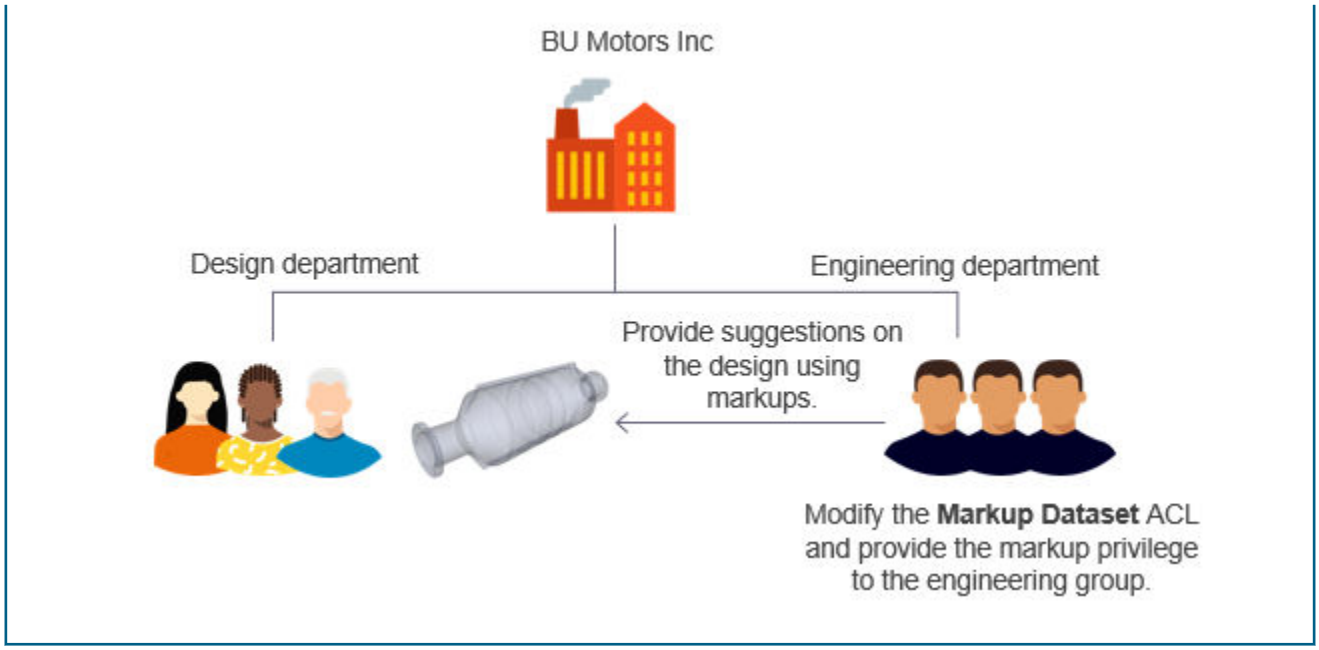

Designer can provide the markup privilege to the manufacturing manager so that the manufacturing manager can mark up the design.

## Provide the markup privilege to all users of another group using Active Workspace

When a user creates a document in Teamcenter, all users from the same group can mark up that document by default. However, users of another group who do not have the markup privilege on that document might be required to provide their suggestions and comments. In such a case, as an administrator, you can provide the markup privilege to the users of another group by modifying the **Markup Dataset** access control list (ACL).

Example:

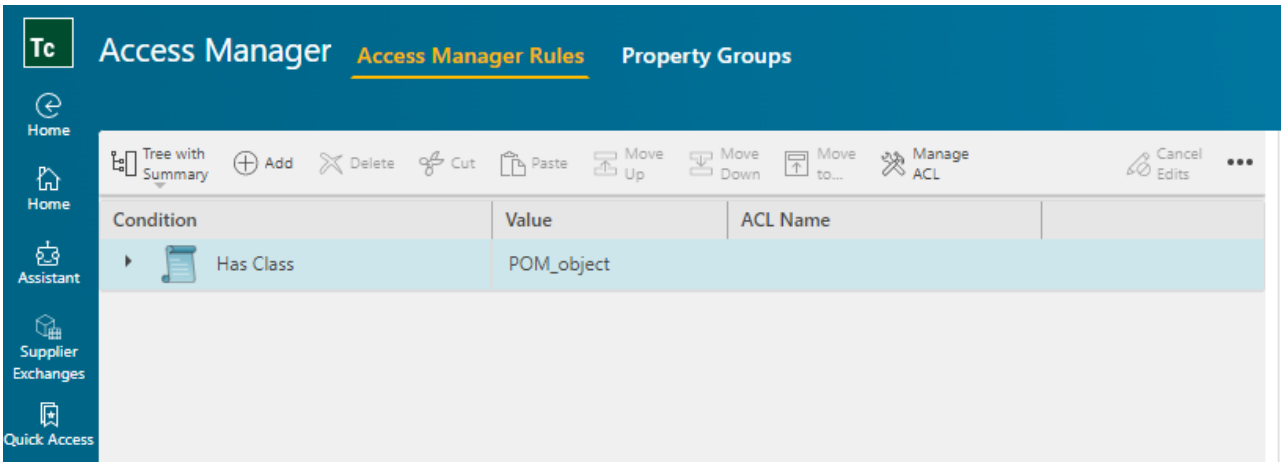
In BU Motors Inc., the designers create the 3D drawings and designs, and then they seek suggestions and comments from the engineering department for better collaboration. Because the engineers are from another group, they cannot provide suggestions and comments on the designs. In such a scenario, as an administrator, you can modify the **Markup Dataset** access control list (ACL), and then you can provide the markup privilege to the engineering group. This way, all engineers can markup the 3D drawings and designs that the designers create in Teamcenter.




## Procedure

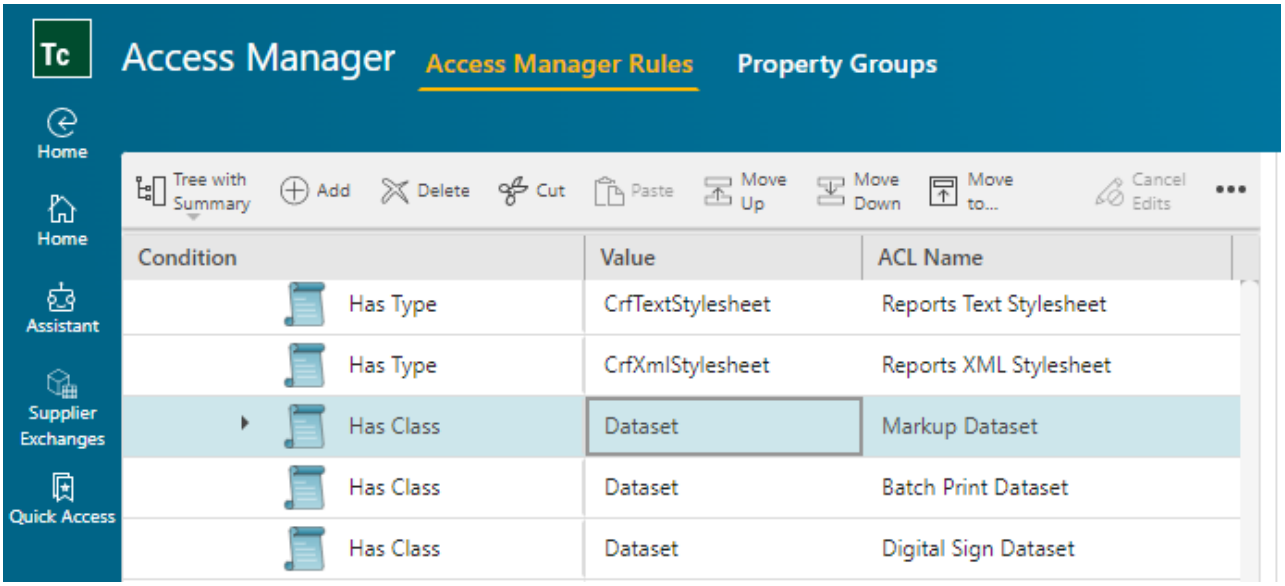
1. As an administrator, log on to Active Workspace and ensure that the **Active Admin** workspace is selected.

2. Click the **ACCESS MANAGER** tile.




3. In the **Access Manager Rules** tab, expand the root-level condition with the name **Has Class** and the value **POM_object**.

4. In the **Has Class** condition, scroll down and expand the **Has Class** condition that has the value of **POM_application_object** and the condition of **Has Class** with the value **Dataset**.

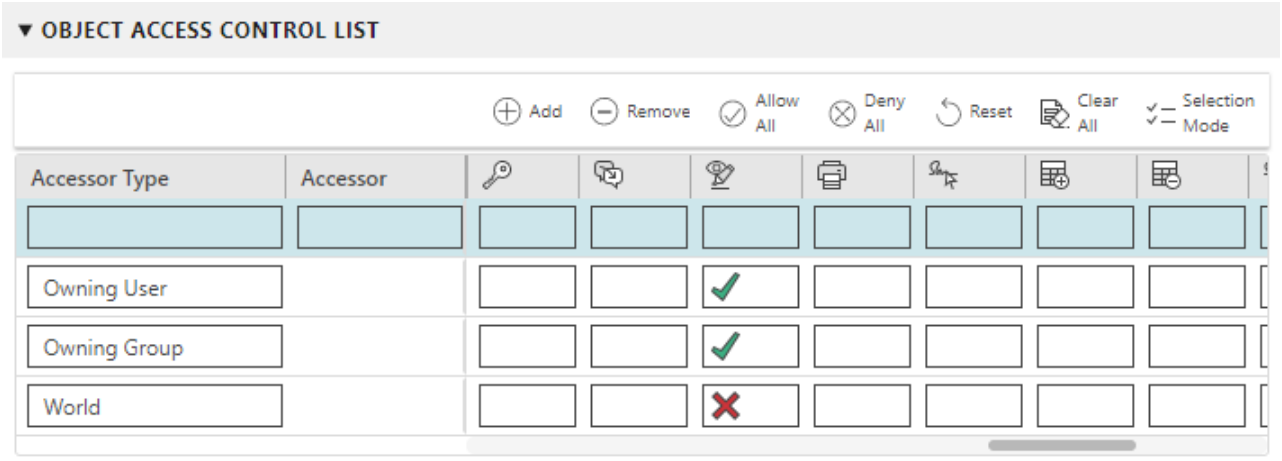



5. Click the **Has Class** condition with value of **Dataset** and the **ACL Name** set as **Markup Dataset**.

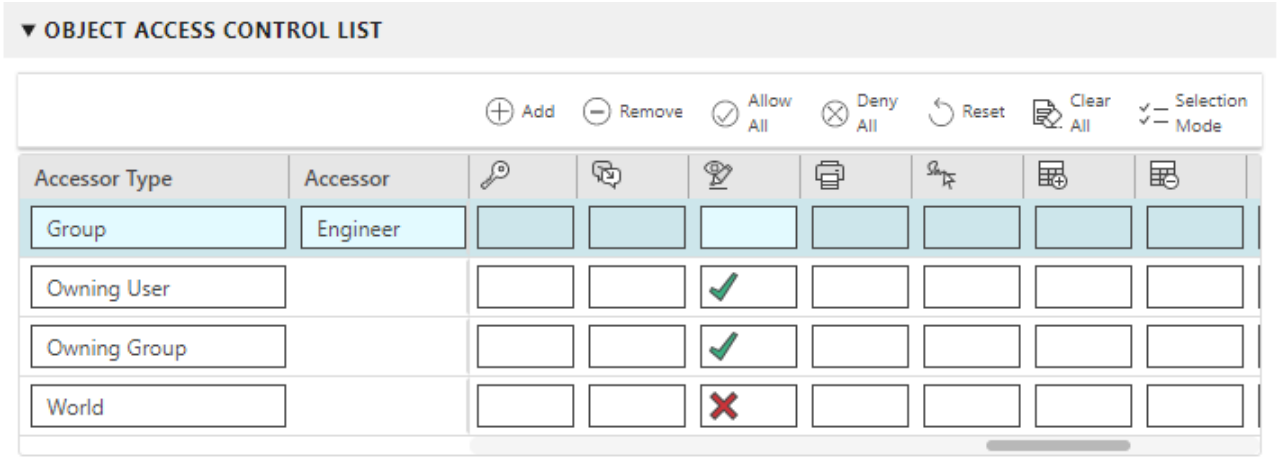
6. In the right pane of **Access Manager**, click **Edit**.

7. In the **OBJECT ACCESS CONTROL LIST** table, click **Add** ⊕.

   A new row is added to the **OBJECT ACCESS CONTROL LIST** table.

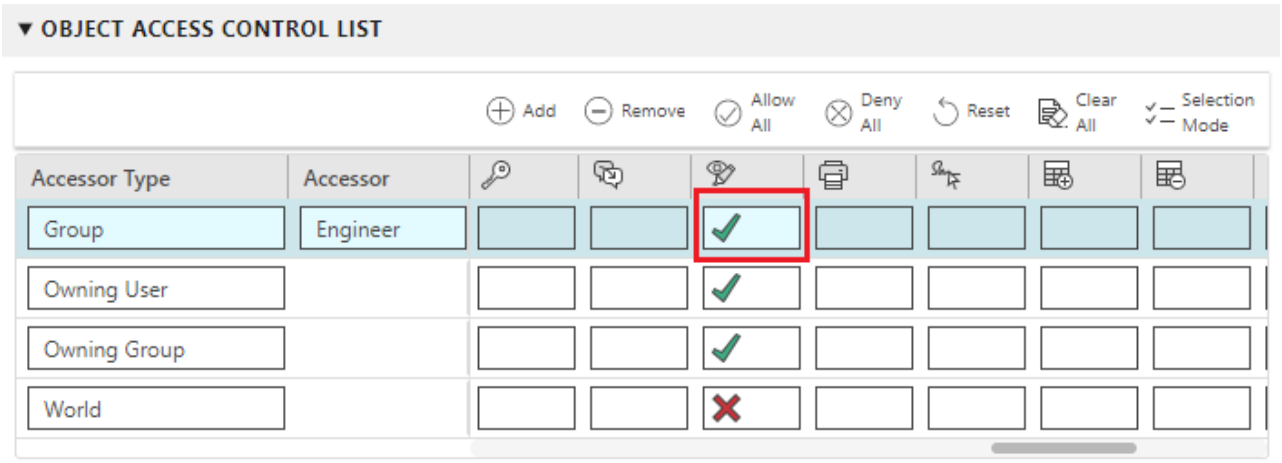



8. In the **Accessor Type** column, click **Group**.




9. In the **Accessor** column, click a group.

   For example, if you want to provide markup privileges to all users in the *Engineering* department, click **Engineer**.




10. In the **Markup** column, double-click and select ✓.

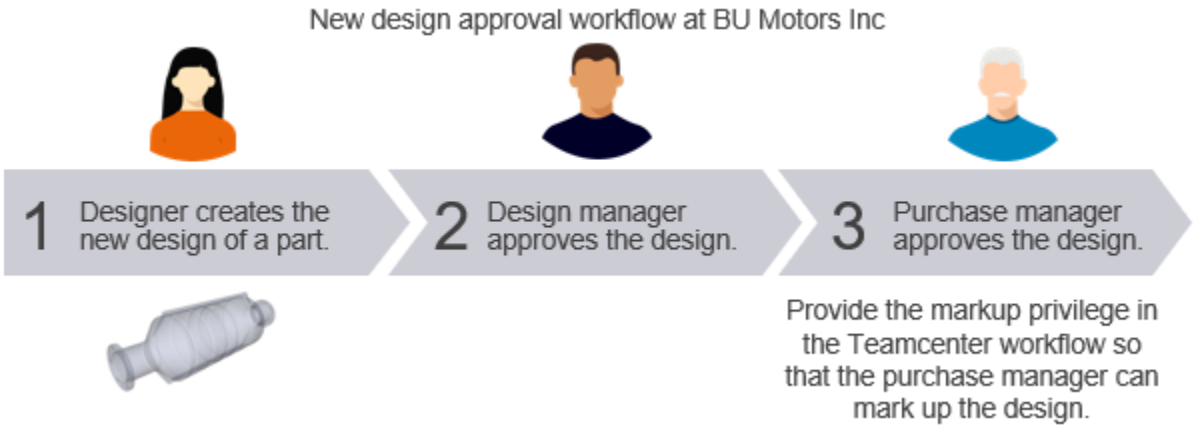
11. Click **Apply**.

## Provide markup privileges to users in a workflow using Active Workspace

Sometimes users from different groups who are working on a Teamcenter workflow are required to provide markups on a document. Because these users are from different groups, they might not have privileges to markup the document. In such a case, as an administrator, you can modify the Teamcenter workflow and provide the markup privileges to the required users.

Example:

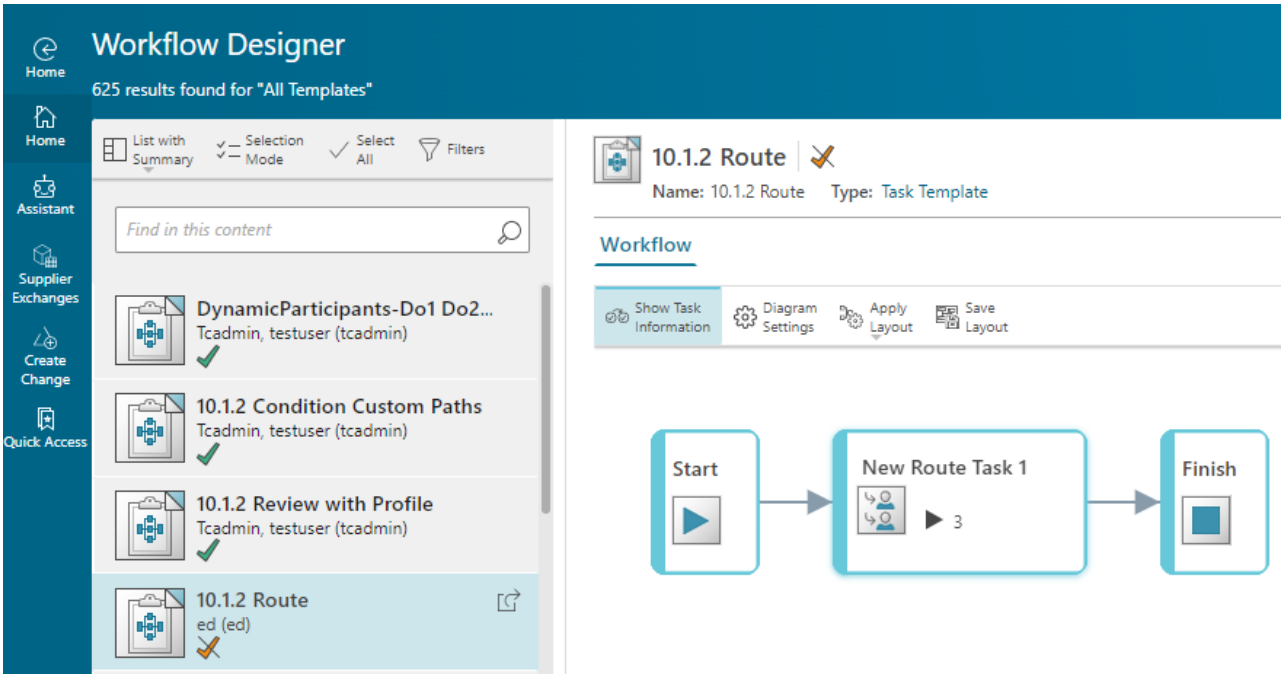
In BU Motors Inc., the purchasing manager always provides suggestions and also approves the new design of the part. By using this company-specific process, BU Motors Inc. ensures that the raw material required for manufacturing the new part can be procured within a reasonable time frame. Because the purchasing manager is from another group, as an administrator, you can modify the required Teamcenter workflow and provide the markup privilege to the purchasing manager. This way, all users involved in the Teamcenter workflow, including the purchase manager, can markup the documents.



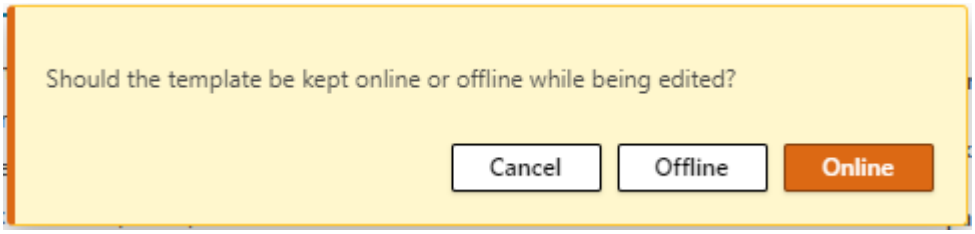

## Procedure

1. As a database administrator, select the **Active Admin** workspace, and then click the **Workflow Designer** tile.

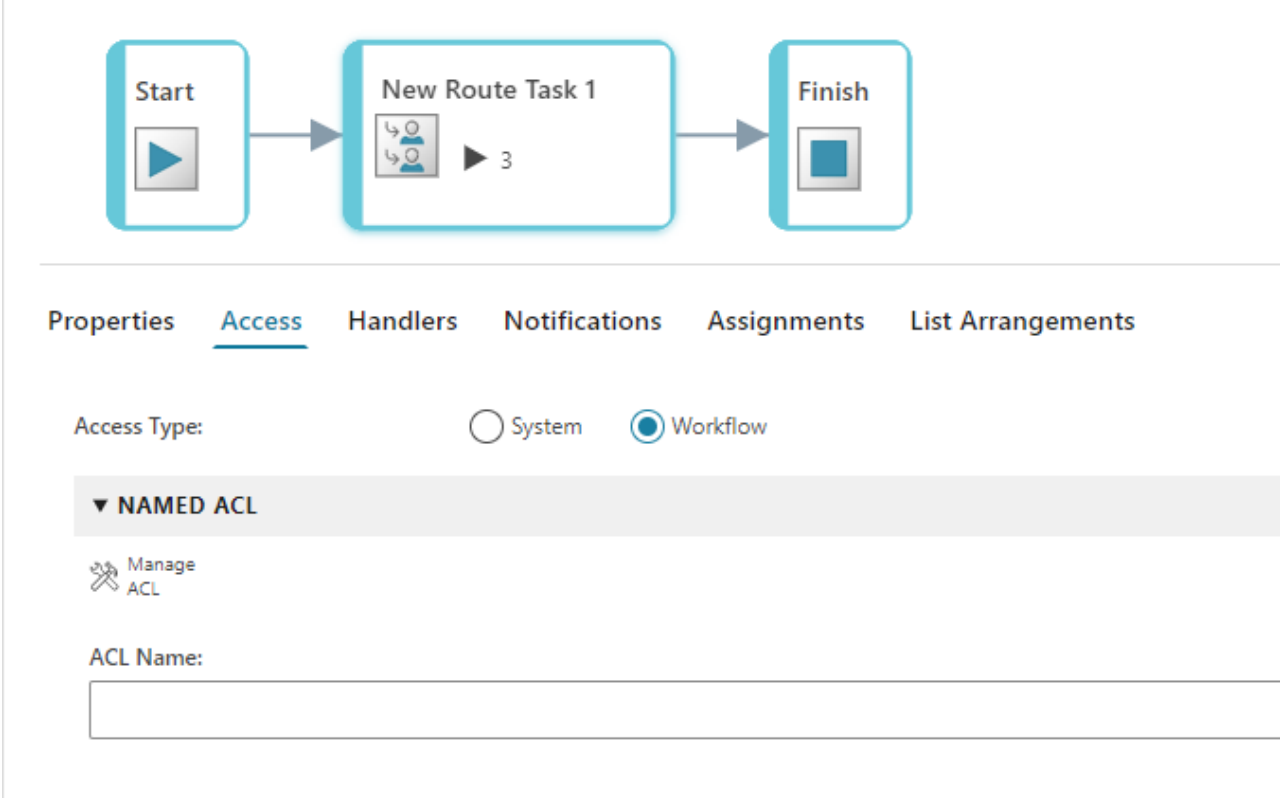
2. Search for and click a workflow process that you want to edit.




