

# TEAMCENTER

## Access Management Using Projects and Programs

Teamcenter 2412

Unpublished work. © 2025 Siemens

This Documentation contains trade secrets or otherwise confidential information owned by Siemens Industry Software Inc. or its affiliates (collectively, "Siemens"), or its licensors. Access to and use of this Documentation is strictly limited as set forth in Customer's applicable agreement(s) with Siemens. This Documentation may not be copied, distributed, or otherwise disclosed by Customer without the express written permission of Siemens, and may not be used in any way not expressly authorized by Siemens.

This Documentation is for information and instruction purposes. Siemens reserves the right to make changes in specifications and other information contained in this Documentation without prior notice, and the reader should, in all cases, consult Siemens to determine whether any changes have been made.

No representation or other affirmation of fact contained in this Documentation shall be deemed to be a warranty or give rise to any liability of Siemens whatsoever.

If you have a signed license agreement with Siemens for the product with which this Documentation will be used, your use of this Documentation is subject to the scope of license and the software protection and security provisions of that agreement. If you do not have such a signed license agreement, your use is subject to the Siemens Universal Customer Agreement, which may be viewed at <https://www.sw.siemens.com/en-US/sw-terms/base/uca/>, as supplemented by the product specific terms which may be viewed at <https://www.sw.siemens.com/en-US/sw-terms/supplements/>.

SIEMENS MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS DOCUMENTATION INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT OF INTELLECTUAL PROPERTY. SIEMENS SHALL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, CONSEQUENTIAL OR PUNITIVE DAMAGES, LOST DATA OR PROFITS, EVEN IF SUCH DAMAGES WERE FORESEEABLE, ARISING OUT OF OR RELATED TO THIS DOCUMENTATION OR THE INFORMATION CONTAINED IN IT, EVEN IF SIEMENS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

TRADEMARKS: The trademarks, logos, and service marks (collectively, "Marks") used herein are the property of Siemens or other parties. No one is permitted to use these Marks without the prior written consent of Siemens or the owner of the Marks, as applicable. The use herein of third party Marks is not an attempt to indicate Siemens as a source of a product, but is intended to indicate a product from, or associated with, a particular third party. A list of Siemens' Marks may be viewed at: [www.plm.automation.siemens.com/global/en/legal/trademarks.html](http://www.plm.automation.siemens.com/global/en/legal/trademarks.html). The registered trademark Linux® is used pursuant to a sublicense from LMI, the exclusive licensee of Linus Torvalds, owner of the mark on a world-wide basis.

## 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](http://support.sw.siemens.com)

Send Feedback on Documentation: [support.sw.siemens.com/doc\\_feedback\\_form](http://support.sw.siemens.com/doc_feedback_form)

# Contents

What are projects and programs?	7
Frequently asked questions for project-level security	7
Before you begin	8
Characteristics of projects and programs	10
Project administrators and team members	11
Ownership of data by a project or program	12
Assigning data to projects	13
<b>Project Administration window</b>	
Project and Program quick links	15
Project administration buttons	15
Project administration tabs	16
<b>Smart Folder Administration window</b>	
Smart folder administration buttons	17
<b>Creating and administering projects</b>	
Using projects	1-1
Search for a project or program	1-1
Search for users, groups, or roles in your projects and programs	1-1
Filter user display in the project member tree	1-3
<b>Creating and modifying projects and programs</b>	1-4
Create a project or program based on an existing project or program	1-4
Create a new project or program	1-4
Control data access in program and child projects	1-6
Control data access based on a project category	1-11
Build hierarchy by adding new child projects under a program	1-12
Build hierarchy by adding existing projects to a new program	1-13
Inherit team members from a parent program	1-14
Modify projects and programs	1-14
Set project status	1-15
Delete a project or program	1-15
<b>Associating data dictionaries with projects</b>	1-16
What are data dictionaries?	1-16