3. Go to **More Commands** ▫▫▫ > **Edit** ✎ > **Start Edit** ✎.

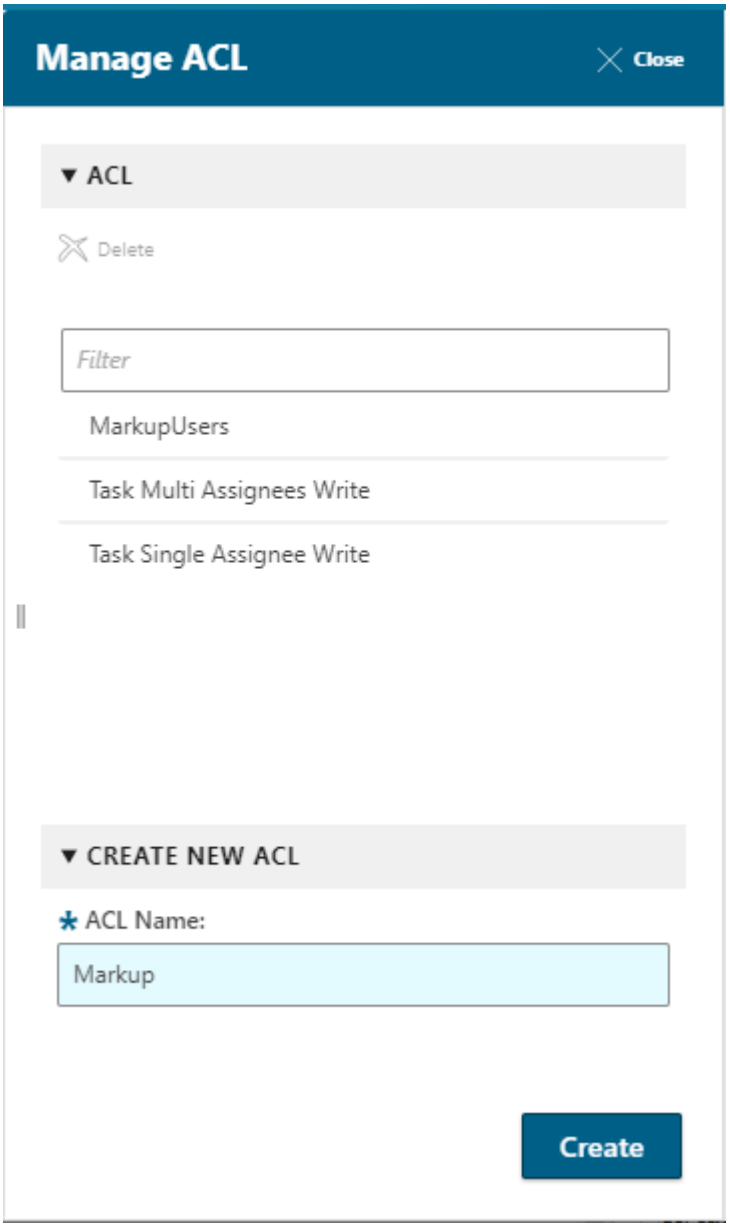



4. In the pop-up message, click **Offline**.




5. Click the **Access** tab.

6. Click **Manage ACL**.



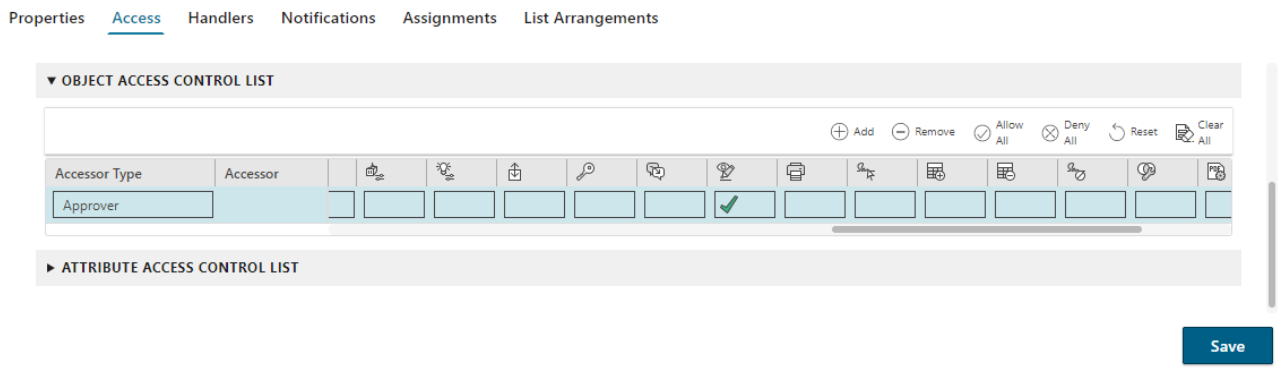

7. In the **Manage ACL** panel, enter a name for the ACL and click **Create**.

8. In the **ACL Name** list, click the newly created ACL.

9. In the **OBJECT ACCESS CONTROL LIST** table, click **Add** ⊕.

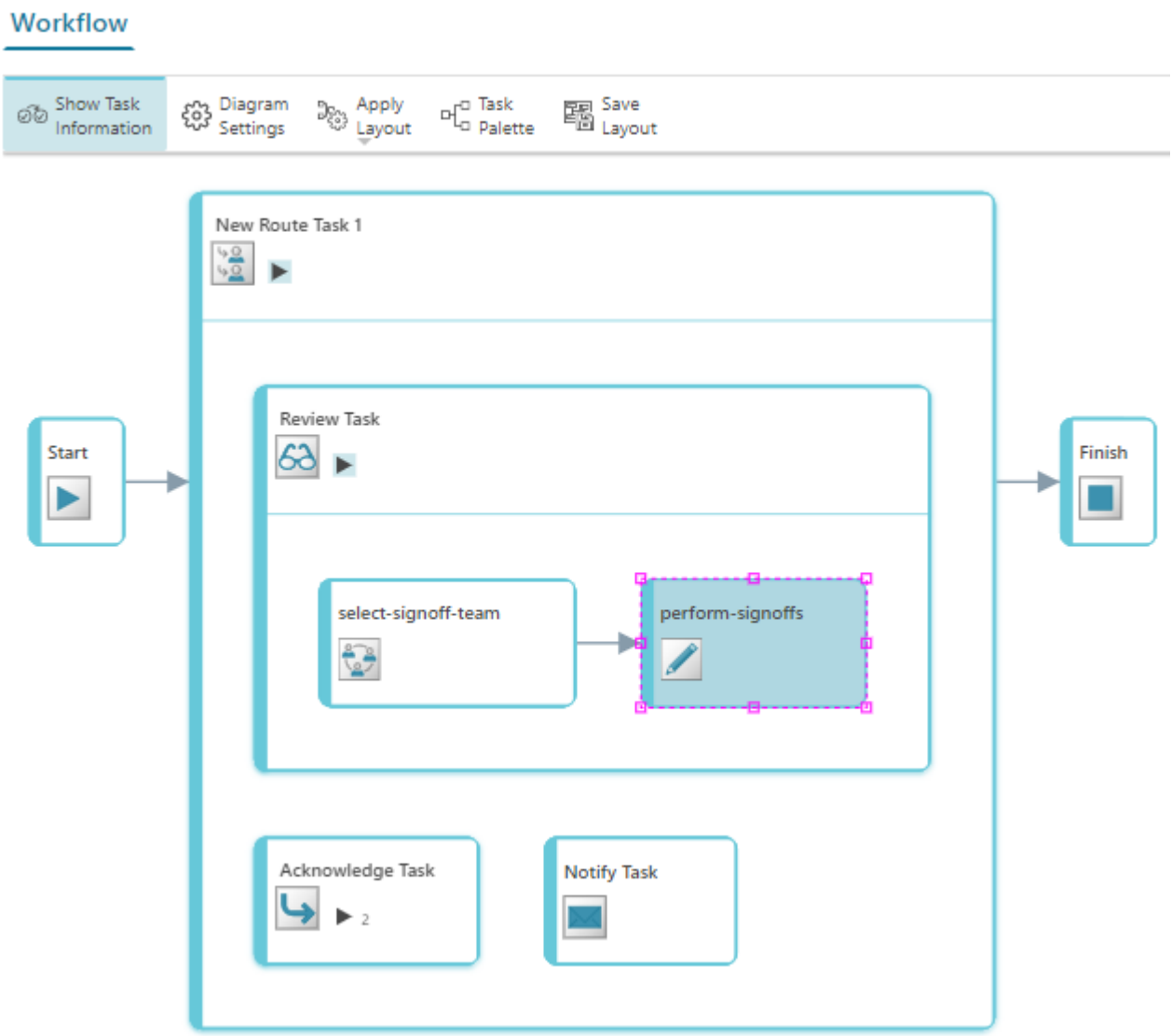

10. Double-click the **Accessor Type** column, and then click **Approver**.

    You can also add users in addition to adding the **Approver**.



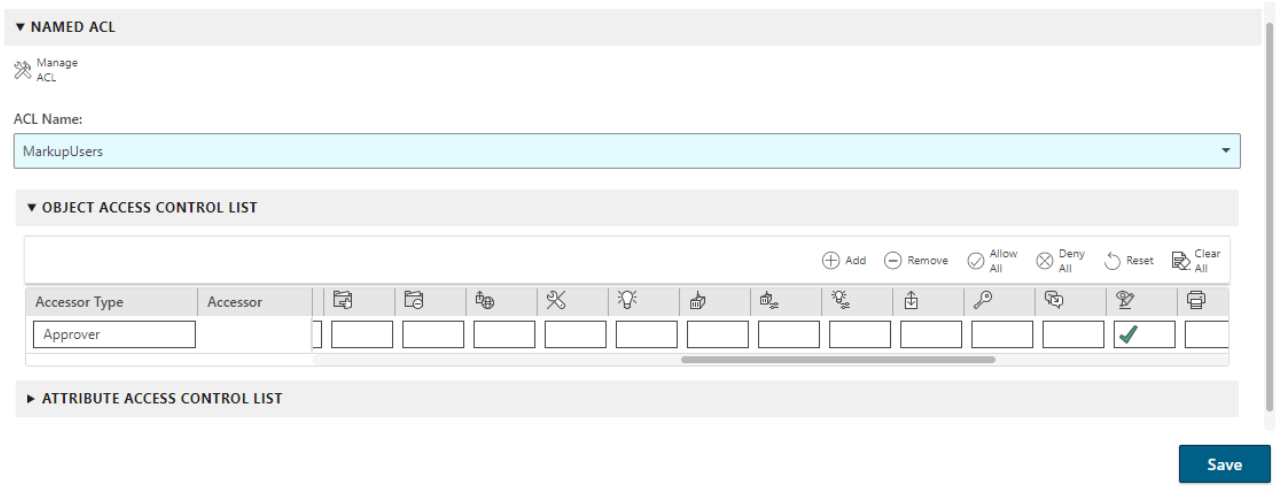

11. Double-click the **Markup** column, and then click ✓.

12. Click **Save**.




13. Expand the workflow, and click the **perform-signoffs** box.

14. Click the **Access** tab, and then click the newly added ACL name in the **ACL Name** list.




15. Click **Save**.

    A pop-up message appears and indicates that the access control entries are saved.

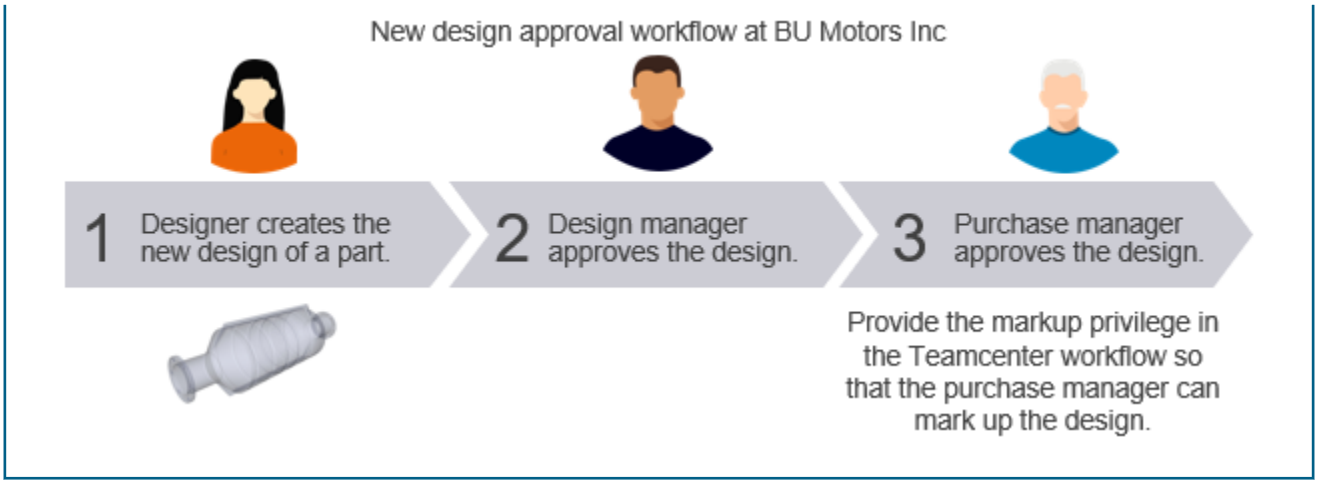
16. Go to **More Commands** ▫▫▫ > **Edit** > **End Edit**.

17. Go to **More Commands** ▫▫▫ > **Edit** > **Online**.

## Provide markup privileges to users in a workflow using rich client

Sometimes users from different groups who are working on a Teamcenter workflow are required to provide markups on a PDF document. Because these users belong to different groups, they might not have the privileges to markup the document. In such a case, as an administrator, you can modify the Teamcenter workflow and provide the markup privileges to the required users.

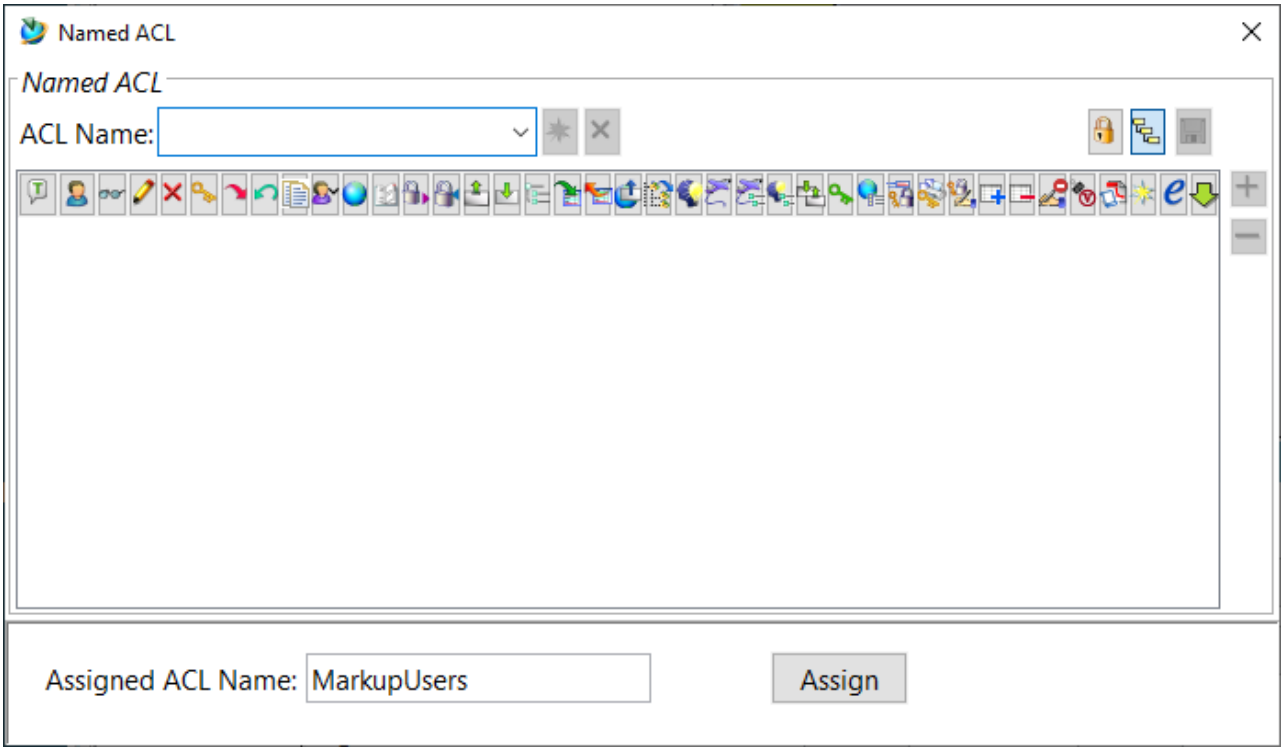
> Example:
>
> In BU Motors Inc., the purchasing manager always provides suggestions and also approves the new design of the part. By using this company-specific process, BU Motors Inc. ensures that the raw material required for manufacturing the new part can be procured within a reasonable time frame. Because the purchasing manager is from another group, as an administrator, you can modify the required Teamcenter workflow and provide the markup privilege to the purchasing manager. This way, all users involved in the Teamcenter workflow, including the purchase manager, can markup the documents.




## Procedure

1. As a database administrator, open the Workflow Designer in rich client.
2. In the **Process Template** list, search for and click the required workflow process.

   For example, click **Route**.
3. Click the top node of the process, and then click **File→Mode→Edit**.
4. On the **Offline?** message window, click **Yes**.
5. In the Workflow Designer, expand **Route→Review Task→perform-signoffs**, and then click **Task Attributes** .
6. In the **Attributes** dialog box, click **Named ACL**.

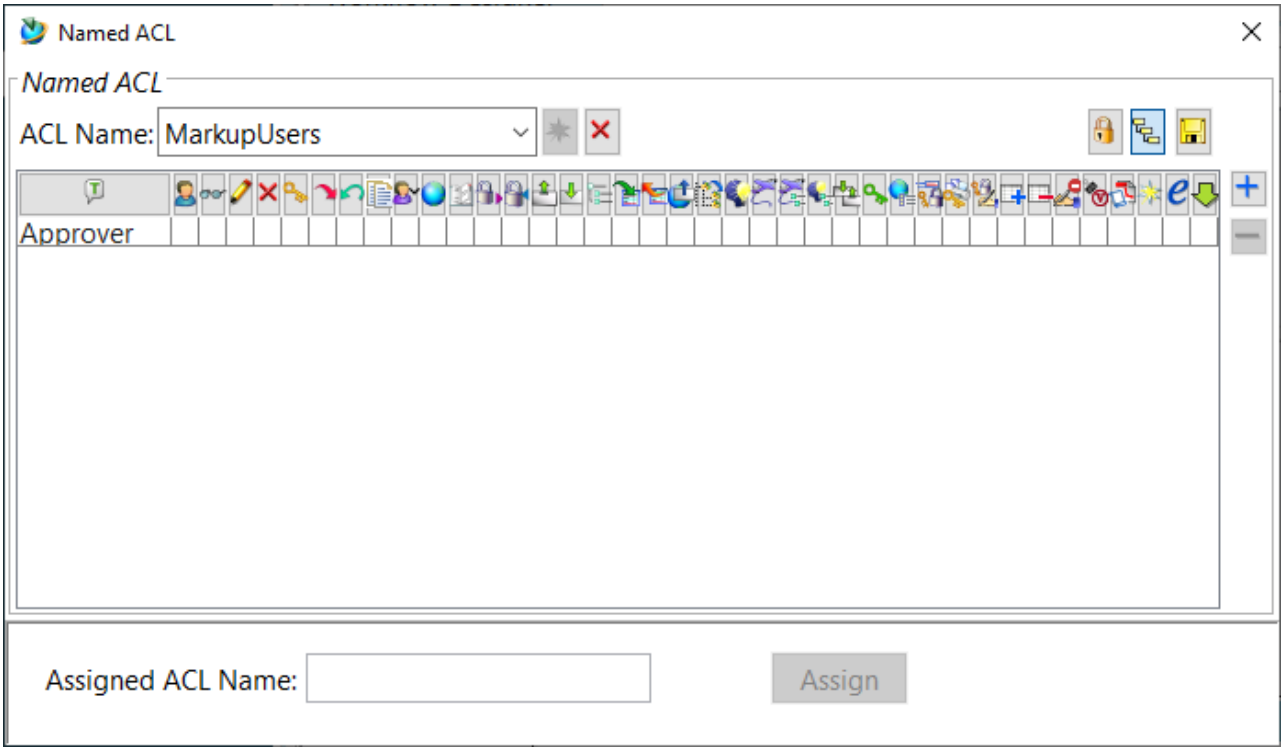



7. In the **Named ACL** dialog box, type a name in the **ACL Name** box, and then click **Save** .

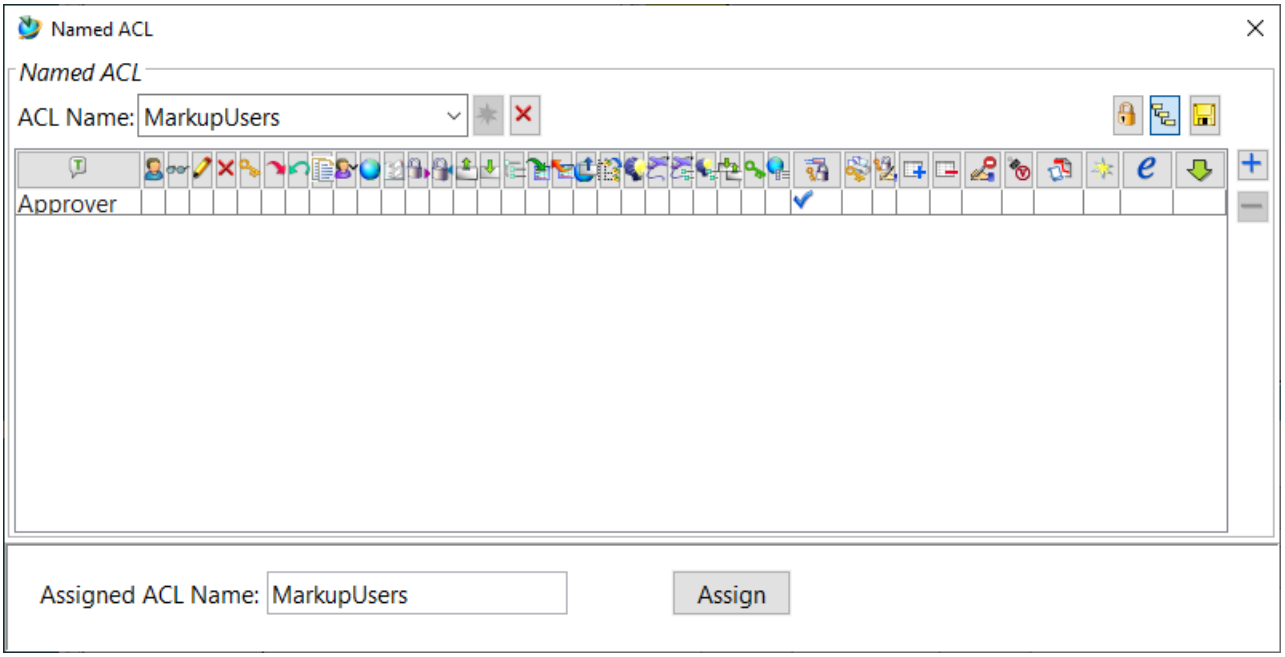
8. Click to add a row to the table below.

9. In the **Type of Accessor** column, double-click and select **Approver**.

   You can also add users in addition to adding the **Approver**.




10. Double-click the **Markup** column, and then click ✔.

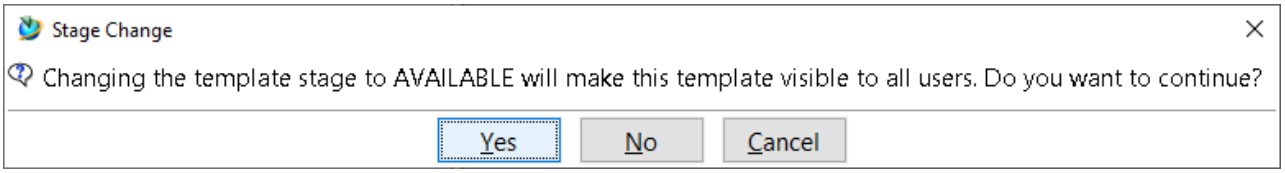



11. Click 💾, and then click **Assign**.

12. Close the **Named ACL** dialog box.

The **Named ACL** box of the **Attributes** dialog box displays the newly created workflow ACL.

13. Close the **Attributes** dialog box, and then click the **Set Stage to Available** check box.




14. Click **Yes**.

## Allow users to download the markup disposition report in Excel

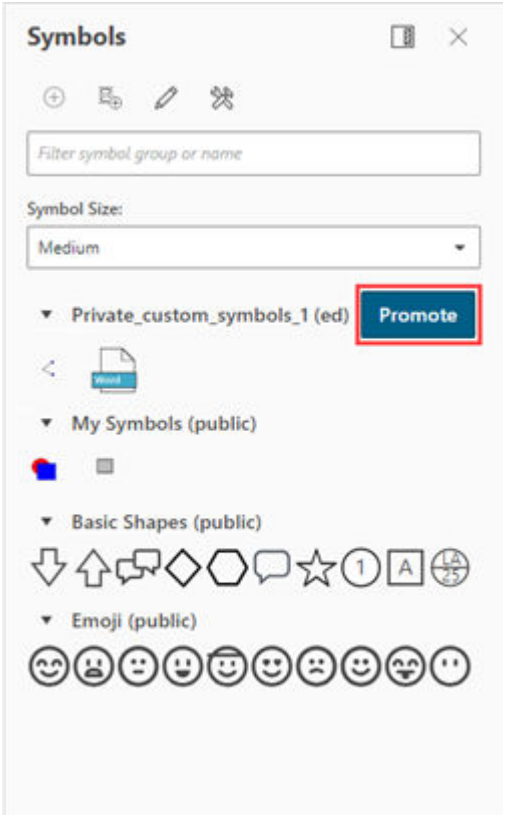
Authors and reviewers can create the markup disposition reports in HTML by default. In addition to HTML, if you set the **Crf_Report_Excel_Use_OOXML** preference to **true**, the authors and reviewers can also create and manage the markup disposition reports in Excel.

## Manage the availability of custom symbols for different users and groups

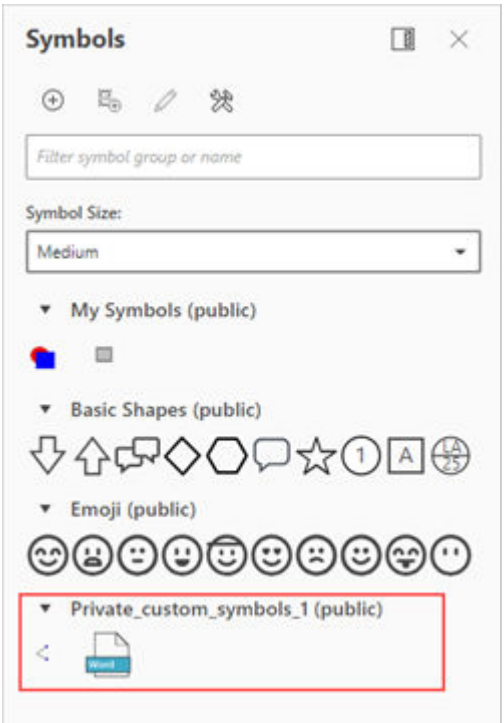
You can manage the availability and use of custom symbols between users. To do this, you can move symbols from one group to another and promote private symbol groups to public so that all the members in the domain can use the symbols in the promoted group.

### Procedure

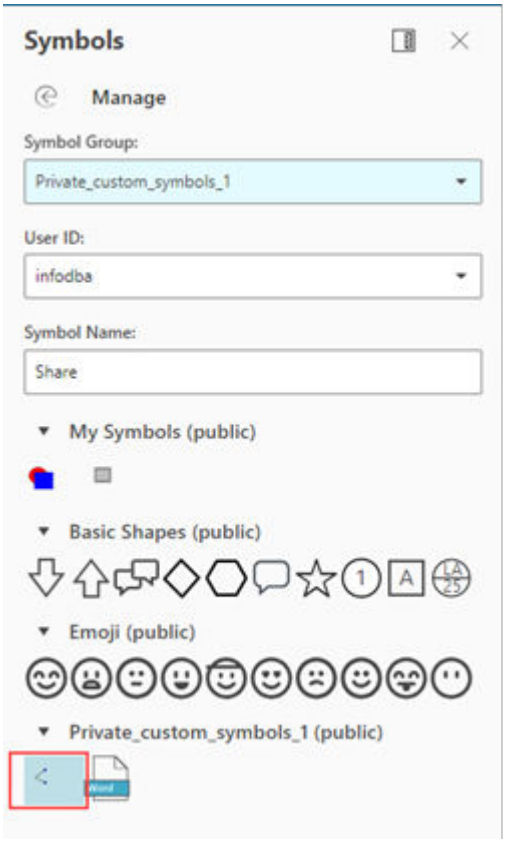
1. Open an image or a PDF file so that it is displayed in **Viewer**.
2. To make the **Symbols** command available, click **Edit Markups**.
3. Click **Symbols** and then click **Manage** .
4. To promote symbols from a private group to public for access to all users, click **Promote**.



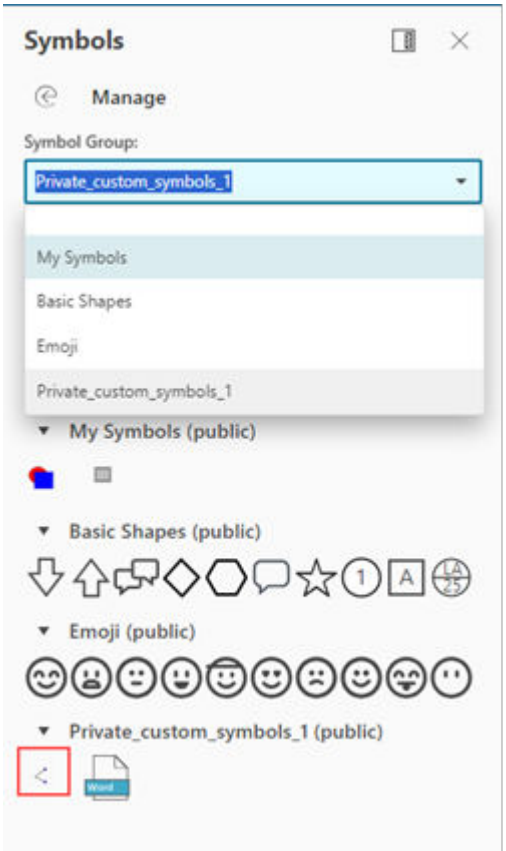

All symbols listed under the private group are now listed as public. Any other user can use these custom symbols to markup documents.



5. To move a custom symbol from one group to a different group, select the symbol.



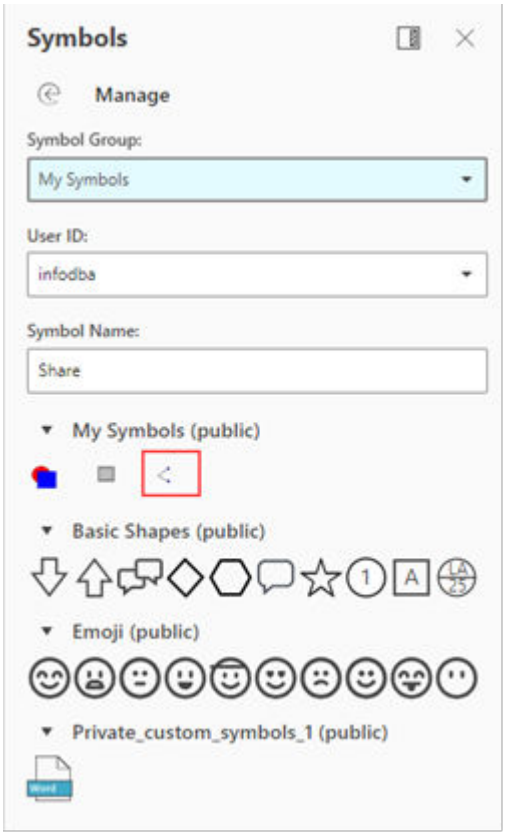

6. From **Symbol Group** list, select the group to which you want to move this symbol to.




7. Specify a new **Symbol Name**, if required.

8. Click **Modify**.

   The symbol is moved under a different group.




9. To save the symbol under the same group but with a different name, do not change the **Symbol Group**, specify a new **Symbol Name**, and click **Modify**.

   Same symbol will be listed under the group with a different name.

# 21. Setting up document printing

## About document printing

You can set up the document printing process such that users can print documents directly from Teamcenter. All documents attached to a document revision can be printed in a batch. You can set up banner pages to group and organize print jobs when documents are printed in a batch. A banner page identifies the end of a document and the start of another documented during printing. Further, you can configure the process to include stamps such as the watermark, logo, user name, date and time, or distribution statement in the printed documents.

To set up document printing:

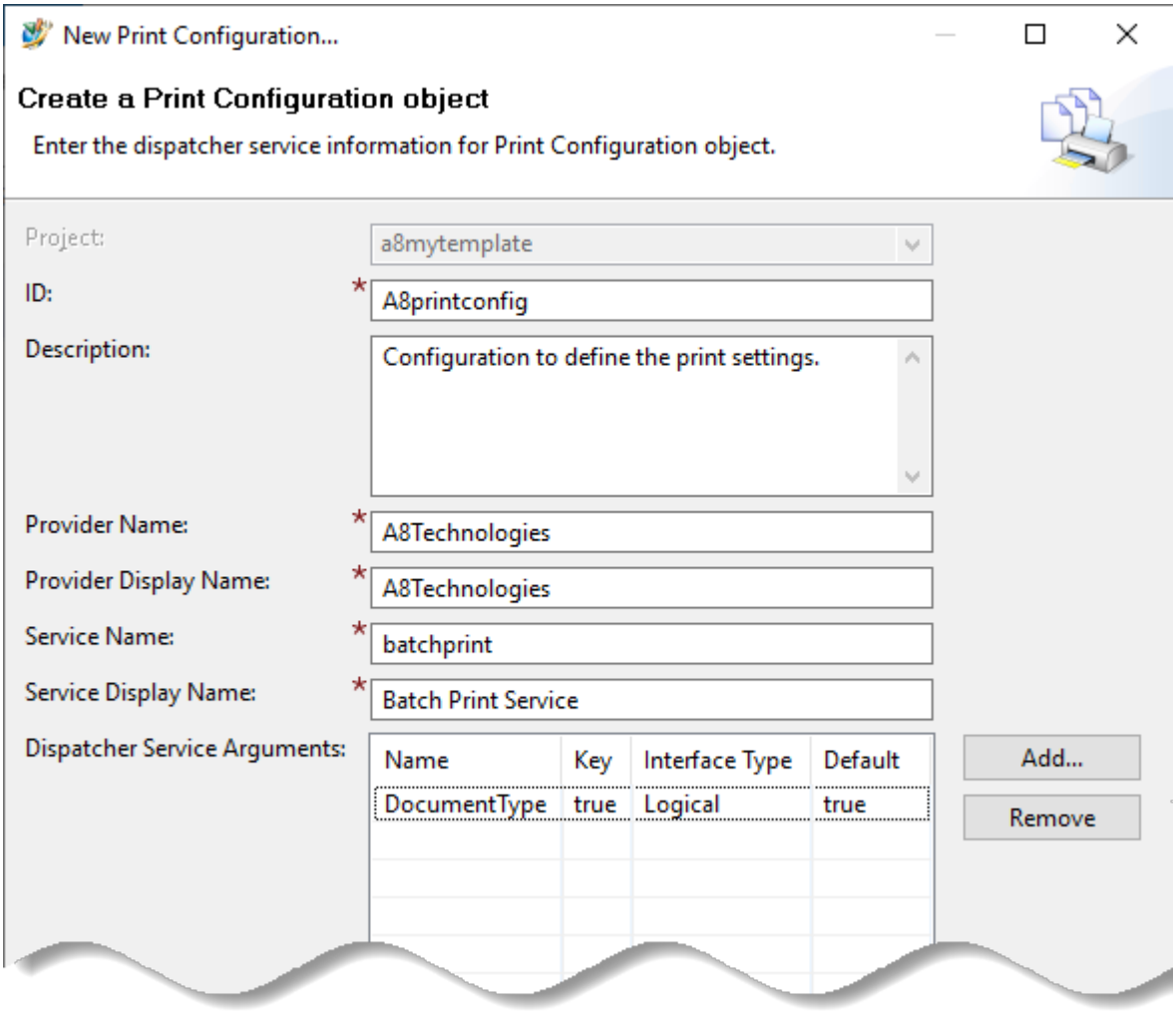
- Ensure that you have **installed Dispatcher** with the **BatchPrint** and **PreviewService** translators.
- Ensure that you have **installed Lifecycle Visualization Convert and Print**.
- Define the print settings by **creating a print configuration**.
- Set the **DM_IsBatchPrintingEnabled** preference to **True** to enable the **Files** print option on Active Workspace **Print** panel.
- **Specify banner pages**, if required.
- **Configure system stamps**, if required.

## Define print settings through a print configuration

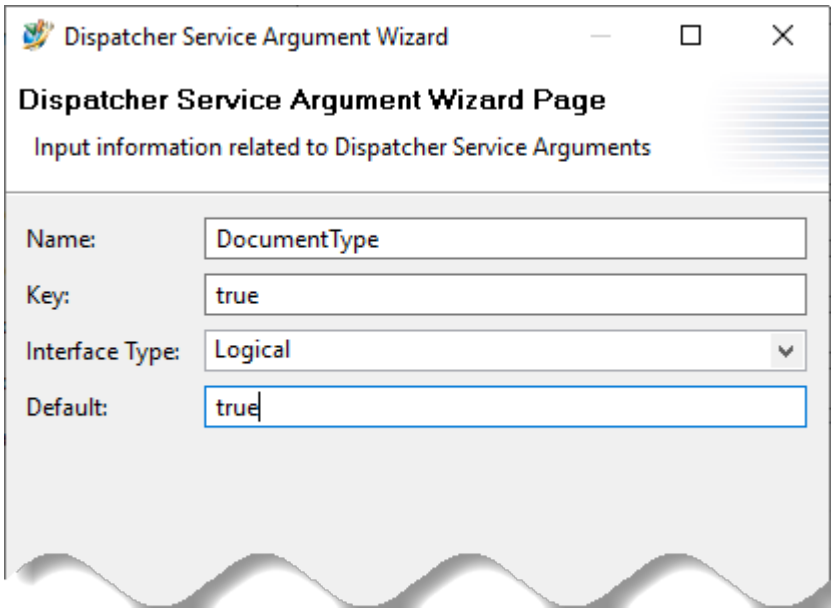
A *print configuration* object defines the print settings. When a user prints a document revision, all documents attached to it are printed. A sample print configuration is available in the *TC_ROOT\samples\document_management\bmide_template\sdm0sampledocmgt* BMIDE template file.

To create a print configuration object:

1. In BMIDE, click **BMIDE→New Model Element**.
2. In the **Add New Model Element** dialog box, select **Print Configuration** in **Wizards** and click **Next**.
3. Enter the following Dispatcher service information for the print configuration and click **Next**:

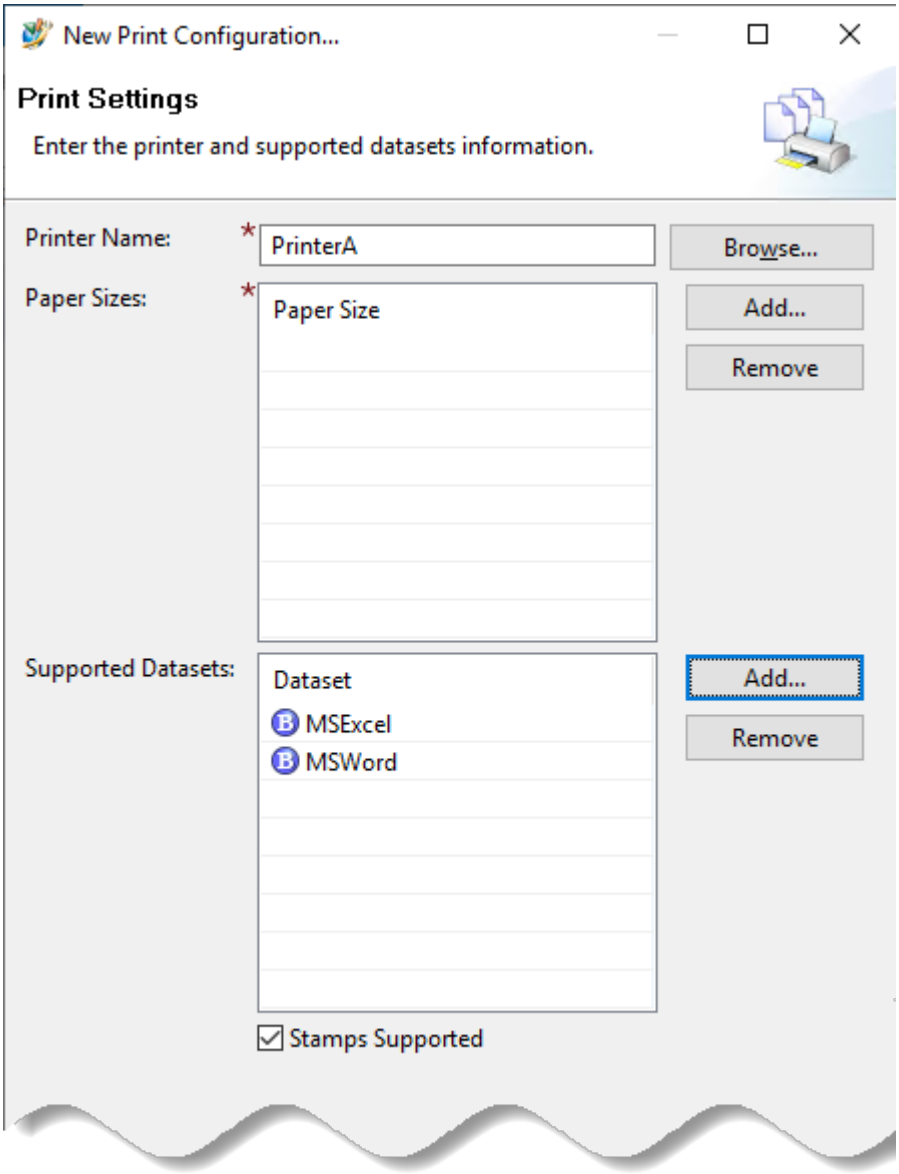



| Field | Action |
|---|---|
| **Provider Name** | Type the name of your organization as the provider of the service, for example, **A4 Technologies**. |
| **Provider Display Name** | Type the name of your organization the way you want it to be displayed on the user interface. |
| **Service Name** | Type the name of the translator that will perform the translation, for example, **batchprint**. |
| **Service Display Name** | Type a name for the translation service the way you want it to be displayed on the user interface, for example, **Batch Print Service**. |
| **Dispatcher Service Arguments** | Click **Add** to add arguments that you want to send to Dispatcher. |

| Field | Action |
| --- | --- |
| |  |



4. Enter the following print settings and click **Finish**:




| Field | Action |
|---|---|
| **Printer Name** | Select a network printer. To obtain the available printer names, Business Modeler IDE must be installed on the same workstation as Lifecycle Visualization. |
| **Paper Size** | Add the supported paper sizes. |
| **Supported Datasets** | Add the supported document file formats that can be printed, for example, **MSWord**. |

5. Ensure that the print configuration object is displayed in the **Print Configuration** folder.

6. Deploy the template from the Business Modeler IDE.

7. Restart Pool Manager and restart the Teamcenter server.

8. Test the batch print.

   a. Ensure that document rendering is installed for the rich client. (The **batch print menu** action is included with the render action.) In the **Features** panel of Teamcenter Environment Manager (TEM), choose **Extensions**>**Enterprise Knowledge Foundation**>**Dispatcher Client for Rich Client** and **Render Document for Rich Client**.

   b. In the rich client, in My Teamcenter, select an object with the associated documents, such as an item revision, and choose **Translation**→**Batch Print**.

      Your print configuration is displayed in the **Print Configuration** menu. Click **Finish** to print the documents.

   c. To monitor printing, choose **Translation**>**Administrator Console**. Press Shift + F5 to refresh the view.

## Specify banner pages

Banner pages are used to separate and organize print jobs. For example, you can use banner page when numerous print jobs are sent to one or more printers, and you want to distinguish where one job ends and another begins.

To specify banner pages:

1. Open the Lifecycle Visualization print configuration file (**vvcp.ini** file on Windows systems and **vvcp.***platform***.cfg** on Linux systems).

   This configuration file is created in the **...\Visualization\VVCP** directory on Windows systems (or the **visualization/app_defaults** directory on Linux systems) when Convert and Print is installed by the Lifecycle Visualization installer or by Teamcenter Environment Manager (TEM).

2. Set the following attributes:

   - **Banner**

     Specifies banner pages for all printers.

   - *Printer***Banner**

     Specifies the banner page for a specific printer. This overrides the **Banner** setting for the specified printer.

   - **BannerFormat**

Specifies either text (TXT) or metadata stamping (MDS) format for banner page processing.

- **BannerFormat=mds** (**BannerFormat: mds** on Linux systems)

  The banner page is a stamp file printed without an underlying image. All the MDS commands are supported except **\File** and **\FilePath**.

- **BannerFormat=txt**

  The banner page content comes from a text file.

  TXT format supports only the following MDS commands:

  **\Attribute\Copies\Date\Time**

> Note:
>
> You must not provide values for **BannerInfo** and *Printer***BannerInfo**. Teamcenter provides these values.

For information about configuring banner page printing, see the *Introduction to Convert and Print* guide provided with the Lifecycle Visualization application Convert and Print software.

## Provide access privileges for batch printing

Teamcenter provides access control lists (ACLs) for managing batch printing information.

The following ACLs are located in Access Manager under **Has Class(POM_application_object)**:

- **Batch Print Item**

  Batch print action privileges for items.

  **Working→Has Class(Item)→Batch Print Item**

- **Batch Print Item Revision**

  Batch print action privileges for item revisions.

  **Working→Has Class(Item Revision)→Batch Print Item Revision**

- **Batch Print Dataset**

  Batch print action privileges for datasets.

**Working→Has Class(Dataset)→Batch Print Dataset**

You can modify these ACLs or create others using **Access Manager**.

> Note:
>
> Commercial off-the-shelf (COTS) ACLs do not grant **World** batch print access privilege. To print dataset files you do not own, the ACLs must be modified to grant batch print access privileges.

# 22. Setting up stamps for rendered or printed files, and existing PDFs

## About setting up stamps

### What are stamps?

In Document Management, the following elements are categorized as stamps:

- Logos
- Distribution statements (for example, Export Controlled ITAR)
- Teamcenter attributes such as document name, description, and date and time.
- *System stamps* such as watermarks, user names, and date and time information.
- Workflow signoff information, such as a workflow template name, group, role, user name, user ID, and comments. This signoff information is the one the author sees in the **Workflow Signoff** dialog box while sending a document revision through a review workflow.

### How to set up stamps?

- **Attach the logo and distribution statement files to the Document Management template revision** for stamps. You can attach multiple logo and distribution statement files.
- Specify the logos, distribution statements, and workflow signoff tables to be displayed in the stamped files in the **Document Management configuration XML file**. Attach this file to the Document Management template revision for stamps.
- Define the styling of the stamps. For example, you can set the font and the position of the watermark, or the position of the logo and workflow signoff table.
  - For rendered or printed files, define the **styling in a MetaData Stamp file** and attach the file to the Document Management template revision for stamps.
  - For existing PDF files, **update the PDF command file** attached to the Document Management template revision for stamps.
- Set up stamps by **creating a system stamp configuration**. You can choose to include these stamps when a file is only rendered, only printed, both rendered and printed, or in existing PDFs.

### Where to include stamps?

After setting up stamps, you can choose to include them in:

- Rendered or printed files while setting up **document rendering** and **printing**.
- Microsoft Office files (Word, Excel, and PowerPoint) and PDFs by **setting up attribute exchange through logical objects**.
- Existing PDF files when authors send them through a stamping workflow. For this, you must **set up a workflow process template** by using the **DOCMGTAPP-apply-pdf-control** action handler.

> Tip:
>
> To **improve the output quality of stamped PDF files**, update the Lifecycle Visualization *vvcp.ini* file.

## Specify the logos, distribution statements, and workflow signoff tables to be displayed in the stamped files

You specify information about logos, distribution statements, and workflow signoff tables in the Document Management configuration XML file. You may refer to the Document Management configuration schema file (**DocMgmtConfig.xsd**) located at *TC_ROOT\sample\document_management\schema* to create the configuration XML file. In this file, you can specify the logos to be used, when to use a logo, and where to place the logo in a PDF file. You can also specify similar information for distribution statements, workflow signoff tables, and Teamcenter attributes. After specifying the required information, you must **attach the configuration XML file** to a Document Management template revision for stamps.

For more information about workflow signoff table, see *server-install-location\samples\document_management\readme.pdf*.

### A snippet of a list of columns in the workflow signoff table

You can define the columns to be displayed in a workflow signoff table. To do so, add the column names in the Document Management configuration XML file:

```
...
<!--workflow signoff table column names to display, can be removed or
reordered
These names listed below are internal names, not the display names-->

<dmcfg:WorkflowSignOffColumnsToDisplay>
   <dmcfg:WorkflowSignoffColumnName>processName</
dmcfg:WorkflowSignoffColumnName>
   <dmcfg:WorkflowSignoffColumnName>group</dmcfg:WorkflowSignoffColumnName>
```

```
   <dmcfg:WorkflowSignoffColumnName>role</dmcfg:WorkflowSignoffColumnName>
   <dmcfg:WorkflowSignoffColumnName>userName</
dmcfg:WorkflowSignoffColumnName>
   <dmcfg:WorkflowSignoffColumnName>userId</dmcfg:WorkflowSignoffColumnName>
   <dmcfg:WorkflowSignoffColumnName>decisionStringDisplay
   </dmcfg:WorkflowSignoffColumnName>
   <dmcfg:WorkflowSignoffColumnName>decisionDateString
   </dmcfg:WorkflowSignoffColumnName>
   <dmcfg:WorkflowSignoffColumnName>comments</
dmcfg:WorkflowSignoffColumnName>
</dmcfg:WorkflowSignOffColumnsToDisplay>
...
```

By default, all columns are displayed in the workflow signoff table. You can add, remove, or reorder the columns as required.

### A snippet of the logo and distribution statement information

```
...
<dmcfg:logoList>
   <dmcfg:logoDataset datasetName="Logo" ID="1">
      <dmcfg:enableWhen>
         <dmcfg:property key="fnd0CurrentLocationCode" value="Shoreview" />
      </dmcfg:enableWhen>

      <dmcfg:MSOfficePlacement>
         <dmcfg:tag value="Logo_1" />
      </dmcfg:MSOfficePlacement>
   </dmcfg:logoDataset>
</dmcfg:logoList>

<dmcfg:distStatementList>
   <dmcfg:distStatementDataset datasetName="ExportControlledITAR" ID="1" />
      <dmcfg:enableWhen>
         <dmcfg:property key="fnd0CurrentLocationCode" value="Shoreview" />
      </dmcfg:enableWhen>

      <dmcfg:MSOfficePlacement>
         <dmcfg:tag value="Statement_1" />
      </dmcfg:MSOfficePlacement>


   <dmcfg:distStatementDataset datasetName="NonExportControlledITAR"
ID="2" />
      <dmcfg:enableWhen>
         <dmcfg:property key="fnd0CurrentLocationCode" value="Cypress" />
      </dmcfg:enableWhen>
```

```
      <dmcfg:MSOfficePlacement>
         <dmcfg:tag value="Statement_2" />
      </dmcfg:MSOfficePlacement>

   </dmcfg:distStatementDataset>
</dmcfg:distStatementList>
...
```

## Attach files to a Document Management template revision for stamps

To attach the logo, distribution statement, Document Management configuration XML file, PDF command file, and MetaData Stamp file to a Document Management template revision that you created for stamps:

1. In My Teamcenter, select the required system stamp Document Management template revision.

2. Click **File→New→Dataset**.

3. In the **New Dataset** dialog box:

   a. Enter the **Name** of the dataset as the name of the logo or distribution statement specified in the Document Management configuration XML file. For example, the **datasetName** of a logo in the configuration XML file is **Logo_1**. Therefore, the name of the dataset for the logo must also be **Logo_1**.

   b. Import the logo image file, distribution statement text file, or Document Management configuration XML file.

   c. In **Relation**, select **Document Page** while attaching the logo, distribution statement, or Document Management configuration XML file.

      For PDF command and MetaData Stamp files, select **Tc_Attaches**.

   d. Click **OK**.

4. Right-click the newly created dataset and click **Properties On Relation**.

5. In the **Properties** dialog box:

   a. Click **Show Empty Properties** to view all properties.

   b. In **Page Type**, select **Logo**, **Distribution Statement**, or **Document Configuration** based on the entity you are creating the dataset for.

   c. Click **OK**.

6. Select the dataset and click **File→New→Workflow Process** to start a workflow process to release the dataset.

7. To release the dataset, in the **New Process Dialog** dialog box:

   a. Select **TCM Release Process** in **Process Template**.

   > Note:
   >
   > The default value of the release status for Document Management template (*DMTemplate*) is **TCM Released**. This value is set to *MaturityStatuses* constant in BMIDE. You can customize it by adding a status type. You can also understand what are global constants, business objects constants, and how to run data model reports to understand the *MaturityStatuses* constant.

   b. In the **Attachments** tab, expand **Targets** to select the logo, distribution statement, or Document Management configuration schema file that is attached to the dataset.

   c. Click **OK**.

   A release flag is displayed next to the released dataset.

## Define the styling of stamps for rendered and printed files

To define the styling of stamps for rendered and printed files, you set a *MetaData Stamp* (MDS) file. In this file, you can specify the styling, such as the font and position of the watermark. You can also specify the position of the logo and the workflow signoff table.