Add a data dictionary to a project or program	1-16
<b>Assigning users to projects</b>	1-17
Assign individual users to a project or program	1-17
Assigning users to a project using a workflow	1-17
Assign groups of users to a project or program	1-19
Assign all users in a role to a project	1-20
Select team administrators	1-21
Designate privileged team members	1-22
<b>Applying security rules to projects and programs</b>	1-23
Overview of applying project and program security (Access Manager) rules	1-23
Default security rules for projects and programs	1-24
Configure evaluation of rules that include the Role in Project and Role in Projects of Object accessors	1-28
Configure evaluation of the ASSIGN_TO_PROJECT and REMOVE_FROM_PROJECT privileges	1-31
Add project categories	1-32
Example of defining Access Manager rules to restrict access based on user and project category	1-36
<b>Set a user's default project or program</b>	1-40
<b>Considerations for importing and exporting project or program data in a Multi-Site Collaboration environment</b>	1-41

## Configuring projects and programs

<b>Project and program configuration tasks</b>	2-1
<b>Default naming rules for projects</b>	2-1
<b>Configuring assignment of related secondary objects</b>	2-1
Assigning secondary objects to projects and programs	2-1
Secondary object propagation rules for projects and programs	2-2
Assigning BOM data to projects and programs	2-2
<b>Configuring the automatic assignment of objects</b>	2-4
Assign projects to objects using workflow	2-4
Automatically assigning objects to projects and programs	2-5
Implications of the autoAssignToProject extension on propagation rules	2-6
<b>Restricting data creation to within the context of a project or program</b>	2-8
<b>Controlling assignment of replicated (imported) data to projects</b>	2-9
<b>Setting the current project for new Teamcenter sessions</b>	2-9
<b>Selecting different object property values based on project assignment</b>	2-9
<b>Displaying projects to users based on current group and role</b>	2-9
<b>Using pseudofolders to organize the display of objects in projects</b>	2-10

## Configuring smart folder filtering

<b>What are project smart folders?</b>	3-1
<b>Understanding the project smart folder hierarchy</b>	3-2
<b>Top-level smart folder hierarchy</b>	3-2
<b>Top-level hierarchy example</b>	3-2
<b>Creating the example project top-level hierarchy</b>	3-3

<b>Associating data objects with smart folders</b>	3-4
<b>Using properties to filter data</b>	3-4
<b>Project smart folder hierarchy</b>	3-5
<b>Project smart folder hierarchy example</b>	3-5
<b>Apply smart folder filtering</b>	3-7
<b>Viewing project data</b>	3-9
Viewing project data in My Teamcenter	3-9
Project smart folder filtering	3-10
Apply project smart folder filtering	3-11



# What are projects and programs?

Project, an application in Teamcenter Foundation, helps you organize data and grant or control access to a collection of related data that may be accessible to multiple organizations, such as project teams, development teams, suppliers, and customers.

## Similarities of projects and programs

Projects and programs enable you to:

- Segregate data and users.
- Write access rules based on a project or program.

## Differences between projects and programs

*Programs* can have projects under them, whereas *projects* cannot have programs under them.

Another difference is that programs use program access rules. For example, the Aerospace and Defense industry works with programs because of the *high level of security provided with the program security rules*. Program-level security can be enabled to restrict access to program data. The default access rules can be extended to grant read access to program data to members of the program team, on a program-by-program basis.

Note:

When the program security attribute on a project is set to *true*, Teamcenter considers the project as a program and program-level access rules are applied.

# Frequently asked questions for project-level security

Before you create your projects and programs, consider these points.

Consideration	Description
Can a user have different privileges in the same project?	If a user has both <b>Project Author</b> and <b>Project Team Administrator</b> privileges, then the privilege with the higher security is considered, in this case, <b>Project Team Administrator</b> .
Can a user <i>not</i> see all the active projects in which a privileged team member participates?	Yes, because the <b>TC_show_all_user_projects</b> setting is set to <b>false</b> , thereby limiting the display to only the projects that correspond with the user's current group and role.