To set an MDS file, you must do one of the following:

- Create your own MDS template.
- Download the **stampinfo.mds** sample and default **batchprint.mds** from DMTemplates with MDS as a prefix for its name (such as, **MDS_docmgt_sample_stampinfo** and **01-DM-print**). Review its attached datasets named reference file.

  The sample DMTemplate is provided in the DMTemplates directory and is loaded to Teamcenter after the Sample Document Management (sdm0sampledocmgt) is installed at the following path:

  *TC_ROOT/sample/document_management/bmide_template/sdm0sampledocmgt/install/sampledata/DMTemplates*

To update an existing MDS file with stamping information:

- Check out the required dataset.

- Click **View→Named References**.
- In the **Named References** dialog box, click **Download** to download the sample MDS file.
- Modify the file with the required stamping information. Use **\Attribute** to define all stamps that you want to include.

  **Example**

```
\RemoveAllStampsSubmitted by
\Attribute(FullUserName) on
\Attribute(DateAndTime)
\Attribute(DistStmtText_1)
\Attribute(DialogUserStamp)
\Pos(36,-72)\Char(8)
OS User: \Attribute(OSUserName)
\Attribute(SystemStamp)
\Font(Courier)
\Char(10)
\AutoPos(5,366)
\Attribute(WorkflowSignOffTable)
\Font(Arial)
\Pos(-170,-72)
Teamcenter User: \Attribute(TCUserName)
\Attribute(UserStamp)
\Watermark(LL2UR,AUTO)
\Attribute(Watermark)
\Logo(\Attribute(LogoFilePath_1), 292, 152)
```

| Attribute | Description |
|---|---|
| **LogoFilePath_1** | Specifies the path of the logo specified in the PDF command XML file. If there are multiple logos specified in the Document Management configuration file, add the `\Logo` attribute for each logo. Also specify the ID of the logo after the underscore character, for example, **LogoFilePath_1**. You can obtain this ID from the **Document Management configuration XML file**.<br><br>You can use Teamcenter Visualization Convert to determine the size of the logo in order to specify the *x* and *y* coordinates accurately. |
| **DistStmtText_1** | Specifies the distribution statement. To include multiple distribution statements that you specified in the **Document Management configuration XML file**, specify the ID of each |

| Attribute | Description |
| --- | --- |
| | distribution statement after the underscore character, for example **DistStmtText_1** |
| **WorkflowSignOffTable** | Specifies the workflow signoff information when a document revision is sent through a review workflow process.<br><br>By default, the workflow signoff table is added to all pages. To include the table only on the first page, update the *Teamcenter Visualization vvcp.ini* file to set **StampPage=first** in the **Prepare** section. However, for rendered or printed PDFs, stamps are included in all pages. The PDF dataset must be related to a document revision. |

- (Optional) To remove previously applied text stamps, use the **\RemoveAllStamps** command. Include this command before any **\Attribute** command.
- (Optional) To move the vertical position of a text stamps when the text stamps is applied to a PDF file that is previously stamped with a text stamp at the same position, use the **\AutoPos** command. This command prevents text stamps from overwriting previously applied text stamps.
- After making the required changes, first delete the existing named reference file and ensure that there is only one named reference file available. Then, click **Upload** to upload the updated MDS file.
- Check in the dataset.

## Define the styling of stamps for existing PDF files

To define the styling of the stamps in existing PDFs, you update the PDF command file attached to the Document Management template revision. Styling a stamp includes specifying the font and position of a watermark, or the position of a logo and the workflow signoff table. The stamps are applied to a PDF when it is sent through a stamping workflow. To define the styling:

1. In My Teamcenter, check out a dataset that has the PDF command XML file attached to it. Files are attached to a dataset as *named references*.
2. Click **View→Named References**.
3. In the **Named References** dialog box, select the attached file and click **Download** to download the files to a specific location.
4. Update the downloaded file to specify the style and placement of the stamps. Include the required stamps in the **propertyDefinitions** command:

```
...
<cmd:propertyDefinitions>
   <cmd:propertyDefinition key="FullUserName" />
   <cmd:propertyDefinition key="DateAndTime" />
   <cmd:propertyDefinition key="SystemStamp" />
   <cmd:propertyDefinition key="Watermark" />
   <cmd:propertyDefinition key="LogoFilePath_1" />
   <cmd:propertyDefinition key="DistStmtText_1" />
...
```

Specify the Teamcenter attributes that must be added to the PDF file. These are the attributes that were defined and related to the PDF dataset by using logical objects.

```
...
<cmd:propertyDefinition key="fnd0Root_object_name" />
<cmd:propertyDefinition key="fnd0Root_object_desc" />
<cmd:propertyDefinition key="m1DocRev_item_revision_id" />
...
```

> Note:
>
> You can refer to the **commands available for styling of stamps** and then update the XML file.

5. Specify the text and its styles, such as font and position, in the **textStamp** command.

**Example 1**

The following snippet shows the styles applied to the *User Name* and *Date and Time* stamps:

```
...
<cmd:textStamp>
   <cmd:text>Submitted by \FullUserName on \DateAndTime</cmd:text>
   <cmd:font>TimesNewRoman</cmd:font>
   <cmd:size>12</cmd:size>
   <cmd:position>
      <cmd:coordinates y="36" x="36" />
   </cmd:position>
<cmd:zOrder>1</cmd:zOrder>
   <cmd:additionalText>
      <cmd:newline/>
   </cmd:additionalText>
      <cmd:text>\UserStamp</cmd:text>
   </cmd:additionalText>
</cmd:textStamp>
...
```

The following snippet shows the styles applied to a distribution statement. If you want to place multiple distribution statements, specify the ID after the underscore character (for example, **\DistStmtText_1**). You can **get the ID from the Document Management configuration XML file**. Include the backslash character in front of the property name to retrieve its value.

```
...
<cmd:textStamp>
   <cmd:text>\DistStmtText_1</cmd:text>
   <cmd:font>TimesNewRoman</cmd:font>
   <cmd:size>10</cmd:size>
   <cmd:position>
      <cmd:coordinates y="138" x="36" />
   </cmd:position>
   <cmd:pages>
      <cmd:pageRange endPage="1" startPage="1" />
   <cmd:pages>
<cmd:textStamp>
...
```

The following snippet shows the styles applied to a logo. If you want to place multiple logos, specify the ID after the underscore character (for example, **LogoFilePath_1**). You can get the **ID from the Document Management configuration XML file**.

```
...
<cmd:logo>
   <cmd:logoFilePathKey>LogoFilePath_1</cmd:logoFilePathKey>
      <cmd:position>
         <cmd:coordinates x="-170" y="72" />
      </cmd:position>
      <cmd:pages>
         <cmd:pageRange startPage="1" endPage="1"/>
      </cmd:pages>
      <cmd:zOrder>0</cmd:zOrder>
</cmd:logo>
```

**Example 2**

The following snippet shows the styles applied to the workflow signoff table:

```
...
<cmd:textStamp>
   <cmd:text>\WorkflowSignoffTable</cmd:text>
   <cmd:font>Courier</cmd:font>
   <cmd:size>10</cmd:size>
   <cmd:position>
      <cmd:coordinates y="366" x="5" />
   </cmd:position>
   <cmd:additionalText>
      <cmd:newline/>
   </cmd:additionalText>
</cmd:textStamp>
...
```

The workflow signoff table consists of the sign off information contained in the **Workflow Signoff** dialog box when a document revision is sent through a review workflow.

The following snippet shows the styles applied to the Teamcenter attributes:

```
...
<cmd:textStamp>
   <cmd:text>Dataset name:\fnd0Root_object_name
Description:\fnd0Root_object_desc
DocRevID:\m1DocRev_item_revision_id
   </cmd:text>
   <cmd:font>TimesNewRoman</cmd:font>
   <cmd:size>10</cmd:size>
   <cmd:position>
      <cmd:coordinates y="36" x="255" />
   </cmd:position>
</cmd:textStamp>
...
```

6. (Optional) To specify the pages in which stamps must be included, use the **pages** command.

   For all pages:

```
...
<cmd:pages>
   <cmd:allPages/>
</cmd:pages>
...
```

For even pages:

```
...
<cmd:pages>
   <cmd:evenPages/>
</cmd:pages>
...
```

For odd pages:

```
...
<cmd:pages>
   <cmd:oddPages/>
</cmd:pages>...
```

For a specific range of pages:

```
...
<cmd:pages>
   <cmd:PageRange startPage="1" endPage="3"/>
</cmd:pages>...
```

7. (Optional) To remove previously applied text stamps, use the **removeStamp** command.

```
...
<!--remove previously applied text stamps-->
<cmd:removeStamp>
   <cmd:text>*</cmd:text>
</cmd:removeStamp>
...
```

8. (Optional) Use the **autoposition** command to move the vertical position of a text stamp when the text stamp is applied to a PDF file that is previously stamped with a text stamp at the same position. This command prevents text stamps from overwriting previously applied text stamps.

```
...
<cmd:textStamp>
   <cmd:text>\DistStmtText_1</cmd:text>
      <cmd:font>Courier</cmd:font>
      <cmd:size>10</cmd:size>
      <cmd:position>
          <cmd:coordinates x="36" y="40" />
      </cmd:position>
      <cmd:autoposition/>
</cmd:textStamp>
...
```

9. The various commands such as **autoposition**, **textStamp**, and **removeStamp** are case-sensitive. After making the required changes, first delete the existing named reference file and ensure that there is only one named reference file available. Then, click **Upload** to upload the updated file.

## Commands available for styling of stamps

When you are styling the stamps, you must adhere to the *<Teamcenter_Installation_Directory>\sample\document_management\schema\PDFCommandTemplate.xsd* XML schema file. The *PDFCommandTemplate.xsd* file also contains the commands that are available for you to create an XML file for formatting or styling the stamps. After you create an XML file that contains the styling information of the stamps, you can **attach it as PDF command XML file to the dataset**. The files are attached to the dataset as *named reference*. The

### Commands that explain different types of stamps

The following code explains the different types of stamps that you can use in the PDF.

```
<xs:element name="textStamp" type="cmd:textStampCmdType" />
<xs:element name="removeStamp" type="cmd:removeStampCmdType" />
<xs:element name="watermark" type="cmd:watermarkCmdType" />
<xs:element name="insertDocument" type="cmd:insertDocCmdType" />
<xs:element name="logo" type="cmd:logoCmdType" />
<xs:element name="image" type="cmd:imageCmdType" />
<xs:element name="drawLine" type="cmd:drawLineCmdType" />
<xs:element name="drawCircle" type="cmd:drawCircleCmdType" />
```

### Commands available for text stamp

The following code and the comments in it explain the available commands when you want to use text stamps.

```
<!-- the beginning of the sequence contains the elements that define the
start of the text stamp -->
<xs:element name="text" type="xs:string" minOccurs="1" maxOccurs="1" />
<xs:element name="font" type="xs:string" minOccurs="1" maxOccurs="1" />
<!-- size is measured in points (1/72 inch) -->
<xs:element name="size" type="xs:positiveInteger" minOccurs="1"
maxOccurs="1" />
<!-- opacity is a percentage, ranging from 0.0 (fully transparent) to
1.0 (fully opaque); if not specified,
it defaults to 1.0 (100% opaque) -->
<xs:element name="opacity" type="xs:float" minOccurs="0" maxOccurs="1" />
<!-- if textColor is not specified, it defaults to black -->
<xs:element name="textColor" type="cmd:colorType" minOccurs="0"
maxOccurs="1" />
<!-- if backgroundColor is not specified, it defaults to no background
coloration -->
```

```
<xs:element name="backgroundColor" type="cmd:colorType" minOccurs="0"
maxOccurs="1" />
<!-- if this element is present, the text will be rendered as an
outline, with empty space inside the lines,
instead of as solid characters -->
<xs:element name="drawAsOutline" type="cmd:emptyElementType"
minOccurs="0" maxOccurs="1" />
<!-- the position coordinate space is measured in points (1/72 inch);
the origin is in the upper-left corner
of the page; negative values can be used to place text relative to the
right or bottom edges -->
<xs:element name="position" type="cmd:positionType" minOccurs="1"
maxOccurs="1" />
<!-- the autoposition command automatically advances the vertical
position of a text stamp when the text stamp
is appled to a PDF file that has been previously stamped with a text
stamp applied at the same position; this command may be used to prevent
text stamps from overwriting previously applied text stamps; use this
command only with unrotated text stamps -->
<xs:element name="autoposition" type="cmd:emptyElementType"
minOccurs="0" maxOccurs="1" />
<!-- the Z-order specifies the order in which overlapping items (text,
images, etc.) are stacked; a higher-number
item in the Z-order will be placed on "top" of lower-number item, with 0
being the lowest value -->
<xs:element name="zOrder" type="xs:positiveInteger" minOccurs="0"
maxOccurs="1" />
<!-- if the user wants to continue adding text but change the font or
font size, or trigger a carriage return,
they can use the following element -->
<xs:element name="additionalText" minOccurs="0" maxOccurs="unbounded">
    <xs:complexType>
        <!-- this element is either a newline, or additional text that
will be placed after the previous block of text -->
          <xs:choice>
            <xs:sequence>
              <xs:element name="text" type="xs:string" minOccurs="1"
maxOccurs="1" />
              <!-- if font, size, textColor, etc. do not change, the
previous values will be used -->
              <xs:element name="font" type="xs:string" minOccurs="0"
maxOccurs="1" />
              <xs:element name="size" type="xs:positiveInteger"
minOccurs="0" maxOccurs="1" />
              <xs:element name="opacity" type="xs:float" minOccurs="0"
maxOccurs="1" />
              <xs:element name="textColor" type="cmd:colorType"
minOccurs="0" maxOccurs="1" />
              <xs:element name="backgroundColor" type="cmd:colorType"
```

```
minOccurs="0" maxOccurs="1" />
              <xs:element name="drawAsOutline"
type="cmd:emptyElementType" minOccurs="0" maxOccurs="1" />
            </xs:sequence>
            <xs:element name="newline" type="cmd:emptyElementType" />
          </xs:choice>
        </xs:complexType>
    </xs:element>
<!-- if rotation is not specified, it defaults to 0 degrees (horizontal)
-->
<xs:element name="rotation" type="xs:float" minOccurs="0"
maxOccurs="1" />
<!-- if pages is not specified, it defaults to "all pages" -->
<xs:element name="pages" type="cmd:whichPagesType" minOccurs="0"
maxOccurs="1" />
```

### Example of commands for text stamp

The following code provides an example of the commands that can be a part of the XML file. You can **attach this XML file as PDF command XML file to the dataset**.

```
<cmd:textStamp>
<cmd:text>Sensitive and internal only</cmd:text>
<cmd:font>Courier</cmd:font>
<cmd:size>12</cmd:size>
<cmd:position><cmd:coordinates x="-170" y="-72" /></cmd:position>
</cmd:textStamp>
```

The `<cmd:text></cmd:text>`, `<cmd:font></cmd:font>`, `<cmd:size></cmd:size>`, and `<cmd:position></cmd:position>` are the required elements for styling the text stamps.

### Commands available for watermark stamp

The following code and its comments explain the commands that are available for watermark stamp.

```
<xs:element name="text" type="xs:string" minOccurs="1" maxOccurs="1" />
<xs:element name="font" type="xs:string" minOccurs="1" maxOccurs="1" />
<!-- size is measured in points (1/72 inch); if not specified, it
defaults to the largest possible value -->
<xs:element name="size" type="xs:positiveInteger" minOccurs="0"
maxOccurs="1" />
<!-- the Z-order specifies the order in which overlapping items (text,
images, etc.) are stacked; a higher-number
item in the Z-order will be placed on "top" of lower-number item, with 0
being the lowest value -->
<xs:element name="zOrder" type="xs:positiveInteger" minOccurs="0"
maxOccurs="1" />
<!-- if rotation is not specified, it defaults to 0 degrees (horizontal)
```

```
-->
<xs:element name="rotation" type="cmd:watermarkRotationType"
minOccurs="0" maxOccurs="1" />
<!-- if pages is not specified, it defaults to "all pages" -->
<xs:element name="pages" type="cmd:whichPagesType" minOccurs="0"
maxOccurs="1" />
<!-- opacity is a percentage, ranging from 0.0 (fully transparent) to
1.0 (fully opaque); if not specified,
it defaults to 0.5 (50%) -->
<xs:element name="opacity" type="xs:float" minOccurs="0" maxOccurs="1" />
<!-- if textColor is not specified, it defaults to black -->
<xs:element name="textColor" type="cmd:colorType" minOccurs="0"
maxOccurs="1" />
<!-- if backgroundColor is not specified, it defaults to no background
coloration -->
<xs:element name="backgroundColor" type="cmd:colorType" minOccurs="0"
maxOccurs="1" />
```

### Commands available for using logo in the stamp

The following code and its comments explain the commands that are available for inserting logo in the document.

```
<!-- the logo file must be a bitmap, gif, jpeg, png, or a tiff file -->
<!-- this element is generated by the server, and is not present in the
command template created by the admin -->
<xs:element name="logoFilePathKey" type="xs:string" minOccurs="0"
maxOccurs="1" />
<!-- optional text to display if the logo cannot be loaded -->
<xs:element name="replacementText" type="xs:string" minOccurs="0"
maxOccurs="1" />
<!-- if logo size is not specified, it defaults to the logo's normal
size -->
<xs:element name="size" type="cmd:imageSizeType" minOccurs="1"
maxOccurs="1" />
<!-- the position coordinate space is measured in points (1/72 inch);
the origin is in the upper-left corner of the page;
negative values can be used to place the logo relative to the right or
bottom edges -->
<xs:element name="position" type="cmd:positionType" minOccurs="1"
maxOccurs="1" />
<!-- the Z-order specifies the order in which overlapping items (text,
images, etc.) are stacked; a higher-number item
in the Z-order will be placed on "top" of lower-number item, with 0
being the lowest value -->
<xs:element name="zOrder" type="xs:positiveInteger" minOccurs="0"
maxOccurs="1" />
<!-- if rotation is not specified, it defaults to 0 degrees (horizontal)
-->
```

```
<xs:element name="rotation" type="xs:float" minOccurs="0"
maxOccurs="1" />
<!-- if pages is not specified, it defaults to "all pages" -->
<xs:element name="pages" type="cmd:whichPagesType" minOccurs="0"
maxOccurs="1" />
```

### Commands available for using image as a stamp

The following code and its comments explain the commands that are available for inserting image as a stamp in the document.

```
<!-- the image file must be a bitmap, gif, jpeg, png, or a tiff file -->
<!-- this element is generated by the server, and is not present in the
command template created by the admin -->
<xs:element name="imageFilePathKey" type="xs:string" minOccurs="0"
maxOccurs="1" />
<!-- optional text to display if the image cannot be loaded -->
<xs:element name="replacementText" type="xs:string" minOccurs="0"
maxOccurs="1" />
<!-- if image size is not specified, it defaults to the image's normal
size -->
<xs:element name="size" type="cmd:imageSizeType" minOccurs="1"
maxOccurs="1" />
<!-- the position coordinate space is measured in points (1/72 inch);
the origin is in the upper-left corner of the page;
 negative values can be used to place the image relative to the right or
bottom edges -->
<xs:element name="position" type="cmd:positionType" minOccurs="1"
maxOccurs="1" />
<!-- the Z-order specifies the order in which overlapping items (text,
images, etc.) are stacked; a higher-number item
 in the Z-order will be placed on "top" of lower-number item, with 0
being the lowest value -->
<xs:element name="zOrder" type="xs:positiveInteger" minOccurs="0"
maxOccurs="1" />
<!-- if rotation is not specified, it defaults to 0 degrees (horizontal)
-->
<xs:element name="rotation" type="xs:float" minOccurs="0"
maxOccurs="1" />
<!-- if pages is not specified, it defaults to "all pages" -->
<xs:element name="pages" type="cmd:whichPagesType" minOccurs="0"
maxOccurs="1" />
```

### Commands available for specifying the pages in the document on which the stamps must be printed

The following code and its comments explain the commands that you can use for specifying the pages in the document on which the stamp must be printed.

```
<!-- print on all pages -->
<xs:element name="allPages" type="cmd:emptyElementType" />
<!-- print on even-numbered pages -->
<xs:element name="evenPages" type="cmd:emptyElementType" />
<!-- print on odd-numbered pages -->
<xs:element name="oddPages" type="cmd:emptyElementType" />
<!-- print on specified ranges of pages -->
<xs:element name="pageRange" type="cmd:pageRangeType" minOccurs="1"
maxOccurs="unbounded" />
```

### Commands available for specifying the page range in the document for printing stamps

The following code and its comment explain the page range in the document for printing stamps.

```
<!-- startPage and endPage are inclusive -->
<xs:attribute name="startPage" type="xs:positiveInteger" />
<xs:attribute name="endPage" type="xs:positiveInteger" />
```

### Commands available for specifying the position of the stamp within the document

The following code and its comments explain the position of the stamp within the document.

```
<!-- position by x, y using points (1/72 inch) -->
<xs:element name="coordinates" type="cmd:coordinatesType" />
<!-- position in the header section of the page -->
<xs:element name="headerCenter" type="cmd:emptyElementType" />
<xs:element name="headerLeft" type="cmd:emptyElementType" />
<xs:element name="headerRight" type="cmd:emptyElementType" />
<!-- position in the footer section of the page -->
<xs:element name="footerCenter" type="cmd:emptyElementType" />
<xs:element name="footerLeft" type="cmd:emptyElementType" />
<xs:element name="footerRight" type="cmd:emptyElementType" />
```

### Commands available for specifying the text color of the stamp

The following code explains the RGB color that you can use to define the text color.

```
<!-- specify color by Red Green Blue values (0-255) -->
<xs:attribute name="red" type="xs:positiveInteger" use="required" />
<xs:attribute name="green" type="xs:positiveInteger" use="required" />
<xs:attribute name="blue" type="xs:positiveInteger" use="required" />
```

### Commands available for specifying the image size

The following code explains the commands that you can use to specify the size of the image when you use as a stamp.

```
<!-- scale the image by the specified percentages -->
<!-- values should be > 0 and <= 1.0 where e.g. 0.5 = 50% -->
<!-- if the element has values for the horizontal and vertical
attributes, those values will be used -->
<!-- otherwise the inner float will be used for both the horizontal and
vertical dimensions -->
<xs:element name="scaling">
  <xs:complexType mixed="true">
    <xs:attribute name="horizontal" type="xs:float" />
    <xs:attribute name="vertical" type="xs:float" />
  </xs:complexType>
</xs:element>
<!-- specify width and height in points (1/72nd of an inch) -->
<xs:element name="dimensions">
  <xs:complexType>
    <xs:attribute name="width" type="xs:positiveInteger" />
    <xs:attribute name="height" type="xs:positiveInteger" />
  </xs:complexType>
</xs:element>
```

### Commands available for rotating the watermark

The following code explains commands that you can use for rotating the watermark.

```
 <!-- rotate the watermark by an arbitrary amount -->
      <xs:element name="degrees" type="xs:float" />
      <!-- rotate the watermark so that it is on the diagonal of the
page, from the upper-left corner to the lower-right corner -->
      <xs:element name="upperLeftToLowerRight"
type="cmd:emptyElementType" />
      <!-- rotate the watermark so that it is on the diagonal of the
page, from the lower-left corner to the upper-right corner -->
      <xs:element name="lowerLeftToUpperRight"
type="cmd:emptyElementType" />
```

## Set up a workflow to include stamp information in PDF files

1. Create a workflow process template.

   a. In Workflow Designer, click **File→New Root Template**.

   b. In the **New Root Template** dialog box, enter a name in **New Root Template Name**.

   c. Select **Empty Template** in **Based On Root Template** and **Process** in **Template Type**.

   d. Click **OK** to create a workflow process template.

2. Add a task to the workflow process template.

   a. On the toolbar, click **Edit Mode** and then click **Task**.

   b. In the process flow pane, double-click where you want to place the new task.

      A new task appears with the default name **New Task #**. In the **Name** box, type a name for the task.

3. Link the task to its predecessor and successor tasks.

   a. Click the task node you want to specify as the predecessor task.

      > Caution:
      >
      > Do not click the title bar of the task node as this action drags the task node to a different location.

   b. Drag your cursor to the task node you want to specify as the successor task.

      A link arrow follows the cursor as you drag. When your cursor moves over a task node, the node is highlighted.

   c. Release the mouse button.

      A link arrow connects the predecessor and successor nodes.

4. Configure the task.

   a. Right-click the task and click **Task Properties**.

   b. Click the **Task Handler** pane.

   c. In **Task Action**, select **Complete**.

   d. In **Action Handler**, select **DOCMGTAPP-apply-pdf-control** to create a workflow task to include stamps in the existing PDF files.

   e. Click **Create**.

### DOCMGTAPP-apply-pdf-control

| | |
|---|---|
| **Description** | Applies a system stamp, watermark, logo, distribution statement text, workflow signoff table (if the target object is in a review task), and Teamcenter attributes when the logical object is related to the attached PDF dataset. A target object can be an item, an item revision or its subtype, or the PDF dataset itself. |

The system stamp is an imprint comprising data such as a watermark and optional boilerplate text. In Business Modeler IDE, the data model administrator creates a system stamp configuration, associating the configuration with the XML command file that defines the watermark and text. The system stamp configuration also contains *Document Configuration*, *Logo*, and *Distribution Statement* datasets.

For this handler to apply the stamp and watermark, the following conditions are required:

- The PDF dataset must be related to the item revision or its subtype.
- The system stamp configuration must be enabled for the item revision or its subtype. The **Applies To** attribute of the system stamp configuration must be set to **PDF_Control**.
- The **PDF Control** access privilege must be granted.
- Logos and distribution statements must be enabled based on their document configuration setting.

**Syntax** **DOCMGTAPP-apply-pdf-control -user_stamp=***text string*

**Arguments** **-user_stamp**

(Optional) Specifies any string for the text portion of the stamp.

**-ignore_error**

(Optional) Specifies that the current task can be continued on to the next task even if the current task fails. If this optional argument is not specified, the current task might stop on failures (including password protected file or dataset is checked out, or object cannot be saved), and might not continue to the next task.

**Placement** Place on the **Start** action or the **Complete** action.

**Restrictions** None

## Set up system stamps

1. In the Business Modeler Integrated Development Environment (BMIDE), click **BMIDE** > **New Model Element**.
2. In the **Add New Model Element** dialog box, select **System Stamp Configuration** in **Wizards** and click **Next**.
3. Enter the following information for the system stamp configuration:

| Field | Action |
|---|---|
| **Business Object** | Select **ItemRevision**. |
| **Condition** | Select the condition under which this stamp is applicable. |
| **Applies To** | Choose one of the following:<br>• **Print**: To include the system stamps in a printed file.<br>• **Render**: To include the system stamps in a rendered file.<br>• **PDF_Control**: To include the system stamps in an existing PDF file when it is sent to a stamp workflow.<br>• **PrintAndRender**: To include the system stamps in both printed and rendered files. |
| **Include User Name?** | Select this check box to include the print requester's name in the printed files. |
| **Include Date And Time?** | Select this check box to include the print date and time in the printed files. |

4. Click **Next** and enter the following stamp information:

| Field | Action |
|---|---|
| **Properties** | Specify the properties that must be included in the printed or rendered files. You can optionally specify a prefix that must precede a property, for example, **Document Name**: <**object_name**>. |
| **User Stamp** | Type the text that you want to appear on the document, such as *Internal Distribution*. |
| **Watermark** | Type the text that you want to appear as a watermark, such as *Confidential*. |
| **MDS Template** | Select the MetaData Stamp template that defines how the stamps must be applied in the printed and rendered files.<br>To apply stamps in existing PDF files, select the PDF control command XML file.<br>To apply stamps during attribute synchronization, select the Document Management template revision for stamps. |

5. Click **Finish**.

The newly created system stamp configuration is listed under **Extensions**>**Document Management** > **System Stamp Configuration**.

## Improve the output quality of stamped PDF files by using Lifecycle Visualization Convert

While applying stamps to rendered PDF files, update the *Teamcenter Visualization vvcp.ini* file by setting **StampFormat=PDF** in the **Prepare** section. This setting improves the performance and output quality, and preserves the text search capability as the stamped PDF files are not rasterized.

You must set this up on the machine where you have installed Dispatcher and Visualization Convert and Print.

The following metadata stamp (MDS) commands are not supported when the stamp format is set to **PDF** in the *Teamcenter Visualization vvcp.ini* file:

- \BackgroundColor
- \Circle
- \Color
- \FooterCenter
- \FooterLeft
- \FooterRight
- \HalfTone
- \HeaderCenter
- \HeaderLeft
- \HeaderRight
- \Image
- \Include
- \Line
- \Outline
- \Size

# 23. Configuring digital signing for PDF documents

## Setting up digital signing for PDF documents

PDF signatures are often used to designate user approval of a document on a PDF file. When a user digitally signs a PDF, a signature stamp is placed on the PDF file. Digitally signing a document is different from applying digital signatures.

A *digital signature* is a mathematical stamp on a document to indicate if the document was modified after the signature was applied. It also identifies who applied the digital signature. It requires public key infrastructure (PKI) authentication when applying the signature. Users can open PDF files from Active Workspace to add digital signatures. For this, you must install the **Active Workspace Launcher** and **Document Management Teamcenter Client PDF Sign** applications on the client machines.

You can specify the applications that may be launched from Microsoft Outlook for digitally signing a PDF by setting the value of the **Outlook_Sign_Launchable_TcTypes** preference to **PDF**.

## Install Active Workspace Launcher to open Microsoft Office and PDF files from Active Workspace

You must install Active Workspace Launcher on client machines so that the users can open the Microsoft Office and PDF files for performing various tasks. The Active Workspace Launcher can also be installed silently on client machines.

## Install Document Management Teamcenter Client PDF Sign for adding digital signatures

Users can open PDF files from Active Workspace to add digital signatures. For this you must:

- Install **Active Workspace Launcher**.
- Install Security Services Session Agent. After you install it, set the Session Agent environmental variable **TCSSO_SESSION_AGENT_PATH** to the path where you installed Session Agent. For more information, see *Installing the Session Agent* in *Security Services Configuration*.
- Install **Document Management Teamcenter Client PDF Sign** either from **Deployment Center** or by **using a ZIP file**.

### Install Document Management Teamcenter Client PDF Sign from Deployment Center

1. Log on to Deployment Center.

2. Click the **SOFTWARE REPOSITORIES** tile and verify the availability of the software media added for the Teamcenter Foundation and Active Workspace software kits.

3. On the Deployment Center home page, click the **ENVIRONMENTS** tile to view the available environments in Deployment Center. Select the environment on which you want to install Document Management Teamcenter Client PDF Sign.

   - In case you are creating a new environment, click the **Add** button on the **Environments** page.

   - You can also add an existing environment by using the **send_configuration_to_dc** utility.

4. Click the **Deploy Software** tab.

   a. In the **Software** task, select **Foundation** from the **Available Software** list.

   b. Click **Update Selected Software** to add **Foundation** to the **Selected Software** list.

   c. After the list is updated, click **Options**.

5. In the **Options** tab, select the **Distributed** environment type and based on your environment, select the appropriate architecture type. Click **Save Environment Options**.

6. In the **Applications** tab, choose **Edit Selected Applications** ⊕. From the **Available Applications** list:

   - Select **Teamcenter** > **Foundation** > **Content and Document Management** > **Document Management Client Application**.

   Click **Update Selected Applications** and verify that the application you selected to install is listed in the **Selected Applications** list.

7. Click **Go to Components** to view the **Components** tab.

   a. Verify that **Document Management Teamcenter Client PDF Sign** is listed as one of the component.

   b. Start the individual components and bring each component to a 100% complete status by supplying the required settings for each. While supplying settings, record values such as those of installation paths for later reference.

   c. Click **Save Component Settings**.

8. Once the configuration displays the status as 100% complete in the **Selected Components** list, click **Go to Deploy** to generate the script.

9. In the **Deploy** task, click **Generate Install Scripts**.

10. Copy the scripts to the target machine and run the scripts. Follow the instructions specified in *Run the deployment scripts*, available in the Deployment Center documentation.

11. To verify if **Document Management Teamcenter Client PDF Sign** (**TcClientPDFSign**) installed successfully, check if the *TcClientPDFSign.jar* file is available at *C:\Program Files (x86)\Siemens\Teamcenter\WSLauncheri*.

### Install Document Management Teamcenter Client PDF Sign by using a ZIP file

Before installing **Document Management Teamcenter Client PDF Sign**, ensure that Teamcenter client communication system (TCCS) is installed. Also verify that the **FMS_HOME** environmental variable is set to the path where TCCS is installed.

To install **Document Management Teamcenter Client PDF Sign** manually:

1. In the Teamcenter release software location, navigate to *Tcrelease_wntx64\wntx64\additional_applications*.

2. Extract the contents of *tcclientpdfsign.zip* to the location where you installed Active Workspace Launcher, for example, *C:\Program Files (x86)\Siemens\Teamcenter\WSLauncheri*.

   Verify that the *TcClientPDFSign.jar* file and the *lib* folder are available at this location.

## Troubleshoot log files if Document Management Teamcenter Client PDF Sign (TcClientPDFSign) does not launch

If the **Document Management Teamcenter Client PDF Sign** (**TcClientPDFSign**) application does not launch when a user is trying to apply digital signature to a PDF file, you can access the log file to troubleshoot this issue.

To access the log file:

- Navigate to *C:\Program Files (x86)\Siemens\Teamcenter\WSLauncheri* and open the *TcClientAppLauncher.exe.config* file.

  The value of the **LogFileAppender** appender contains the path of the log file.

## Install Teamcenter Client Acrobat Launcher for opening a PDF document in Adobe Acrobat or Reader DC

After installing Teamcenter Client Acrobat Launcher, users can open a PDF document from Active Workspace in Adobe Acrobat or Reader DC and then perform the following tasks:

- Add markups

- Digitally sign the PDF file

However, for this you must:

- Install **Active Workspace Launcher**.
- Install Security Services Session Agent. After you install it, set the Session Agent environmental variable **TCSSO_SESSION_AGENT_PATH** to the path where you installed Session Agent. For more information, see *Installing the Session Agent* in *Security Services Configuration*.
- Install Adobe Acrobat or Reader DC.

  You can either use 32-bit or 64-bit of Adobe Acrobat or Reader DC considering your business requirement.

### Install Teamcenter Client Acrobat Launcher from Deployment Center

1. Log on to Deployment Center.
2. Click the **SOFTWARE REPOSITORIES** tile and verify the availability of the software media added to the Teamcenter Foundation and Active Workspace software kits.
3. On the Deployment Center home page, click the **ENVIRONMENTS** tile to view the available environments in Deployment Center. Select the environment on which you want to install Teamcenter Client Acrobat Launcher.

   - If you are creating a new environment, click the **Add** button on the **Environments** page.
   - Add an existing environment by using the **send_configuration_to_dc** utility.

4. Click the **Deploy Software** tab.

   a. In the **Software** task, select **Foundation** from the **Available Software** list.

   b. Click **Update Selected Software** to add **Foundation** to the **Selected Software** list.

   c. After the list is updated, click **Options**.

5. In the **Options** tab, select the **Distributed** environment type and based on your environment, select the appropriate architecture type. Click **Save Environment Options**.
6. In the **Applications** tab, choose **Edit Selected Applications** ⊕. From the **Available Applications** list:

   - Select **Teamcenter** > **Foundation** > **Content and Document Management** > **Teamcenter Client Acrobat Launcher**.

After you select the **Teamcenter Client Acrobat Launcher** application, the applications **Acrobat/ Reader Plugin** and **Teamcenter Client Communication System (TCCS)** are selected automatically as they are dependent applications.

7. Click **Update Selected Applications** and verify that the application you selected to install is listed in the **Selected Applications** list.

8. Click **Go to Components** to view the **Components** tab.

   a. Verify that **Document Management Teamcenter Client PDF Sign** is listed as one of the components.

   > Note:
   >
   > The **Teamcenter Client Acrobat Launcher** application and the **Document Management Teamcenter Client PDF Sign** application share the same component.

   b. Start the individual components and bring each component to a 100% complete status by supplying the required settings for each. While supplying settings, record values such as those of installation paths for later reference.

   c. Click **Save Component Settings**.

9. Once the configuration displays the status as 100% complete in the **Selected Components** list, click **Go to Deploy** to generate the script.

10. In the **Deploy** task, click **Generate Install Scripts**.

11. Copy the scripts to the target machine and run the scripts. Follow the instructions specified in *Run the deployment scripts*, available in the Deployment Center documentation.

12. To verify if **Teamcenter Client Acrobat Launcher** installed successfully, check if the *TcAcrobatLauncher.jar* file is available at *C:\Program Files (x86)\Siemens\Teamcenter\WSLauncheri*.

### Install Teamcenter Client Acrobat Launcher by using a ZIP file

Before installing **Teamcenter Client Acrobat Launcher**, ensure that Teamcenter client communication system (TCCS) is installed. Additionally, verify that the **FMS_HOME** environmental variable is set to the path where TCCS is installed.

To install **Teamcenter Client Acrobat Launcher** manually:

1. In the Teamcenter release software location, navigate to *Tc release_wntx64\wntx64\additional_applications*.

2. Extract the contents of *tcarobatlauncher.zip* to the location where you installed Active Workspace Launcher, for example, *C:\Program Files (x86)\Siemens\Teamcenter\WSLauncheri*.

   Verify that the *TcAcrobatLauncher.jar* file and the *lib* folder are available at this location.

### Install Teamcenter Acrobat/Reader plugin by using a ZIP file

Before installing **Teamcenter Acrobat/Reader plugin**, ensure that Adobe Acrobat/Reader DC application is installed. The Adobe Acrobat/Reader DC application can be either 32-bit or 64-bit application.

To install Teamcenter Acrobat/Reader plugin manually:

1. In the Teamcenter release software location, navigate to *Tc release_wntx64\wntx64\additional_applications*.

2. Perform one of the following actions:

   - For Acrobat/Reader (32-bit) application, extract the contents of *vmeplgin.zip* to a temporary directory, for example, *C:\temp\vmeplgin directory*.

   - For Acrobat/Reader (64-bit) application, extract the contents of *vmeplgin_x64.zip* to a temporary directory, for example, *C:\temp\vmeplgin_x64 directory*.

3. In the Teamcenter release software location, navigate to *Tcrelease_wntx64\wntx64\additional_applications*.

4. Perform one of the following actions:

   - For Acrobat/Reader (32-bit) application, perform the following steps:

     a. Extract the contents of the *vmeplgin.zip* file to the same temporary directory from the step 2 above.

        This action might overwrite some of the files from the release.

     b. Copy both the *plug_ins* and *Javascripts* directories to the location where you installed Adobe Acrobat/Reader DC (32-bit) application, for example, *C:\Program Files (x86)\Adobe\Acrobat Reader DC\Reader*.

     c. Verify that the *vmeplgin* directory is under *C:\Program Files (x86)\Adobe\Acrobat Reader DC\Reader\plugins directory*.

   - For Acrobat/Reader (64-bit) application, perform the following steps:

     a. Extract the *vmeplgin_x64.zip* file to the same temporary directory from the step 2 above.

This action might overwrite some of the files from the release.

b. Copy both the *plug_ins* and *Javascripts* directories to the location where you installed Adobe Acrobat/Reader DC (64-bit) application, for example, *C:\Program Files\Adobe\Acrobat DC\Acrobat*.

c. Verify that the *vmeplgin* directory is under *C:\Program Files\Adobe\Acrobat DC\Acrobat\plugins directory*.

> Note:
>
> The 32-bit *vmeplgin.zip* must be used for the (32-bit) application and the 64-bit *vmeplgin_x64.zip* must be used for (64-bit) application.

5. **Turn off Adobe protected mode** for the Teamcenter menu to appear in Adobe Acrobat/Reader DC application.

6. Verify that the Teamcenter menu appears in Adobe Acrobat/Reader application.

   To verify the Acrobat/Reader plugin is installed and loaded successfully, start Adobe Acrobat/Reader application and then click **Help→About Third-Party Plug-ins→Teamcenter** menu. The Teamcenter plugin version information dialog box appears.

## Provide access privileges for digitally signing PDF files

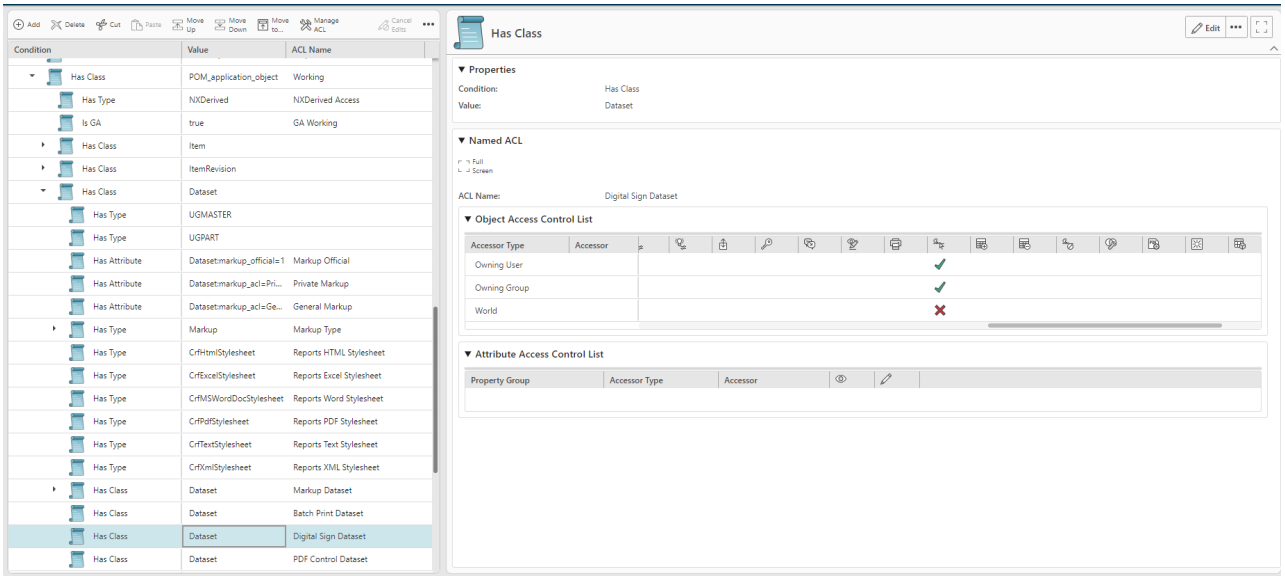
As an administrator, you can use Access Manager in Teamcenter to provide access privileges for digitally signing PDF files. The **Digital Sign Dataset** ACL rule grants the owning user and owning group digital signing privileges for the dataset object by default.

To update the access privileges for digitally signing PDF files:

1. Log on to Active Workspace as a administrator and click Access Manager.

2. In the **Access Manager Rules** tab, expand the following tree:

   **Has Class** condition with **POM_application_object** value and **Working ACL** name > **Has Class** condition with **Dataset** value > **Has Class** condition with **Dataset** value and **Digital Sign Dataset** ACL name




**World** access users do not have digitally signing privileges.

The **Digital Sign Dataset** ACL rule may not be present in an upgraded system. To use the digital signing capability, you must create this rule.

3. Click **Edit** and modify the details as needed.

By default, markups are not included while digitally signing PDF files

The initial configuration for the digital signing of PDF files does not include markups as the default value of the **Fnd0IncludeMarkupsWithSignedFile** business object constant is **DoNotIncludeMarkups**.

# 24. Define how a document revision is handled in Teamcenter

You can define how a document revision is handled during its lifecycle in Teamcenter. For example, you can define that every time a document revision is checked in, a PDF of the dataset attached to the item revision must be generated. You can also choose to include stamps in the rendered PDF. You can also set up *deep copy rules* that define till what level an item revision must be copied when a new item revision is created from an existing one.

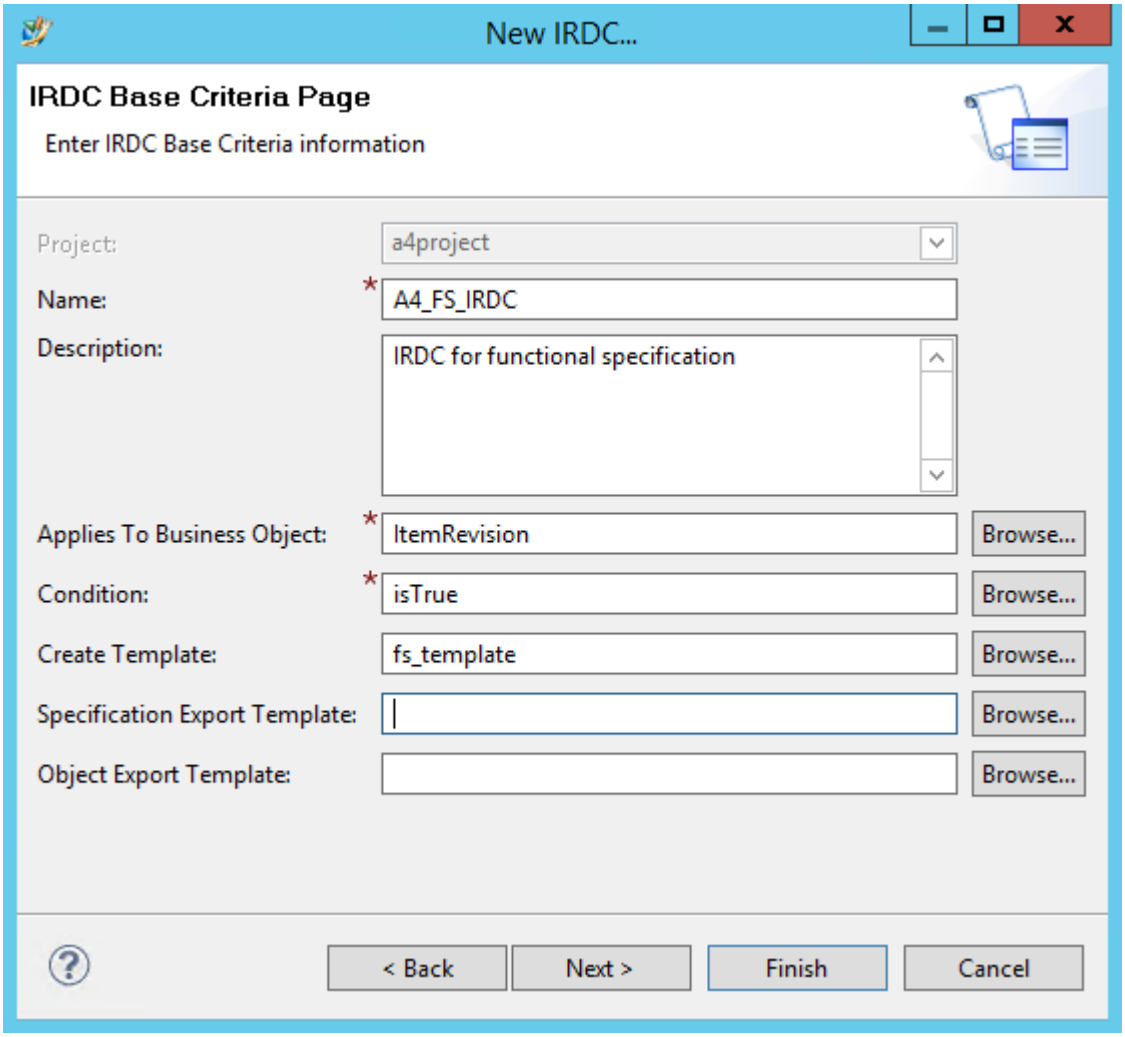
You set up this definition by using *item revision definition configuration (IRDC)*. You must create an IRDC for every document for which you want a standardized behavior. For example, you can create an IRDC for a specification document. Similarly, you can create another IRDC for the regulatory filing document.

In IRDC, you:

1. **Specify the base criteria information and template** for which you want a standardized behavior.
2. **Specify the source and derived file formats** for a document revision.
3. **Specify the naming convention** of the source and rendered file formats.
4. **Set the check in behavior** of a document revision.
5. **Set the deep copy rules**. *Deep copy rules* govern how item revisions are copied during *save as* and *revise* operations.
6. **Specify the markup information**.

## Specify IRDC base criteria information

1. In BMIDE, click **BMIDE→New Model Element**.
2. In the **New Model Element** dialog box, type **IRDC** in **Wizards**.
3. In the **New IRDC** wizard, specify the IRDC base criteria information and click **Next**:

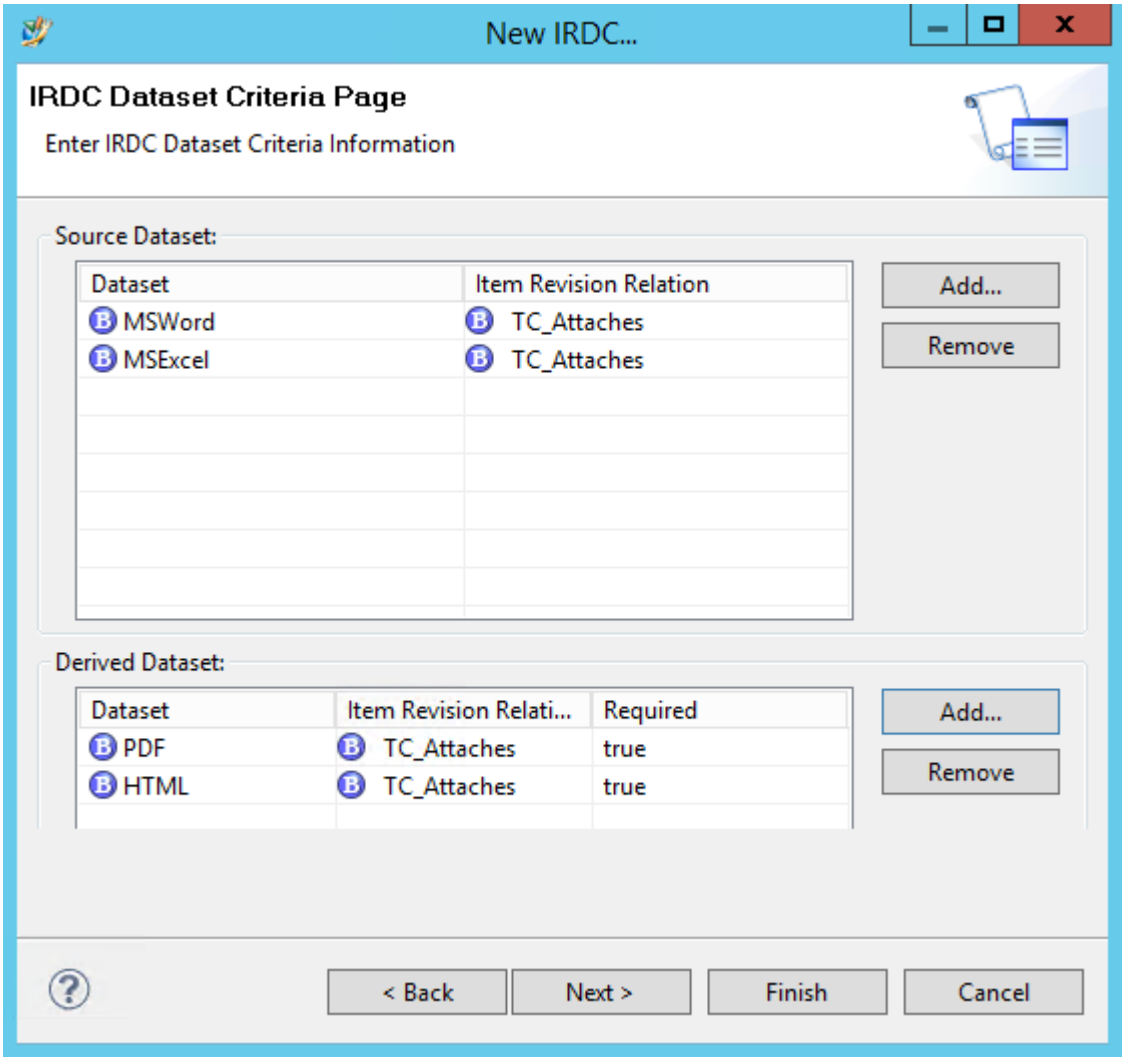



| Field | Action |
|---|---|
| **Applies to Business Object** | Browse for the business object for which you want the standardized behavior, for example, search for **ItemRevision**. |
| **Condition** | Browse for a condition that is applicable to the IRDC. By default, the **isTrue** condition is selected.<br><br>If the condition set for a parent business object (**ItemRevision**) resolves to **true** but the condition set for its child business object (**DocumentRevision**) evaluates to **false**, the parent's condition is used. However, if conditions of both parent and child business objects evaluate to **false**, the **isTrue** condition is used. |
| **Create Template** | Browse for the required Document Management template. |