Consideration	Description
	When this preference is set to <b>true</b> , all active projects in which the user is a privileged team member are displayed.
What is the maximum number of projects you can have in Teamcenter?	<p>Although you can have any number of projects in Teamcenter, the number of projects impacts the following use cases:</p> <ul style="list-style-type: none"> <li>• Project application load time in the rich client.</li> <li>• Object load time.</li> </ul> <p>For example, if the object is assigned to projects, access evaluation checks if the logged on user is a member of the projects assigned to the object to evaluate the <b>In Project</b> and <b>In Current Program</b> conditions.</p> <ul style="list-style-type: none"> <li>• Project <b>Assign/Remove</b> dialog load time in the rich client.</li> <li>• User session dialog load time in the rich client.</li> <li>• Complex organization structure, such as when the logged on user belongs to a multi-level subgroup and the parent group is assigned to the project.</li> </ul> <p>For optimal performance, Siemens Digital Industries Software recommends the number of projects be between 5,000 to 10,000 projects. This includes all active and inactive projects. A potential mitigation to enhance the performance is to:</p> <ul style="list-style-type: none"> <li>• Deactivate projects that are no longer needed.</li> <li>• Set your <b>TC_show_all_user_projects</b> preference to false.</li> </ul>

## Before you begin

### Prerequisites

Access to Project and Program is controlled by your Teamcenter administrator.

Only users in the following roles and groups, or users in the **DBA** role within the **dba** group, can perform tasks in Project and Program:

- **Project Administrator** in the **Project Administration** group can create, modify, and delete projects and programs. They can also add and remove team members from projects and programs. However, the **Project Administrator** must belong to the **dba** group to add access rules.

Note:

Users in the **Project Administrator** role can only manage projects and programs that they own. They cannot manage projects or programs owned by other users in the **Project Administrator** role. Ownership is established when the project is created.

- **Team Administrator** in the **Project Administration** group can modify project and program information and add and remove team members from the project or program. However, they cannot create or delete a project.

## Enable Project and Program

Project does not need to be enabled before you use it, but during installation, this feature must be selected.

The system can be configured to display the Project application as Program.

To do this, set the display name of the **TC\_Project** business object in the *BMIDE for Data Model Design* to **Program**.

Note:

The Project application is automatically displayed as Program when you install the ADS Foundation template.

If you have trouble accessing Project or Program, see your system administrator; it may be a licensing issue.

Note:

You can log on to Teamcenter only once. If you try to log on to more than one workstation at a time, you see an error message.


## Configure Project and Program

You can set various options to:

- Configure the level of propagation applied when objects are assigned to projects or programs.
- Specify whether project or program security rules are applied to all projects and programs or only to the user's current project or program
- Specify whether BOM views and BOM view revisions are assigned to the project or program when the items or item revisions are assigned to the project or program.

Default rules for project-level or program-level security are automatically defined when a project or program is created. Additional security rules can be configured in Project or Program by a project administrator or in Access Manager by a Teamcenter administrator.

Start Project or Program

Click **Project**  in the navigation pane.

## Characteristics of projects and programs

*Projects and programs* organize data and are the basis for granting data access to project and program team members. The following concepts apply to projects and programs:

- Only privileged team members or regular team members who are explicitly granted **ASSIGN\_TO\_PROJECT** or **REMOVE\_FROM\_PROJECT** privileges can assign data to and remove data from projects and programs.

The **TC\_project\_validate\_conditions** preference controls which team roles and access privileges are required to add and delete projects.

- Project and program names must be unique within your site. Projects and programs cannot have the same name as any group at the site.
- Data can be assigned to or removed from projects and programs manually or automatically when the data item is created, and items can be assigned to more than one project or program.
- Propagation rules define the associated data that is implicitly assigned to a project or program when a primary item is assigned to the project or program.
- All items in a complete product structure can be assigned to a project or program using the **update\_project\_bom** utility.
- Ownership of data can be assigned to a project or program by configuring the **autoAssignToProject** extension in the Business Modeler IDE.
- Creation and maintenance of data can be restricted to within the context of a program.

Note:

When the program security attribute on a project is set to **true**, the project is considered to be a program and is subject to program-level access rules.

Programs offer all the basic features of projects, but in addition you can:

- Control access to program data at a higher level than typically applied to project data.
- Share data between programs by assigning the data to multiple programs.

## Project administrators and team members

The following table describes the project administrators and team members who can create, manage, and use projects and programs.