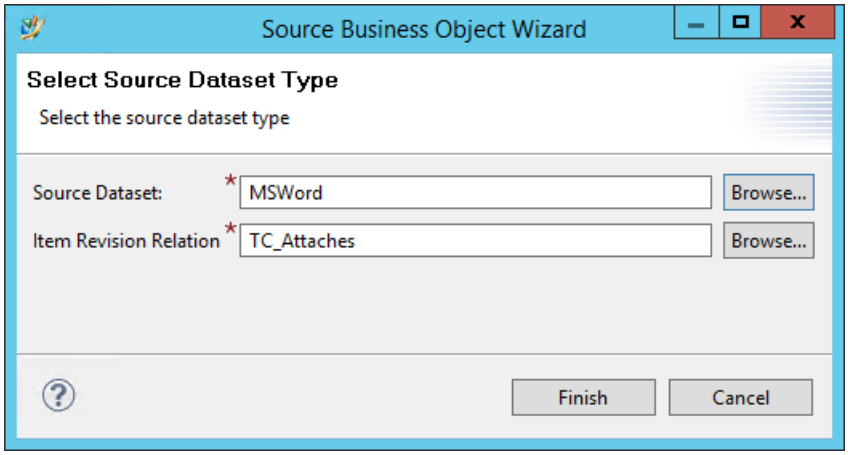
| Field | Action |
| --- | --- |
| | Note:<br>When an IRDC is defined at the parent business object, such as **ItemRevision**, and when an IRDC condition does not evaluate to true for the child business objects, the parent business object's condition is evaluated. If this condition is **true**, the IRDC defined for the parent is used when the business object is created.<br>If you want to assign a template, click the **Browse** button to the right of the **Create Template** box to select the Document Management template to provide initial source datasets for the item revision when it is created. The **Teamcenter Repository Connection** wizard prompts you to log on to a server to look up the available templates. |

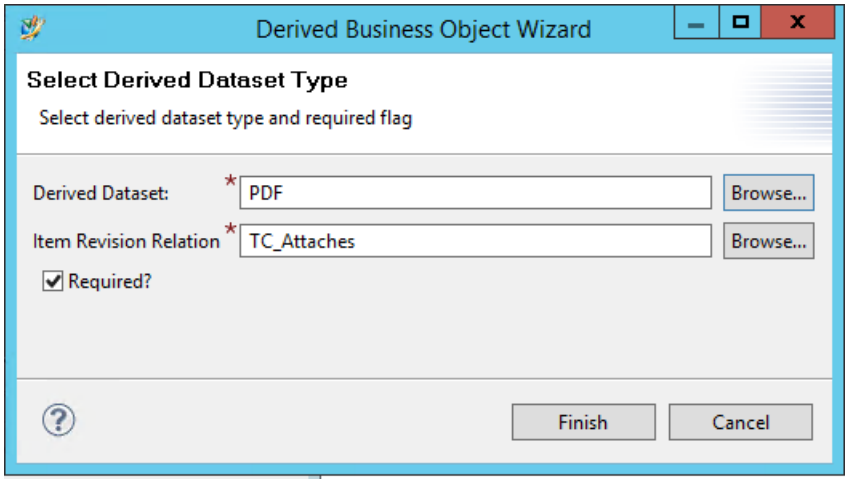
### Specify the source and derived file formats

1. Specify the source and derived dataset and click **Next**:




| Field | Action |
|---|---|
| **Source Dataset** | Click **Add** to locate the source dataset. |



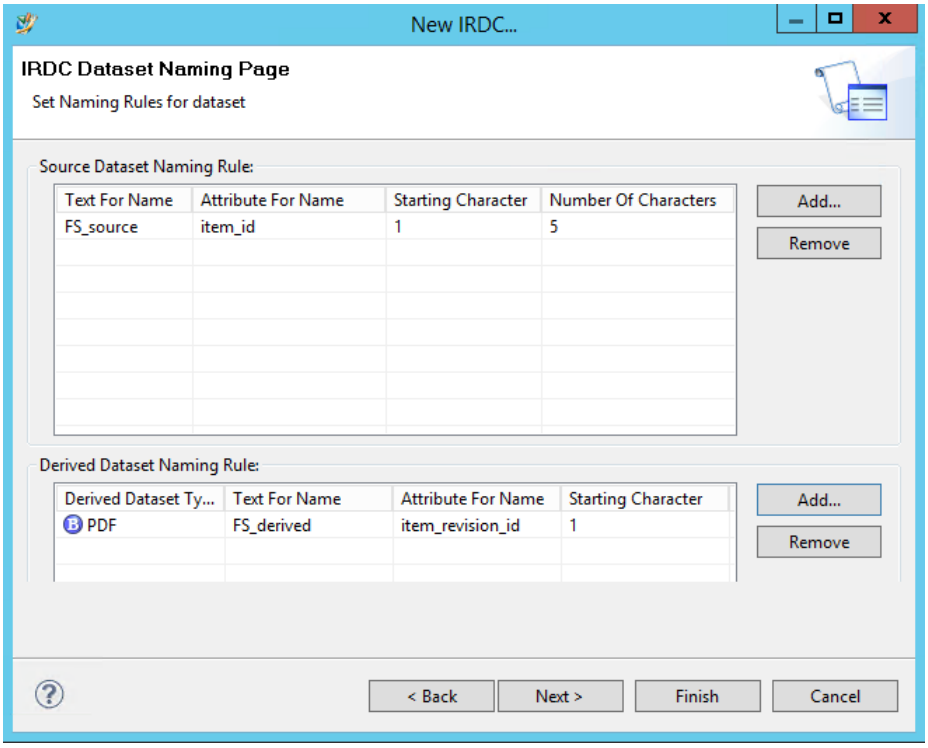
| Field | Action |
| --- | --- |
| | For datasets that contain the Document Management template files, select **TC_Attaches** in **Item Revision Relation**.<br><br>For datasets that contains the page information, such as cover page, signature page, or index page, select the **Fnd0DocPageTypeRel** in **Item Revision Relation**. |
| **Derived Dataset** | Click **Add** to locate the derived dataset.<br><br>  <br><br><br>For datasets that contain the Document Management template files, select **TC_Attaches** in **Item Revision Relation**.<br><br>For datasets that contains the page information, such as cover page, signature page, or index page, select the **Fnd0DocPageTypeRel** in **Item Revision Relation**. |

The order of the source datasets is used to resolve the dataset type for attaching files.

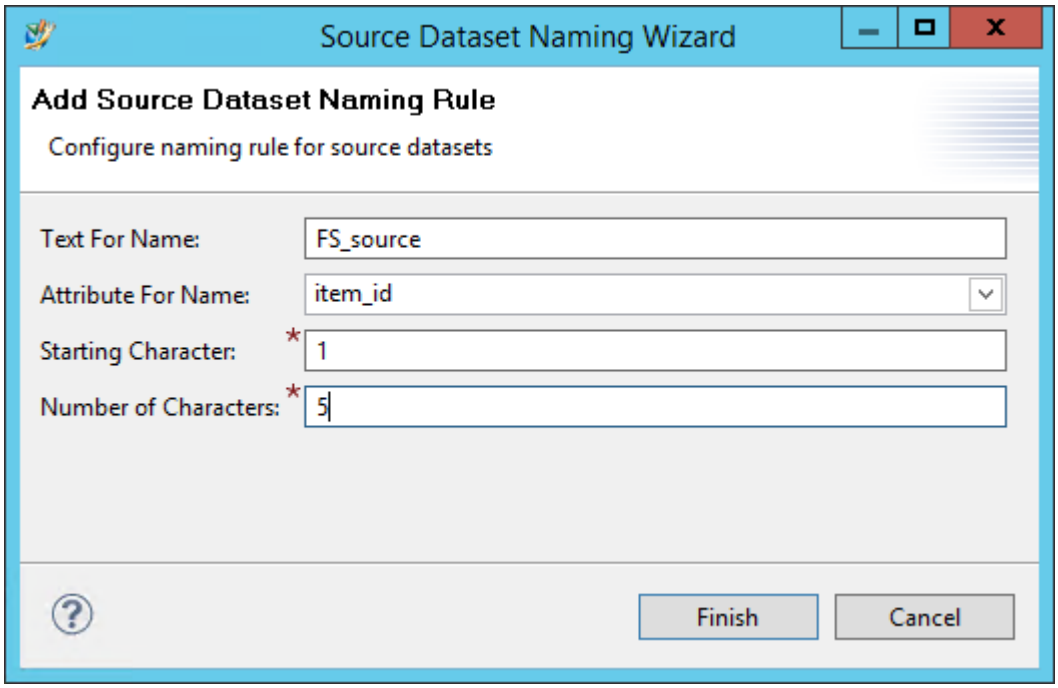
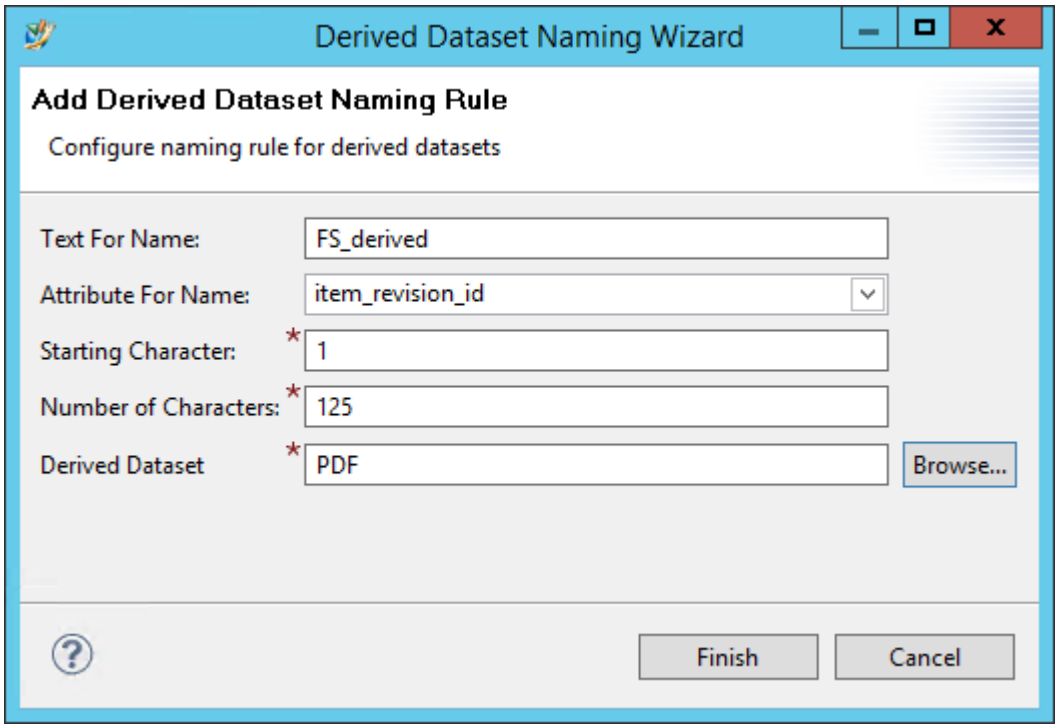
Any dataset type such as **Text** that allows any file extension must be defined last in the order in the source dataset list. If it is the first in the list, all the source datasets defined later are ignored when automatically selecting a type for a file extension.

### Specify the naming convention of the source and rendered file formats

1. Set the naming conventions for the source and derived datasets and click **Next**:

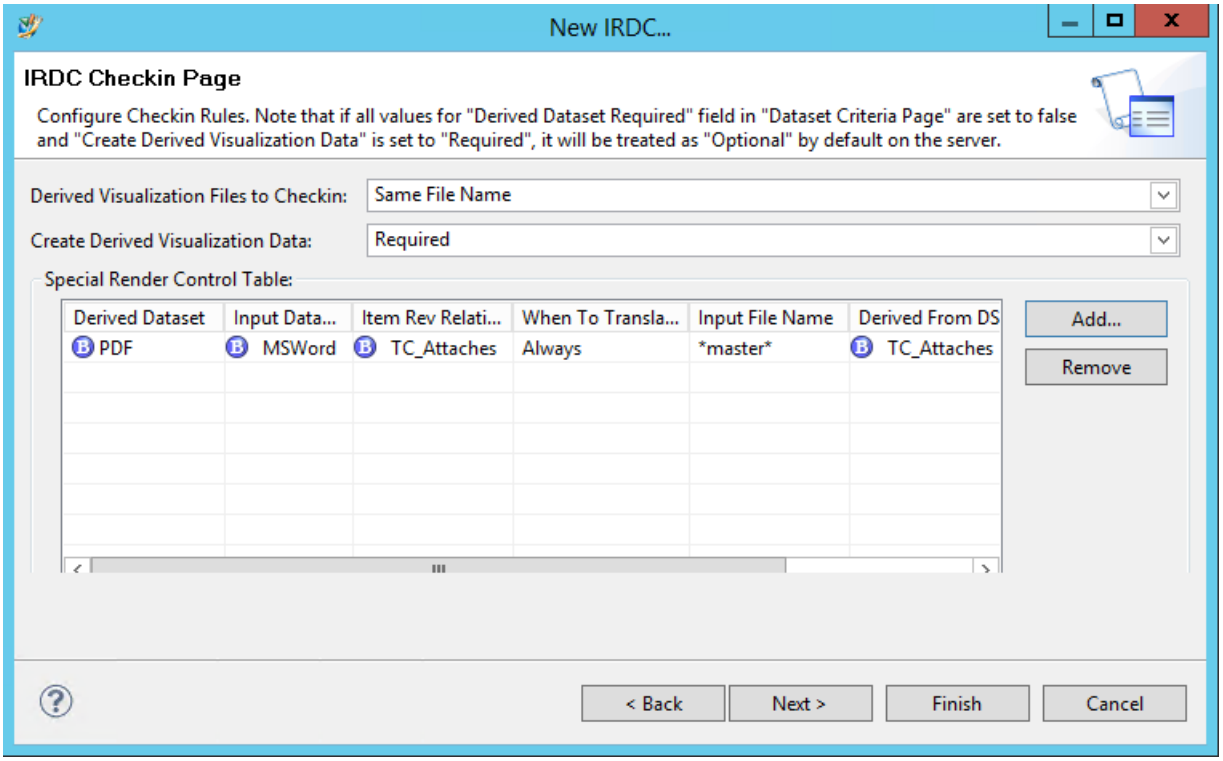



| Field | Action |
|---|---|
| **Source Dataset Naming Rule** | Click **Add** to define a naming convention for the source dataset.<br><br>The name is built using the information you provide in **Source Dataset Naming Wizard**. It concatenates the entries made in this table (text and specified portions of all the selected attributes) to create the new file name and then adds the file extension that is appropriate for the dataset type. For example, the name of the source file will be **FS_source***<item_name><item_id> (start from the first character of the item id till the 5th character).<file extension of the dataset>* |

| Field | Action |
| --- | --- |
| |  |
| **Derived Dataset Naming Rule** | Click **Add** to define a naming convention for the derived dataset.<br>The name is built using the information you provide in **Derived Dataset Naming Wizard**. |
| |  |

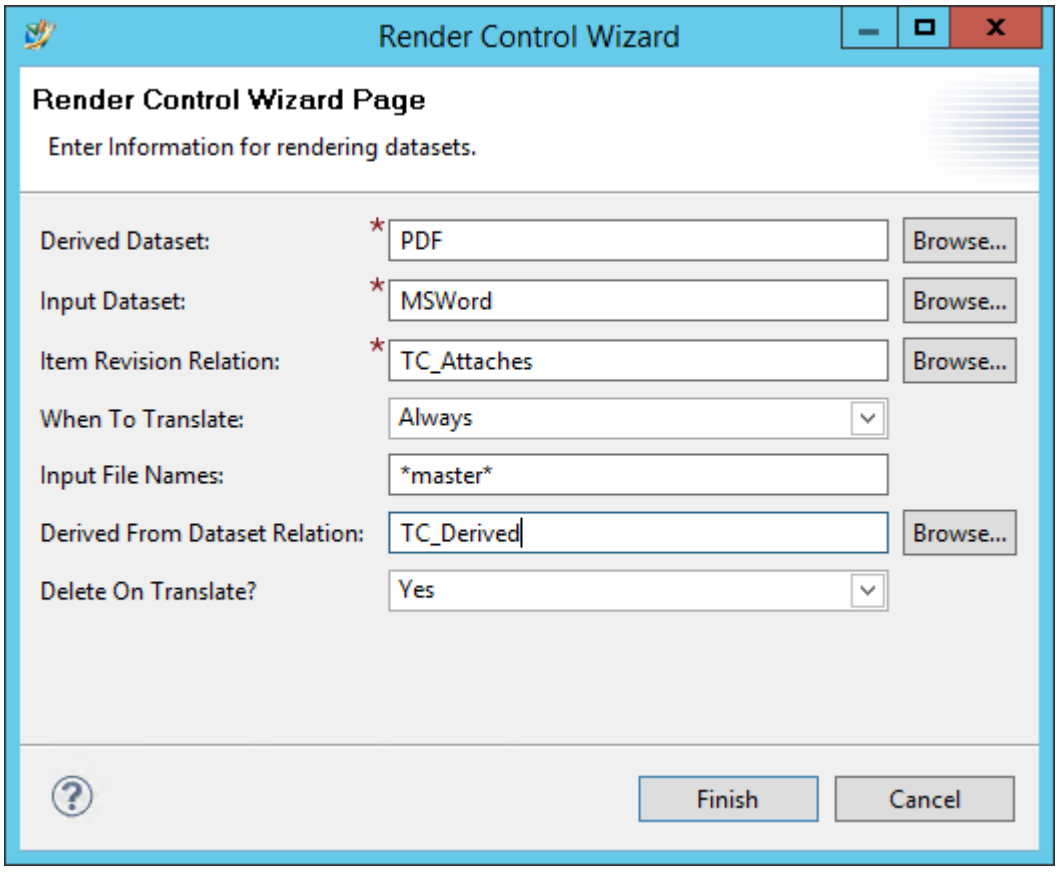
## Set the check-in behavior of a document revision

1. Specify the behavior of the item revision when it is checked in and click **Next**:




| Field | Action |
|---|---|
| **Derived Visualization Files to Checkin** | Choose one of the following options to standardize the checkin behavior of an item revision:<br>• **Same File Name**<br>Attaches and checks in the derived files only if they have the same name as the source dataset.<br>• **Any File Name**<br>Attaches and checks in the derived files irrespective of the names.<br>• **None**<br>Does not attach and check in any derived files. |
| **Create Derived Visualization Data** | Choose one of the following options to create the derived files on checkin: |

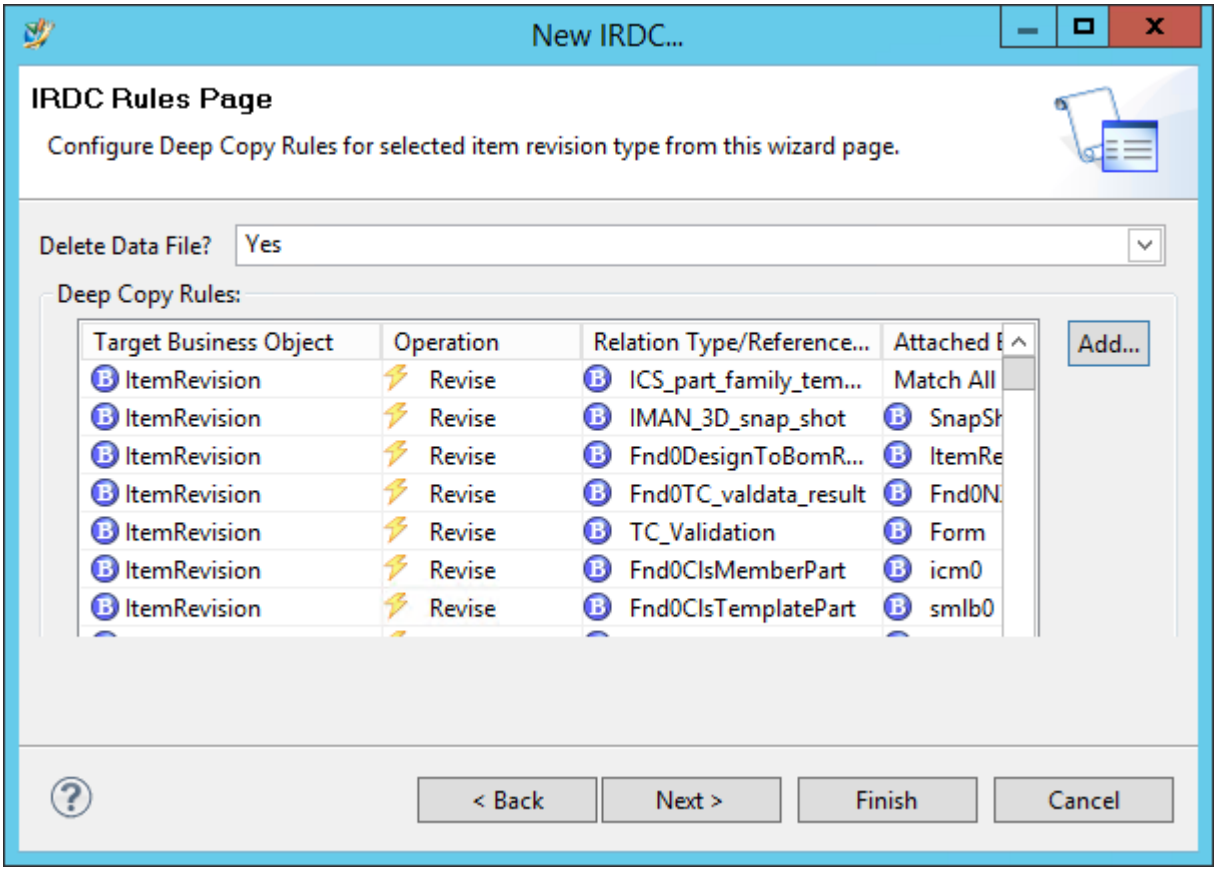
| Field | Action |
| --- | --- |
| | • **Required**<br><br>Creates derived files at checkin. If the required derived file cannot be generated or located, checkin cannot proceed.<br><br>• **Optional**<br><br>Creates derived files if possible, but if not, continues to check in the source dataset.<br><br>• **No**<br><br>Does not create derived files at checkin. |
| **Special Render Control Table** | Click **Add** to define which source files must be rendered to the specific derived formats during checkin: |



| Field | Action |
|---|---|
| | **Field** / **Action** |
| | **Item Revision Relation** — Type the relation between the source dataset and the document revision. |
| | **When To Translate** — • **Always**<br>Specifies that the source dataset must always be translated.<br>• **If First**<br>Specifies that the source dataset must be translated if it is the first one found in the table for this derived dataset type. This is because the rows are processed in order. For example, an item may have an IRDC defined with both **MSWord** and **MSExcel** source files, but a particular item may have only an **MSWord** or only an **MSExcel** file. If the table is defined with **MSWord to PDF** in the first row and **MSExcel to PDF** in the second row, and both rows are set as **If First**, an item with both files types only gets the **MSWord** type translated. Similarly, an item with only an **MSWord** setting gets the **MSWord** type translated, and an item with only the **MSExcel** setting gets the **MSExcel** type translated. |
| | **Input File Names** — Type the file names of the source dataset that must be translated during check in. For example, if you specify ***master***, all files with the word, **master**, in their name will be translated to the specified derived format. |
| | **Derived From Dataset Relation** — Type the relation between the derived dataset and the document revision. |

### Set the deep copy rules

1. Set the deep copy rules for the item revision and click **Next**:




| Field | Action |
|---|---|
| **Delete Data File?** | Specify whether files attached to the item revision should also be deleted when the item revision is deleted. Select **Yes** to delete the files. |
| **Deep Copy Rules** | Click **Add** to create a deep copy rule. |

## Specify the markup information

1. In the **Enter Markup Information** dialog box, configure the markup rules and click **Finish**:

| Field | Action |
|---|---|
| **Markup Application** | Specify the application to be used for marking up documents, for example, **MSWord**. If specified, users can mark up documents only in the specified application. This is to prevent some users marking up |

| Field | Action |
| --- | --- |
| | with Lifecycle Visualization and other users marking up with Adobe tools. |
| **Markup Dataset** | Specify the document type that can be marked up. For example, if you specify **MSWordX**, users can mark up only Microsoft Word documents. |

The new IRDC appears in the **Extensions→Document Management→IRDC** folder. Next, you must save the changes to the data model and deploy the changes to the server:

1. To save the changes to the data model, click **BMIDE→Save Data Model**.

2. To deploy the changes to the server, click **BMIDE→Deploy Template**.

   Restart Pool Manager to get the servers use the new configuration. You must also restart the Dispatcher client if it is running.

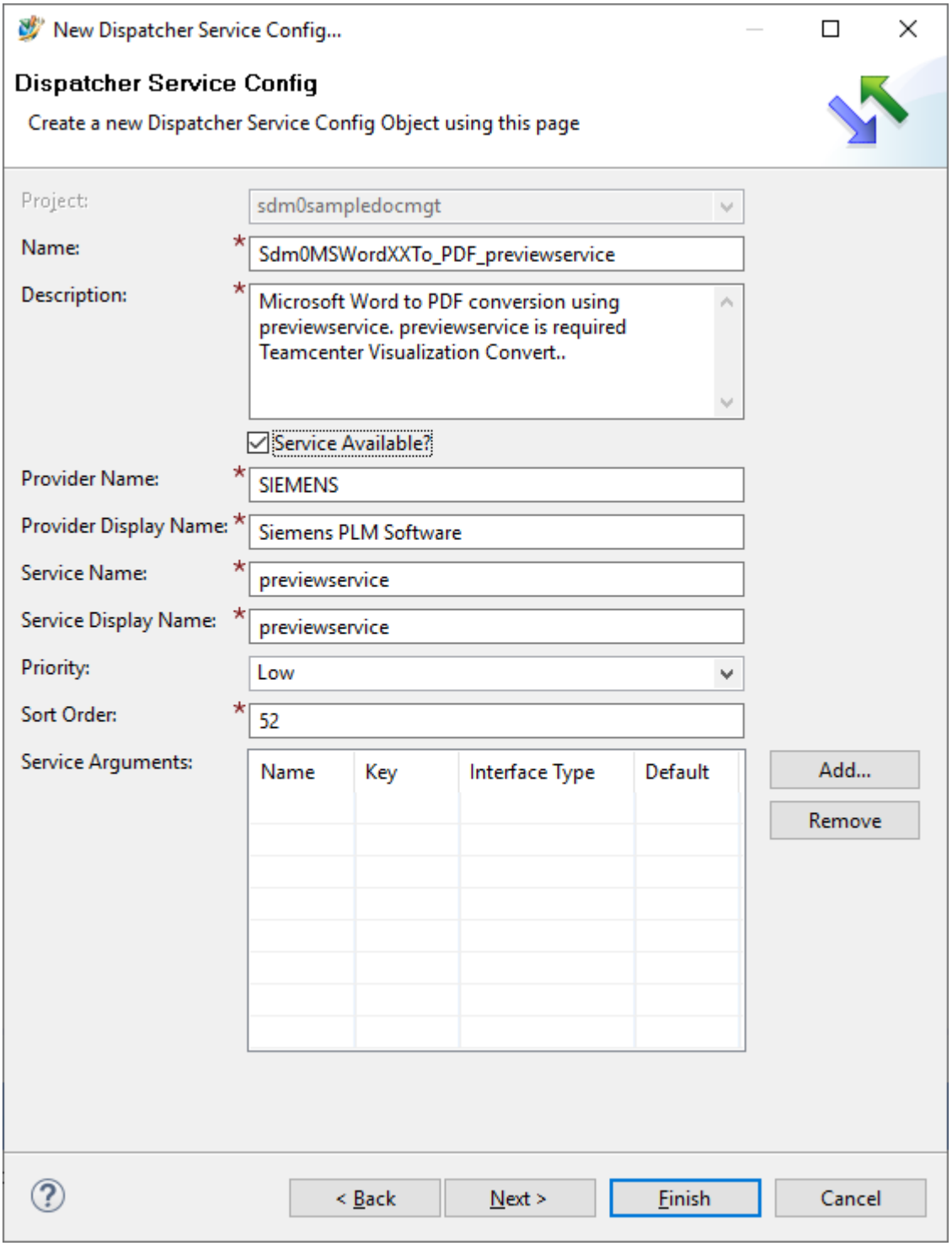
# 25. Set up file translations by creating a dispatcher service configuration

Documents can be rendered to different file formats. For example, a Microsoft Word file (source dataset) can be rendered to a PDF file (presentation or derived dataset). To translate a document from one format to another, you create a *dispatcher service configuration* that maps the source and derived formats. To do so:

1. In BMIDE, click **BMIDE→New Model Element**.

2. In the **New Model Element** dialog box, type **Dispatcher Service Config** folder in **Wizards** and click **Next**.

3. In the **New Dispatcher Service Config** dialog box, enter the required information and click **Finish**:



| Field | Action |
| --- | --- |
| **Service Available** | Select this check box to make the service available for use. |
| **Provider Name** | Type the name of your organization as the provider of the service, for example, **SIEMENS**. |
| **Provider Display Name** | Type the name of your organization the way you want it to be displayed on the user interface. |
| **Service Name** | Type the name of the translator that will perform the translation, for example, **previewservice**. |
| **Service Display Name** | Type a name for the translation service the way you want it to be displayed on the user interface, for example, **previewservice**. |
| **Priority** | Select the importance of this service configuration when a multiple configurations are queued to be executed by Teamcenter Dispatcher. Configurations with a high priority are executed first, while those with a low priority are executed last. |
| **Sort Order** | Enter a number to assign a precedence for this Dispatcher service when multiple source dataset types are available to create a given output. The Dispatcher **RenderMgtTranslator** translator module uses the highest sort order number when multiple service configurations are available. The lowest number entered in this box is considered to have the lowest sort order.<br><br>The following table illustrates the effect of the sort order setting when Microsoft Office dataset types are rendered. Assume that on the **IRDC Dataset Criteria Page** dialog box, the **Source Dataset** table lists **MSWord**, **MSExcel**, and **MSPowerPoint**, and the **Derived Dataset** table lists **PDF**. |

| Source dataset | Sort order | Derived dataset | Comments |
| --- | --- | --- | --- |
| MSWordA.doc<br>MSWordB.doc<br>MSWordC.doc | MSWord = 3<br>MSExcel = 2<br>MSPowerPoint = 1 | MSWordA.doc.pdf<br>MSWordB.doc001.pdf<br>MSWordC.doc002.pdf | Translate the **MSWord** dataset type because **MSWord** is the only attached source dataset type. |
| MSExcelA.xls<br>MSExcelB.xls<br>MSExcelC.xls | MSWord = 3<br>MSExcel = 2<br>MSPowerPoint = 1 | MSExcelA.xls.pdf<br>MSExcelB.xls001.pdf<br>MSExcelC.xls002.pdf | Translate the **MSExcel** dataset type because **MSExcel** is the only attached |

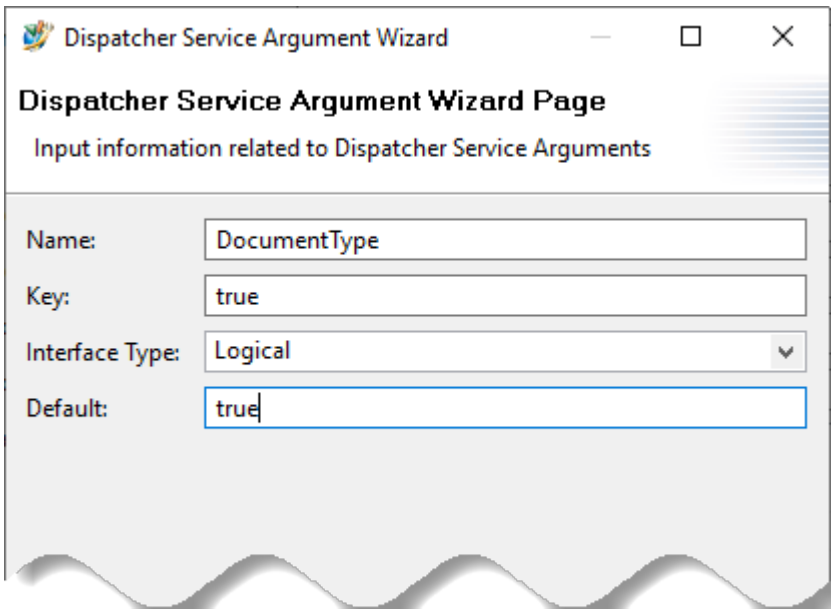
| Field | Action |
|---|---|

| Source dataset | Sort order | Derived dataset | Comments |
|---|---|---|---|
| | | | source dataset type. |
| MSWordA.doc<br>MSExcelA.xls<br>MSPower-PointA.ppt | MSWord = 3<br>MSExcel = 2<br>PowerPoint = 1 | MSWordA.doc.pdf | Translate the **MSWord** dataset type because it has the highest sort order. |
| MSWordA.doc<br>MSExcelA.xls<br>MSPower-PointA.ppt | MSWord = 2<br>MSExcel = 3<br>MSPowerPoint = 1 | MSExcelA.pdf | Translate the **MSExcel** dataset type because it has the highest sort order. |

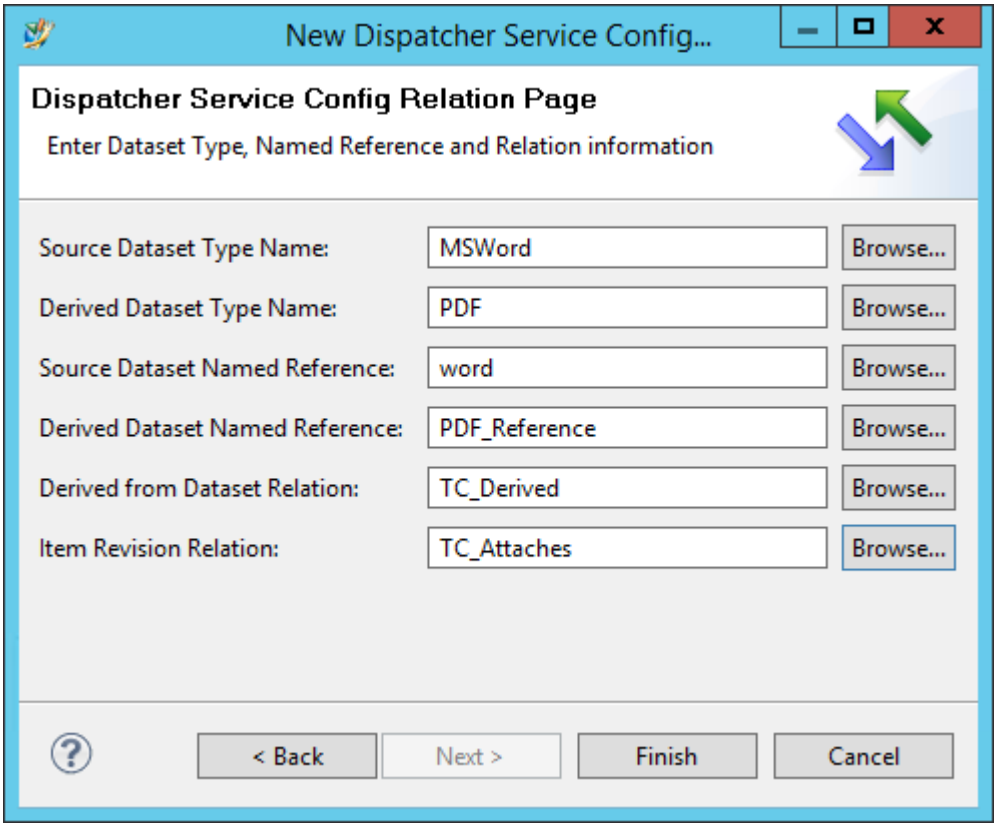
For a document revision, only one dataset type is rendered for a specified derived dataset type.

**Service Arguments**

Click **Add** to add arguments that you want to use in the file translation.




4. Enter the dataset type, named reference, and relation information and click **Finish**:




| Field | Action |
|---|---|
| **Source Dataset Named Reference** | Select the type of named reference that the source dataset uses.<br>Datasets are often used to manage several different types of files. These files are the named references of the dataset. Each dataset type uses a predefined set of named references. For example, if your source dataset is an **MSWord** dataset type, select **Word** for the named reference. |
| **Derived Dataset Named Reference** | Select the type of named reference that the derived dataset uses.<br>For example, if the translated file type is PDF, select **PDF_Reference**. |
| **Derived From Dataset Relation** | Select **TC_Derived** to show the relationship that the derived dataset has with the source dataset. The source dataset is the primary object and the derived dataset is the secondary object. This field is optional. If it is not defined, each time the source dataset is translated, a new derived dataset is created. If this relation is defined, the named reference file of the existing dataset is updated. |
| **Item Revision Relation** | Select the relationship that the derived dataset has with the document revision. The document revision is the primary object and the derived dataset is the secondary object. This field is optional, and |

| Field | Action |
| --- | --- |
| | **TC_Attaches** is the default relation. The IRDC **Derived Dataset** table can specify the item revision relation that overrides this. |

The new dispatcher service configuration appears in the **Extensions→Document Management→Dispatcher Service Config** folder. Next, you must save the changes to the data model and deploy the changes to the server:

1. To save the changes to the data model, click **BMIDE→Save Data Model**.

2. To deploy the changes to the server, click **BMIDE→Deploy Template**.

# 26. Enable thumbnails for documents

Authors attach thumbnails (images) to a document revision to represent the type of documents they are saving in the Teamcenter database. However, these images are displayed only when you set the value of the **TC_display_thumbnail_in_UI** preference to **true**.

If multiple documents are attached to a document revision, you can prioritize the documents to determine which document's thumbnail must be displayed by setting the **ItemRevision_thumbnail_relations** and **ItemRevision_thumbnail_references** preferences.

# 27. Export files along with their attributes to another site

You can export files attached to a document revision from one site to another site. Along with the files, their attributes also get exported. Before exporting the files,you must first **set up logical objects**. Document authors later attach files as datasets to a document revision by using **Logical Object Type Relation**. In this relation, they select the logical object that you set.

### Exporting files in a multisite environment

The export of the **Fnd0LogicalObjectTypeRel** relation is not supported by the **objio** export mechanism.

1. Import the logical object that you created at the source and the target sites.

   **admin_data_import -adminDataTypes=LogicalObjects -u=***username* **-p=***password* **-g=***group_name* **-inputPackage=***logical_object_path*

   For example:

   **admin_data_import -adminDataTypes=LogicalObjects -u=***Tc-admin-user* **-p=***password* **-g=***group* **-inputPackage=***C:\Development\Export\LO\LogicalObjectsExport106.zip*

2. Export the document revision from the source site.

   **data_share -u=***username* **-p=***password* **-f=offline_export -item_id=***document_rev_item_id* **-optionset -dir=***folder_name*

3. Import the document revision at the target site.

   **data_share -u=***username* **-p=***password* **-f=offline_export -optionset -dir=***folder_name*

### Exporting files in a site consolidated environment

1. Import the logical object that you created at the source and the target sites.

   **admin_data_import -adminDataTypes=LogicalObjects -u=***username* **-p=***password* **-g=***group_name* **-inputPackage=***logical_object_path*

   For example:

   **admin_data_import -adminDataTypes=LogicalObjects -u=***Tc-admin-user* **-p=***password* **-g=***group* **-inputPackage=***C:\Development\Export\LO\LogicalObjectsExport106.zip*

2. Export the document revision.

**tcxml_export -u=***username* **-p=***password* **-f=***Site_Consolidation_xml_output*
**-item=***document_rev_item_id* **-low_level -force_retraverse**

# 28. Setting up the Active Workspace client for Document Management tasks

## Requirements for setting up the Active Workspace for Document Management

Before you set up Active Workspace for Document Management, you must deploy Active Workspace with the following server extensions and client features:

- **Base Install→Active Workspace→Server Extensions→Active Workspace Document Management** and **Digital Signatures**
- **Base Install→Active Workspace→Client→Document Management Client**, **Digital Signatures**, and **Markup**
- **Base Install→Active Workspace→Visualization Server→Visualization Server Manager**

For installing these features, refer to *Active Workspace Deployment and Configuration*.

## Set the drag and drop behavior to create document revisions in Active Workspace

By default, when files are dragged from the computer to a folder in Active Workspace, they are created as standalone datasets and attached to the folder. You can change this behavior by using the **Dma1CreateDocOnDrop** preference. The default value of this preference is blank. You can change the value to:

- An internal name of an object type, for example, **Document**. If set to **Document**, when files are dragged to a folder, a document revision is created within the folder and the dragged files are attached to the document revision as datasets. However, when files are dragged to a non-folder object, a document revision is not created. The dragged files are created as standalone datasets instead.

  You can also use the **Dma1CreateDocOnDrop** preference to create other items, such as a part. For this, you must set the value of the preference to **Part**.

- **ShowCreatePanel:***object_type*, which changes the default behavior to display the **Add** panel when files are dragged to a folder. In this panel, authors enter or modify the necessary document attributes and optionally send the document revision to a workflow.

  **Example**

  If you set the value of the **Dma1CreateDocOnDrop** preference to:

- **ShowCreatePanel:Document**, the **Add** panel for creating a document revision is displayed.
- **ShowCreatePanel:Drawing**, the panel for creating a drawing revision is displayed.
- **ShowCreatePanel:Document&Drawing**, both options are displayed on the user interface. The user can choose one option to create either a document or a drawing revision.
- **ShowCreatePanel:Sdm0GenDoc**, the **Fnd0DM_IsSimpleDocumentEnabled** preference is set to **true**, and the required IRDC rule is set, Active Workspace enables the display of the icon of the file type that you added. For example, when you create a Document object and attach a Microsoft Word file, Active Workspace displays the Word icon for the simplified document.

## View the document page type in Active Workspace

By default, the **Document Page** type relation of a document is not displayed on the Active Workspace user interface, for example, cover page. However, you can modify the **Awp0DocumentRevSummary** style sheet to include the **Fnd0DocPageTypeRel.Dataset** relation so that it is displayed in Active Workspace.

```
...
<section titleKey="tc_xrt_files">
<objectSet source="TC_Attaches.Dataset, IMAN_reference.Dataset,
Fnd0DocPageTypeRel.Dataset"
 defaultdisplay="listDisplay" sortby="object_string" sortdirection="ascending">
   <tableDisplay>
      <property name="object_string"/>
      <property name="object_type"/>
      <property name="relation" modifiable="true"/>
      <property name="release_status_list"/>
      <property.name="date_released"/>
      <property name="owning_user"/>
   </tableDisplay>
...
```

However, once you set the value of the **modifiable** attribute of the **relation** property to **true**, the user can modify all the relations that are displayed on the user interface.

## Enable PDF streaming in the Active Workspace universal viewer

To improve the performance of the Active Workspace universal viewer while loading PDF files, you must enable PDF streaming. When you stream a PDF file, each page of the file is displayed as it is downloaded. Therefore, you need not to wait for the entire PDF to be downloaded to view it.

To enable PDF streaming, add **pdf** to the **FSC_DoNotCompressExtensions** property in the FSC configuration file. PDF streaming reduces the time taken to display the first page of a PDF file in the universal viewer. The reduction in the time depends on the size of and the number of pages in the PDF file. However, if you enable PDF streaming, PDF files are not compressed when they are downloaded. Due to this, it may take longer to download PDF files. It also uses more network bandwidth to complete the download.

You can enable PDF streaming only for PDF files that are optimized for page-by-page viewing. To verify that a PDF file is optimized by this type of viewing:

1. Open the PDF file in Adobe Acrobat or Reader.

2. Click **File→Properties** and verify that the **Fast Web View:Yes** is displayed.

To generate PDF files that are optimized for **Fast Web View** when using Visualization Convert, set `PDFLinearized=om` in the converter section of the *vvcp.ini* configuration file.

PDF streaming also requires the web server to return the **Content-Length** HTTP response header for a PDF file. Currently, only IIS servers return this header. IBM WebSphere do not return this header.

# Standardize document behavior in Active Workspace

An *item revision definition configuration (IRDC)* standardizes the behavior of a document at specific stages of its life cycle, including the create, checkin, checkout, revise, and save as actions. An IRDC is provided by default when you install the **Active Workspace Document Management** feature. When you render a Microsoft Word, Excel, PowerPoint, or an image dataset to a PDF dataset, IRDC uses the **TC_Derived** relation.

By default, the IRDC is applied to **Document Revision** and **Drawing Revision**. To change this behavior, modify the **Dma1UseCOTSIRDC** preference for the Drawing Revision business object and modify the **Fnd0DM_IsSimpleDocumentEnabled** preference for the Document Revision business object.

> Note:
>
> From Active Workspace 6.0 onwards, a Microsoft Word file will not be created automatically when you create a document object using the Document Management template. This is because the Dma1COTSDMTemplate for Document Management Template is removed and no longer available.
>
> If you want to create a Microsoft Word document automatically, the administrator must perform the following tasks:
>
> 1. Create the IRDC for Document Revision business object.
>
> 2. Create the Document Management template (DMTemplate).
>
> 3. Specify the DMTemplate in the Create Template field in the IRDC.

## Apply a workflow to a document in Active Workspace

To apply the required workflows when users create a document in Active Workspace, you must set the **WRKFLW_show_non_condition_templates** preference to **true**. For more information about the **WRKFLW_show_non_condition_templates** preference, see Use conditions to filter workflow template availability.

### Enable the setting for simplified document behavior in Active Workspace

By default, the **Fnd0DM_IsSimpleDocumentEnabled** preference is set to **true**. When this preference is set to **true** and the following other conditions are met, Active Workspace enables the display of the icon of the file type that you added.

- **The document type, dataset (file) type, and its relation must be already defined in IRDC.**
- The dataset (file) that will be attached to the document object must be one of the types of the datasets that are already defined in IRDC and the dataset's relation must also be one of the relations that are already defined in the IRDC.

For example, when you create a Document object that has an IRDC that lists MSWordX as a source Dataset type and attach only a single Microsoft Word file, Active Workspace displays the Word icon for the simplified document.

Additionally, when you attach another file to a document, Active Workspace performs the following tasks:

- Adds the file to the Document object.
- Displays the icon of the Document object.

Consider the example of a simplified document that contains a Microsoft Word file and displays the Word icon. When you attach a Microsoft Excel file to the same simplified document, Active Workspace changes the object type to Document, adds the Microsoft Excel file to the Document object, and displays the Document icon.

When the **Fnd0DM_IsSimpleDocumentEnabled** preference is set to **false**, Active Workspace displays the icon of the Document object by default.

## Create shared stamps for markup

You can create and share stamps so that authors and reviewers can use the stamps to mark up their documents. To create shared stamps:

1. Log on to Active Workspace with DBA credentials.
2. Create a document revision:
   a. In the **Explorer** or any other folder of your choice, choose **More Commands** > **New** > **Add** ⊕.
   b. In the **Add** panel, locate **document** under **OTHERS**.
   c. Enter the required details for the document and click **Add**.

3. From the **FILES** section of the document revision, select the file to which you want a stamp.
4. Click **Show Markup Panel**.
5. Create a freehand or a GD&T markup. For example, to create a freehand markup:
   a. Click **Create Freehand**.
   b. Draw an area for the stamp and click .
6. In the **Markup** panel, click **Insert Image**.
7. In the **Image Properties** dialog box, choose the image that you want to add as a stamp and click **OK**.
8. To include text as a stamp, type the text in the comment box.
9. In **Text Shown**, select **First Line**.
10. In **Share**, select **Public**.
11. Select the **Create Shared Stamp** check box.
12. Enter **Stamp Name** and click **Create**.

# Enable the viewing of 2D images, and electrical and printed circuit board schematics in Active Workspace

## Enable the viewing of 2D images

By default, users can only view 2D images of type **TIF**, **HPGL**, and **DXF** in the **VIEWER** pane of Active Workspace.

To enable the viewing of all types of 2D images:

1. Remove the following value from the **AWC_defaultViewerConfig.VIEWERCONFIG** preference:

   ```
   Image.AWp0ImageViewer = Image
   ```

2. Add the following value:

   ```
   Image.Awp02dViewer = Image
   ```

### Enable the viewing of the electrical and printed circuit board schematics

1. Add the following values in the **AWC_defaultViewerConfig.VIEWERCONFIG** preference:

   - `PCBFATF.Awp02dViewer=XFatf`
   - `SCHFATF.Awp02dViewer=XSCH`

2. Add the following values in the **VMU_Datasets** preference: `SCHFATF` and `PCBFATF`.

3. Add the following values in the **VMU_FileSearchOrder** preference: `XSCH` and `XFatf`.

## Install Teamcenter Client Acrobat Launcher for opening a PDF document in Adobe Acrobat or Reader DC

After installing Teamcenter Client Acrobat Launcher, users can open a PDF document from Active Workspace in Adobe Acrobat or Reader DC and then perform the following tasks:

- Add markups
- Digitally sign the PDF file

However, for this you must:

- Install **Active Workspace Launcher**.
- Install Security Services Session Agent. After you install it, set the Session Agent environmental variable **TCSSO_SESSION_AGENT_PATH** to the path where you installed Session Agent. For more information, see *Installing the Session Agent* in *Security Services Configuration*.
- Install Adobe Acrobat or Reader DC.

  You can either use 32-bit or 64-bit of Adobe Acrobat or Reader DC considering your business requirement.

### Install Teamcenter Client Acrobat Launcher from Deployment Center

1. Log on to Deployment Center.
2. Click the **SOFTWARE REPOSITORIES** tile and verify the availability of the software media added to the Teamcenter Foundation and Active Workspace software kits.
3. On the Deployment Center home page, click the **ENVIRONMENTS** tile to view the available environments in Deployment Center. Select the environment on which you want to install Teamcenter Client Acrobat Launcher.

- If you are creating a new environment, click the **Add** button on the **Environments** page.
- Add an existing environment by using the **send_configuration_to_dc** utility.

4. Click the **Deploy Software** tab.
   a. In the **Software** task, select **Foundation** from the **Available Software** list.
   b. Click **Update Selected Software** to add **Foundation** to the **Selected Software** list.
   c. After the list is updated, click **Options**.
5. In the **Options** tab, select the **Distributed** environment type and based on your environment, select the appropriate architecture type. Click **Save Environment Options**.
6. In the **Applications** tab, choose **Edit Selected Applications** ⊕. From the **Available Applications** list:
   - Select **Teamcenter** > **Foundation** > **Content and Document Management** > **Teamcenter Client Acrobat Launcher**.

   After you select the **Teamcenter Client Acrobat Launcher** application, the applications **Acrobat/ Reader Plugin** and **Teamcenter Client Communication System (TCCS)** are selected automatically as they are dependent applications.
7. Click **Update Selected Applications** and verify that the application you selected to install is listed in the **Selected Applications** list.
8. Click **Go to Components** to view the **Components** tab.
   a. Verify that **Document Management Teamcenter Client PDF Sign** is listed as one of the components.

   > Note:
   >
   > The **Teamcenter Client Acrobat Launcher** application and the **Document Management Teamcenter Client PDF Sign** application share the same component.

   b. Start the individual components and bring each component to a 100% complete status by supplying the required settings for each. While supplying settings, record values such as those of installation paths for later reference.
   c. Click **Save Component Settings**.
9. Once the configuration displays the status as 100% complete in the **Selected Components** list, click **Go to Deploy** to generate the script.

10. In the **Deploy** task, click **Generate Install Scripts**.

11. Copy the scripts to the target machine and run the scripts. Follow the instructions specified in *Run the deployment scripts*, available in the Deployment Center documentation.