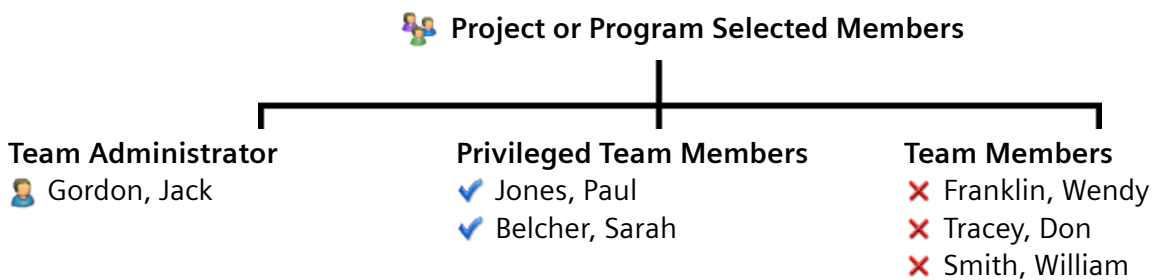
Team role	Definition
<b>Project Administrator</b> role in <b>Project Administration</b> group	<p>Teamcenter user with privileges to create and administer projects and programs.</p> <p>Users in the <b>Project Administrator</b> role can:</p> <ul style="list-style-type: none"> <li>• Modify projects and programs.</li> <li>• Delete projects and programs.</li> <li>• Add team members to projects and programs.</li> <li>• Assign privileges to team members.</li> <li>• Remove team members from projects and programs.</li> </ul>
<b>Team Administrator</b> role in <b>Project Administration</b> group	<p>Project or program team members with privileges to modify project or program information. These privileges apply to the project or program metadata, not to the data assigned to projects and programs.</p> <p>Users in the <b>Team Administrator</b> role can:</p> <ul style="list-style-type: none"> <li>• Add team members to projects or programs in which the team administrator is also a member.</li> <li>• Remove team members from projects or programs in which the team administrator is also a member.</li> </ul>

Note:

These privileges only apply to projects and programs that the project administrator owns.

Team role	Definition
	You can designate multiple team administrators for each project or program. This is often necessary to balance resource management tasks for large projects.
Privileged team members	Project or program team members with privileges to assign or remove objects from their projects or programs.
Team members	Team members (users) who do not have privileges to assign objects to or remove objects from their projects or programs.

The following diagram illustrates a typical project or program hierarchy.



## Ownership of data by a project or program

The following concepts apply to ownership of data by a project or program:

- Ownership of data by a project or program is established when an object is created.
- Ownership is assigned to the user's current project or program. Users can change the current project in their user session by choosing **Edit**→**User Setting** and selecting a project or program from the list.
- Users can be prevented from creating an object without an owning project or program. This can be accomplished in two ways:
  - Set the value of the **CreateDataWithNoActiveProgram** global constant to **false**.
  - Configure the **autoAssignToProject** extension in the Business Modeler IDE to automatically assign objects to projects upon creation.
- Owning project or owning program can be used as a basis for controlling access to data.
- Once established, the relationship between an object and its owning project can only be broken if the **TC\_allow\_remove\_owning\_project** preference is set to allow users to remove objects from owning projects.

# Assigning data to projects

End users can manually assign data objects to projects and programs. In addition, you can configure Teamcenter to automatically assign related objects to a project or program when the primary object is assigned to the project or program, and you can use the **update\_project\_bom** utility to assign or remove objects in a product or manufacturing structure to or from a project or program. Teamcenter can also be configured to automatically assign data objects to projects and programs when the object is created.

Data objects can be assigned to projects and programs in the following ways:

- Manually assigned to projects or programs by users who are designated as privileged team members. Privileged team members include:
  - Project or program team administrator
  - Privileged team members

Note:

Additional project or program security can be achieved using the **ASSIGN\_TO\_PROJECT** and **REMOVE\_FROM\_PROJECT** privileges. The exact behavior of these privileges is controlled by the **TC\_project\_validate\_conditions** preference.

- Automatically assigned to projects or programs when the object is created. When a new item revision is created, it is automatically assigned to the project or program in which the user is currently working.

Teamcenter administrators define which objects can be automatically assigned to projects and programs using the Business Modeler IDE to configure the **autoAssignToProject** extension located in the **Extension Definitions** folder under **Rules**.

Note:

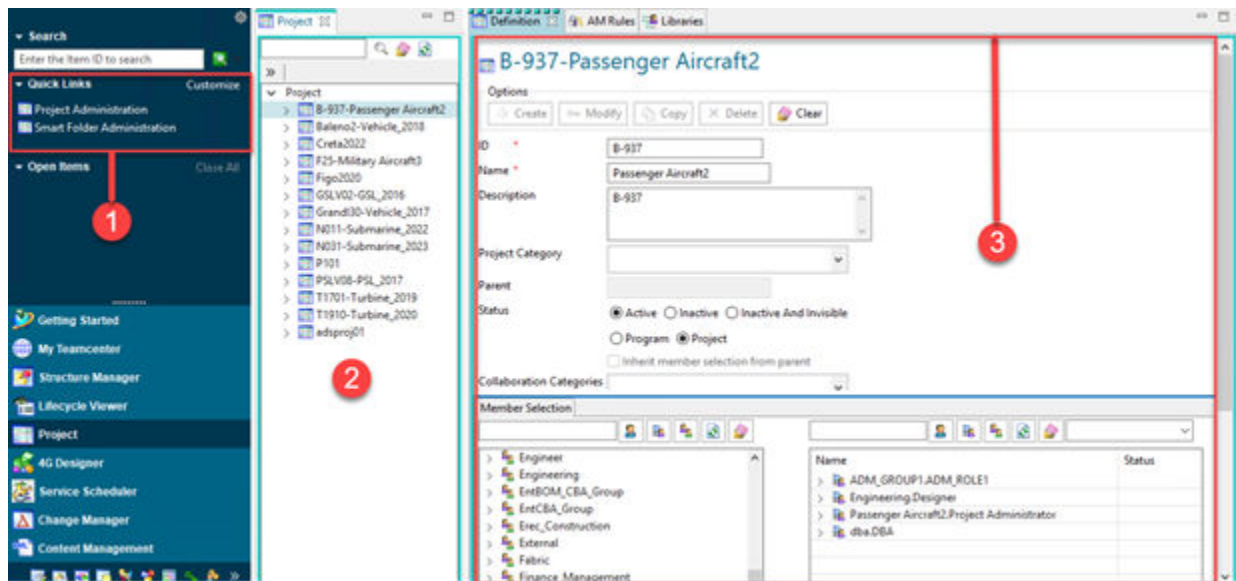
When importing or exporting project or program data in a Multi-Site Collaboration environment, you must first create the project on both the export and the import site prior to using the **autoAssignToProject** extension.

- Automatically assigned to projects or programs when the primary object is assigned to the project or program. Project and program propagation rules determine which secondary objects are assigned to a project or program when a primary object type is assigned.
- Using the **update\_project\_bom** utility to assign or remove objects in a product or manufacturing structure to or from a project or program.

Data can be assigned to projects when working in the following applications:

- My Teamcenter
- Change Manager
- Multi-Structure Manager
- Structure Manager
- Part Planner
- 4G Designer
- Manufacturing Process Planner
- Content Management

## Project Administration window



- 1 **Quick Links** Displays links to navigate between the **Project Administration** and **Smart Folder Administration** windows.
- 2 **Project or Program tree** Displays the list of projects or programs in Teamcenter. New projects and programs appear in this list.
- 3 **Project or program definition and rules** Displays information about the selected project or program. If the **Definition** tab is selected, the properties of the project or program are displayed. If the **AM Rules** tab is selected, the security rules for the project or program

are displayed. If the **Libraries** tab is selected, you can view which data dictionaries are assigned to the project or assign new dictionaries to the project.









## Project and Program quick links

There are two links in the **Quick Links** area of the navigation pane that are specific to Project and Program:

<b>Project Administration</b>	Provides access to the project administration interface used to create and activate projects and programs, assign users membership in projects and programs, and designate project and program administrators and privileged team members.
<b>Smart Folder Administration</b>	Provides access to the smart folder filter configuration interface used to define filtering criteria based on the smart folder hierarchy. These filters control how project and program data is displayed to users.

These links appear in both the **Project Administration** window and the **Smart Folder Administration** window.