12. To verify if **Teamcenter Client Acrobat Launcher** installed successfully, check if the *TcAcrobatLauncher.jar* file is available at *C:\Program Files (x86)\Siemens\Teamcenter\WSLauncheri*.

### Install Teamcenter Client Acrobat Launcher by using a ZIP file

Before installing **Teamcenter Client Acrobat Launcher**, ensure that Teamcenter client communication system (TCCS) is installed. Additionally, verify that the **FMS_HOME** environmental variable is set to the path where TCCS is installed.

To install **Teamcenter Client Acrobat Launcher** manually:

1. In the Teamcenter release software location, navigate to *Tc release_wntx64\wntx64\additional_applications*.

2. Extract the contents of *tcarobatlauncher.zip* to the location where you installed Active Workspace Launcher, for example, *C:\Program Files (x86)\Siemens\Teamcenter\WSLauncheri*.

   Verify that the *TcAcrobatLauncher.jar* file and the *lib* folder are available at this location.

### Install Teamcenter Acrobat/Reader plugin by using a ZIP file

Before installing **Teamcenter Acrobat/Reader plugin**, ensure that Adobe Acrobat/Reader DC application is installed. The Adobe Acrobat/Reader DC application can be either 32-bit or 64-bit application.

To install Teamcenter Acrobat/Reader plugin manually:

1. In the Teamcenter release software location, navigate to *Tc release_wntx64\wntx64\additional_applications*.

2. Perform one of the following actions:

   - For Acrobat/Reader (32-bit) application, extract the contents of *vmeplgin.zip* to a temporary directory, for example, *C:\temp\vmeplgin directory*.

   - For Acrobat/Reader (64-bit) application, extract the contents of *vmeplgin_x64.zip* to a temporary directory, for example, *C:\temp\vmeplgin_x64 directory*.

3. In the Teamcenter release software location, navigate to *Tcrelease_wntx64\wntx64\additional_applications*.

4. Perform one of the following actions:

- For Acrobat/Reader (32-bit) application, perform the following steps:

  a. Extract the contents of the *vmeplgin.zip* file to the same temporary directory from the step 2 above.

     This action might overwrite some of the files from the release.

  b. Copy both the *plug_ins* and *Javascripts* directories to the location where you installed Adobe Acrobat/Reader DC (32-bit) application, for example, *C:\Program Files (x86)\Adobe\Acrobat Reader DC\Reader*.

  c. Verify that the *vmeplgin* directory is under *C:\Program Files (x86)\Adobe\Acrobat Reader DC\Reader\plugins directory*.

- For Acrobat/Reader (64-bit) application, perform the following steps:

  a. Extract the *vmeplgin_x64.zip* file to the same temporary directory from the step 2 above.

     This action might overwrite some of the files from the release.

  b. Copy both the *plug_ins* and *Javascripts* directories to the location where you installed Adobe Acrobat/Reader DC (64-bit) application, for example, *C:\Program Files\Adobe\Acrobat DC\Acrobat*.

  c. Verify that the *vmeplgin* directory is under *C:\Program Files\Adobe\Acrobat DC\Acrobat\plugins directory*.

> Note:
>
> The 32-bit *vmeplgin.zip* must be used for the (32-bit) application and the 64-bit *vmeplgin_x64.zip* must be used for (64-bit) application.

5. **Turn off Adobe protected mode** for the Teamcenter menu to appear in Adobe Acrobat/Reader DC application.

6. Verify that the Teamcenter menu appears in Adobe Acrobat/Reader application.

   To verify the Acrobat/Reader plugin is installed and loaded successfully, start Adobe Acrobat/Reader application and then click **Help→About Third-Party Plug-ins→Teamcenter** menu. The Teamcenter plugin version information dialog box appears.

## Allow users to download the markup disposition report in Excel

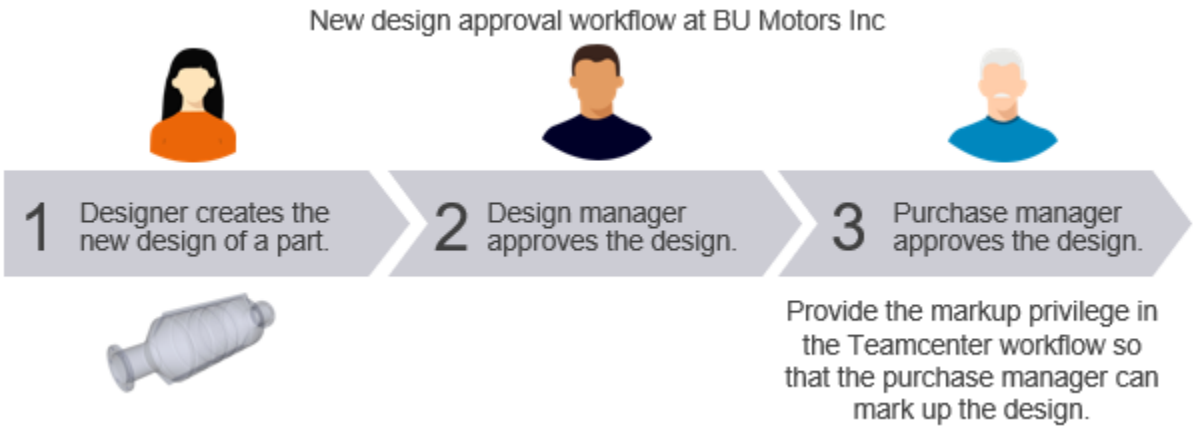
Authors and reviewers can create the markup disposition reports in HTML by default. In addition to HTML, if you set the **Crf_Report_Excel_Use_OOXML** preference to **true**, the authors and reviewers can also create and manage the markup disposition reports in Excel.

# Provide markup privileges to users in a workflow using Active Workspace

Sometimes users from different groups who are working on a Teamcenter workflow are required to provide markups on a document. Because these users are from different groups, they might not have privileges to markup the document. In such a case, as an administrator, you can modify the Teamcenter workflow and provide the markup privileges to the required users.

Example:

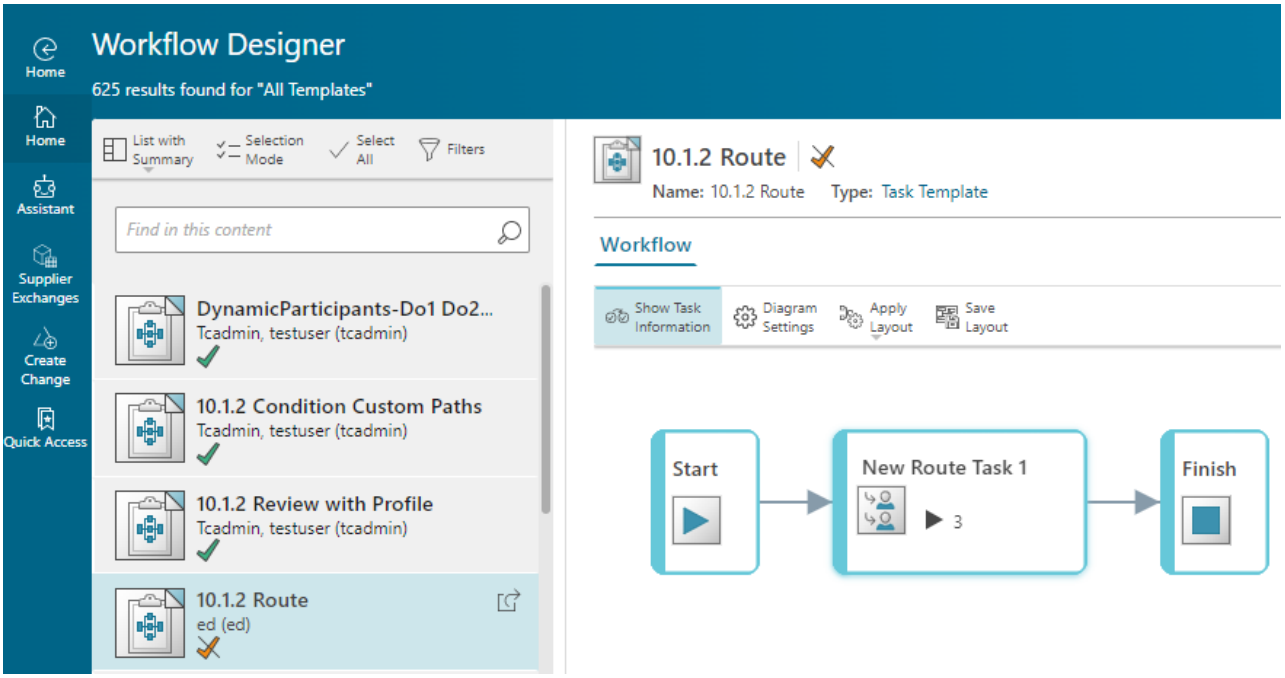
In BU Motors Inc., the purchasing manager always provides suggestions and also approves the new design of the part. By using this company-specific process, BU Motors Inc. ensures that the raw material required for manufacturing the new part can be procured within a reasonable time frame. Because the purchasing manager is from another group, as an administrator, you can modify the required Teamcenter workflow and provide the markup privilege to the purchasing manager. This way, all users involved in the Teamcenter workflow, including the purchase manager, can markup the documents.



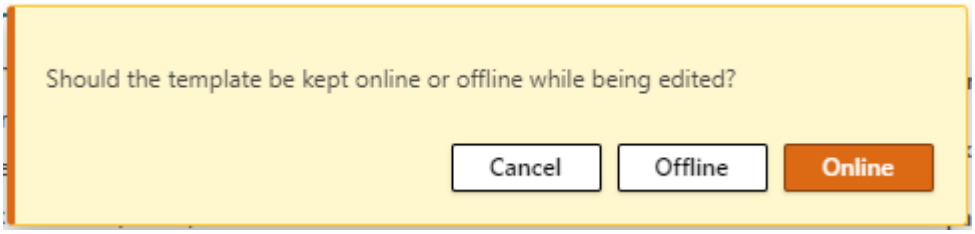

## Procedure

1. As a database administrator, select the **Active Admin** workspace, and then click the **Workflow Designer** tile.

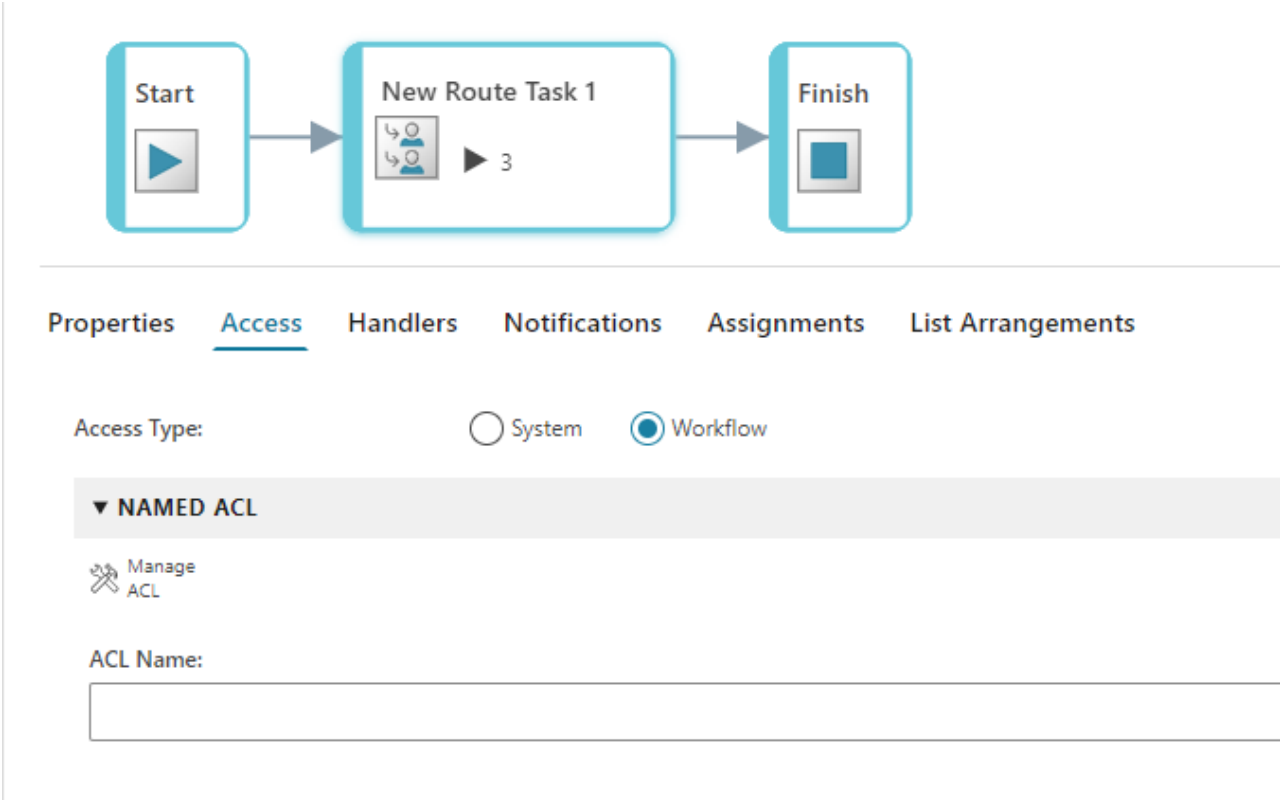
2. Search for and click a workflow process that you want to edit.




3. Go to **More Commands** ▫▫▫ > **Edit** ✎ > **Start Edit** ✎.

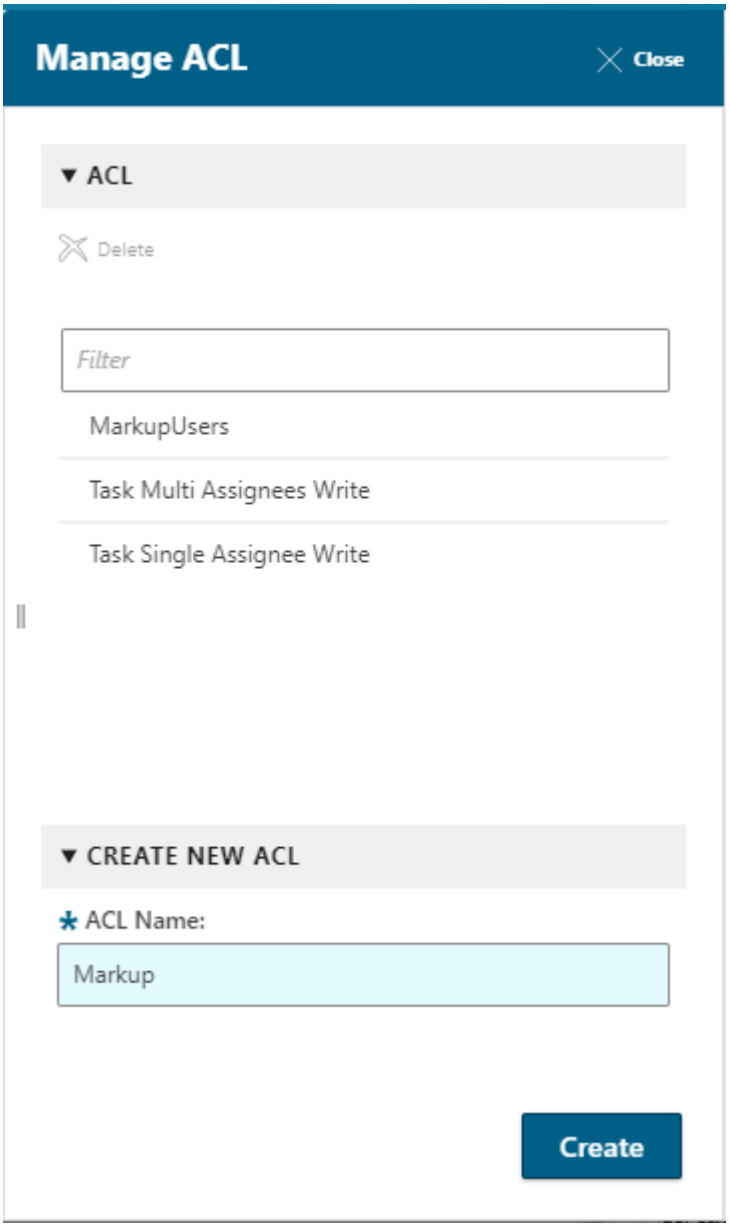



4. In the pop-up message, click **Offline**.

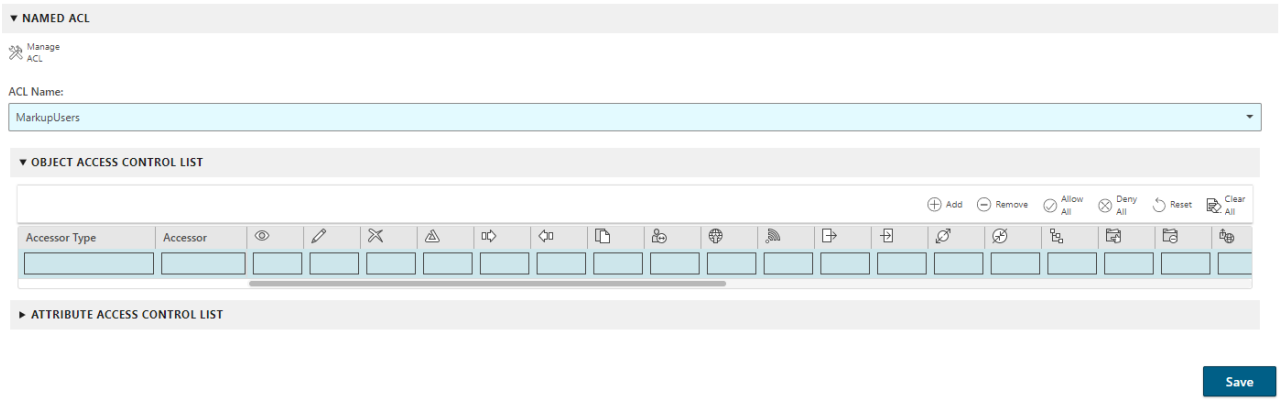



5. Click the **Access** tab.

6. Click **Manage ACL**.

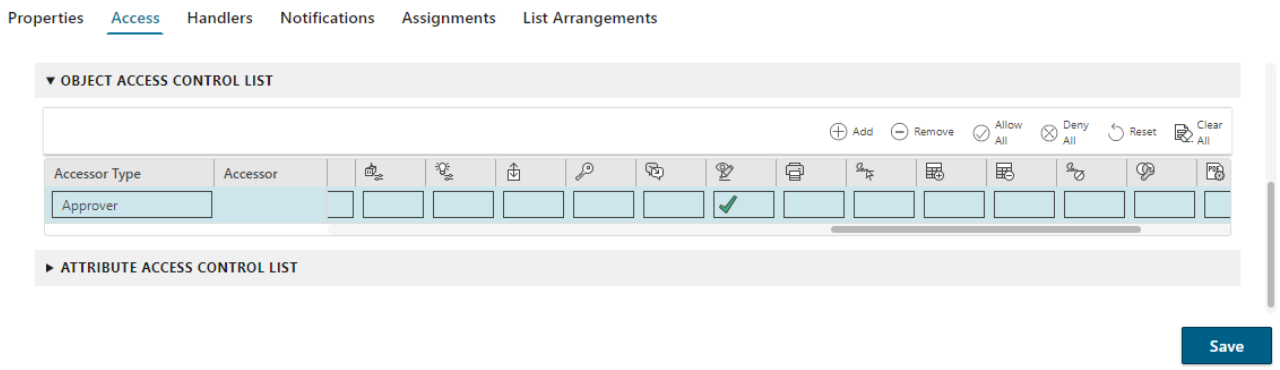



7. In the **Manage ACL** panel, enter a name for the ACL and click **Create**.
8. In the **ACL Name** list, click the newly created ACL.
9. In the **OBJECT ACCESS CONTROL LIST** table, click **Add** ⊕.



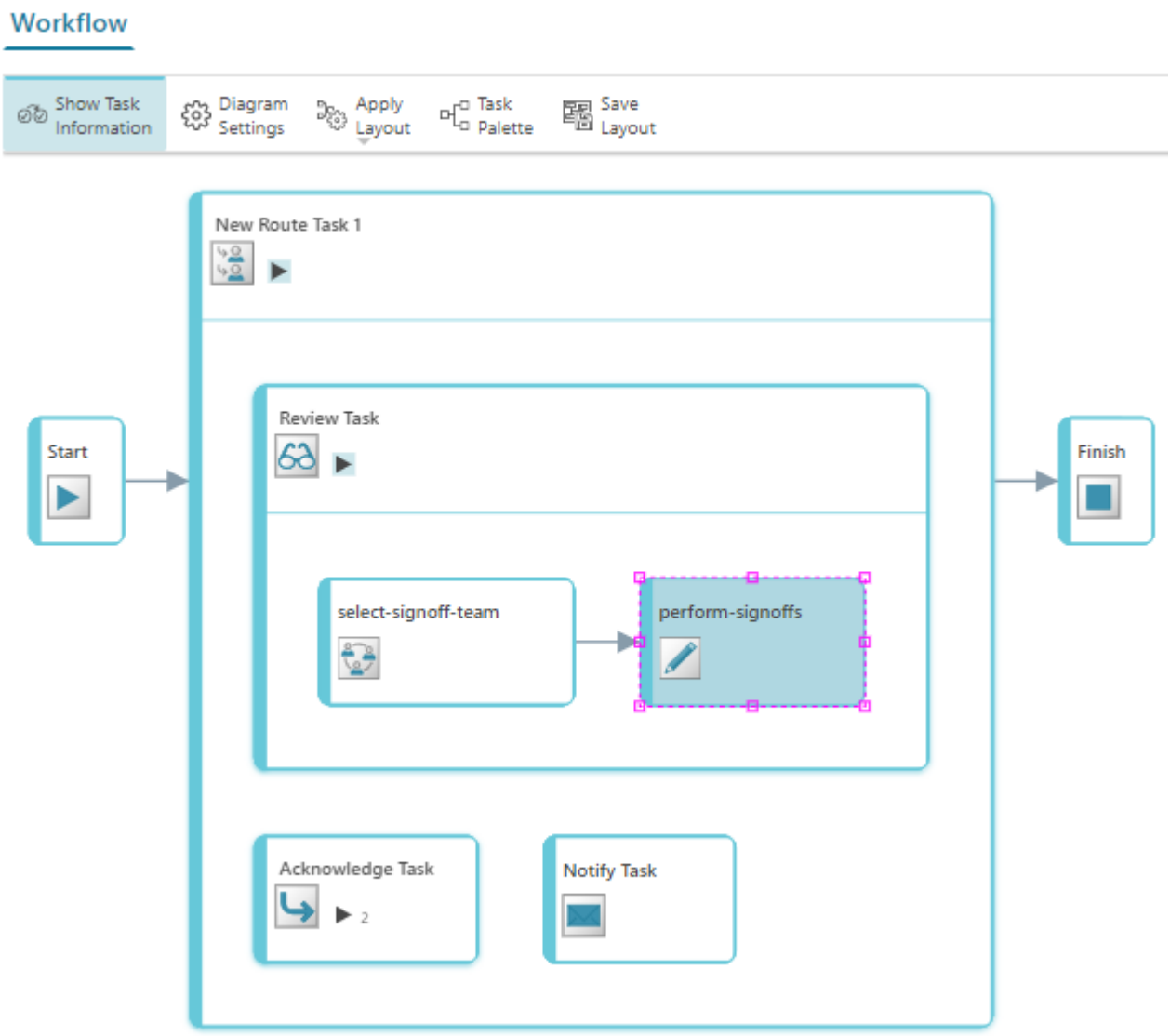

10. Double-click the **Accessor Type** column, and then click **Approver**.

    You can also add users in addition to adding the **Approver**.



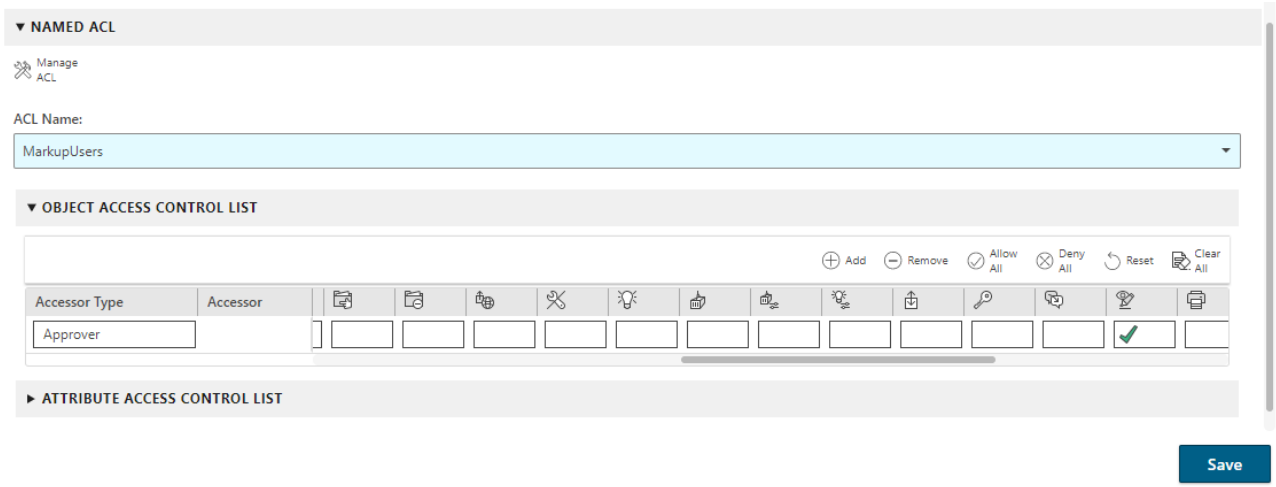

11. Double-click the **Markup** column, and then click ✓.

12. Click **Save**.




13. Expand the workflow, and click the **perform-signoffs** box.

14. Click the **Access** tab, and then click the newly added ACL name in the **ACL Name** list.




15. Click **Save**.

    A pop-up message appears and indicates that the access control entries are saved.

16. Go to **More Commands** ▫▫▫ > **Edit** > **End Edit**.

17. Go to **More Commands** ▫▫▫ > **Edit** > **Online**.