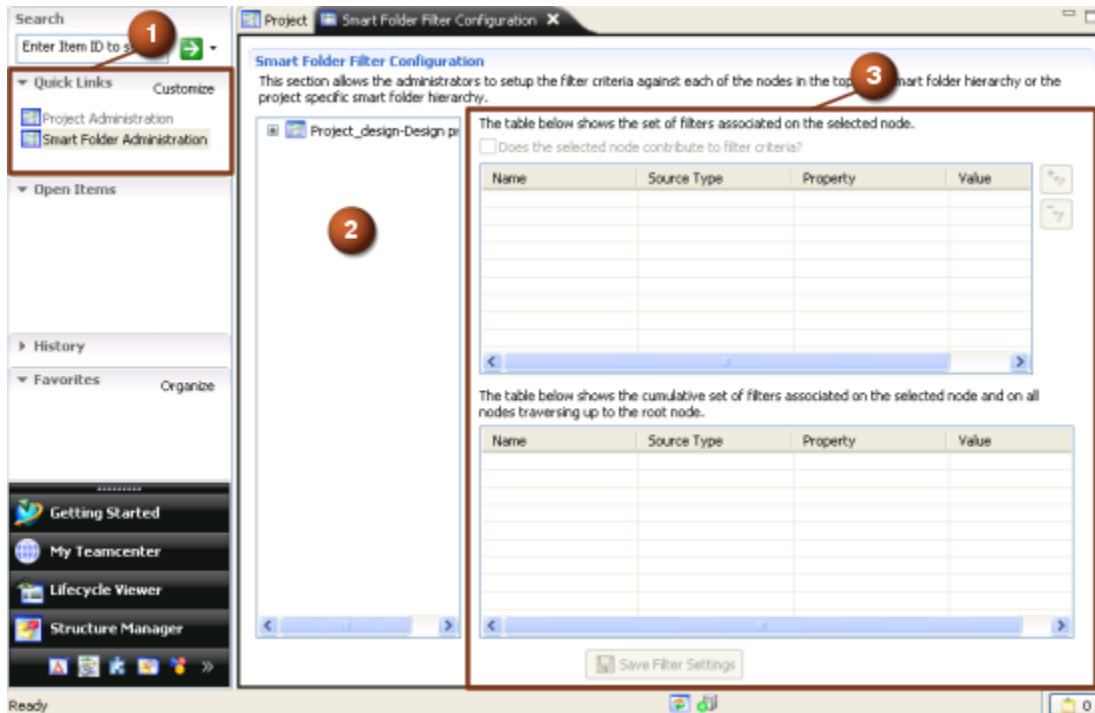
## Project administration buttons

Button	Description
 Groups	Filters the display of the organization tree or project member tree by groups. You can search for a specific group in either tree by entering the group name in the text box and clicking the <b>Groups</b> button  .
 Roles	Filters the display of the organization tree or project member tree by roles. You can search for a specific role in either tree by entering the role name in the text box and clicking the <b>Roles</b> button  .
 Users	Filters the display of the organization tree or project member tree by users. You can search for a specific user in either tree by entering the user name in the text box and clicking the <b>Users</b> button  .
 Clear	Clears the search criteria and restores the organization tree or project member tree to its previous state.
 Refresh tree	Refreshes the organization tree or project member tree and restores it to its original state.

# Project administration tabs



Tab	Description
Definition	Displays the <b>Definition</b> pane. Use this pane to create the project, to assign project team members, and to designate privileged team members.
AM Rules	Displays the <b>Access Manager Rules</b> pane. Use this pane to apply access rules to a project.
	<div style="border: 1px solid black; padding: 10px;"> <p>Note:</p> <p>Project administrators only have access to the <b>In Project</b> branch of the rule tree. You cannot modify other branches of the rule tree from within Project. In addition, moving the <b>In Project</b> branch to a different position in the tree requires Teamcenter administrative privileges and must be done using the Access Manager application.</p> </div>
Libraries	Displays data dictionaries and libraries that have been defined in the Classification Admin application, and provides tools for associating these libraries with specific projects or programs.

## Smart Folder Administration window



1	<b>Quick Links</b>	Displays links to navigate between the <b>Project Administration</b> and <b>Smart Folder Administration</b> windows.
2	Smart folder tree	Displays the <b>smart folder hierarchy</b> . The smart folder hierarchy consists of top-level abstract folders and low-level data-driven folders.
3	Filter tables	Displays the set of <b>filters</b> that apply to the selected smart folder hierarchy node.

## Smart folder administration buttons

Button	Description
 Add filter	Adds a new row to the filter table that is used to define additional filters for data-driven folders. Additional filtering criteria can only be applied to data-driven folders. Abstract folders in the hierarchy can only apply a single filter criteria.
 Remove filter	Removes the selected row from either of the filter tables.



# 1. Creating and administering projects

## Using projects


Project creation and administration tasks are performed by the Teamcenter administrator in the following order:

1. A project administrator is added to the **Project Administrator** role in the **Project Administration** group.
2. A project is created with specific groups, users or roles assigned as team members, privileged team members, and team administrators.
3. The user's default project is set.

To assign a default project, you must select the **Use Program Security** option when you define the project or program.

4. As database objects are created, they are assigned to the project automatically or manually by privileged team members.

## Search for a project or program

1. In the Project or Program application, type a keyword (project or program ID or name) in the box below the **Project** or **Program** tree.
2. Click **Search** .

Projects and programs matching the keyword are displayed in the tree.

Note:

If no projects or programs are found that match your search criteria, the **Project** tree pane is empty.

To restore the tree display, click **Reload** .

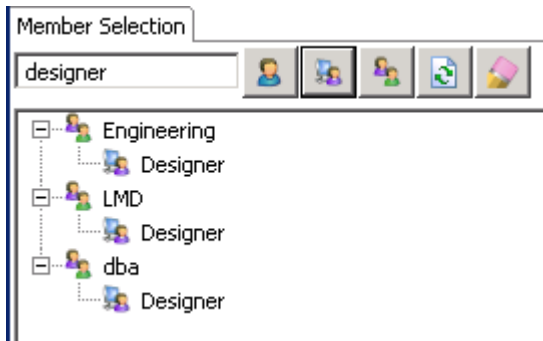
## Search for users, groups, or roles in your projects and programs

When you administer projects or programs, you will search the organization for individual users, users in certain roles, or users in certain groups to add to your projects or programs. You will also search your project membership for users, groups, and roles as part of ongoing project or program maintenance.




## Organization search

Use the search box above the tree on the left side of the **Member Selection** area to search the organization for users, groups, or roles to add as project members.

1. Enter the name of a user, group, or role in the text box.





2. Select the appropriate action.

To...	Click...
Search for a user in the organization	The <b>User</b> button  .
Search for a role in the organization	The <b>Roles</b> button  .
Search for a group in the organization	The <b>Groups</b> button  .

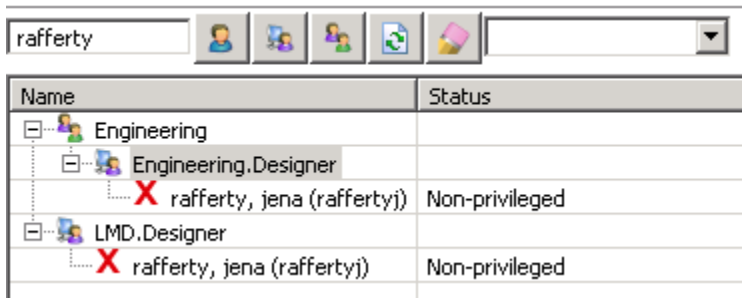
When you search for a role, the organization tree displays the role within context of the groups with which it is associated.

### Note:




If your search yields no results, the tree pane will be empty. Click the **Refresh** button  or the **Clear** button  to restore the display.

## Member search

Use the search box above the tree on the right side of the **Member Selection** area to search the organization for users, groups, or roles to add as project members.





1. Enter the name of a user, group, or role in the text box.
2. Select the appropriate action.

To...	Click...
Search for a user in the project or program	The <b>User</b> button  .
Search for a role in the project or program	The <b>Roles</b> button  .
Search for a group in the project or program	The <b>Groups</b> button  .

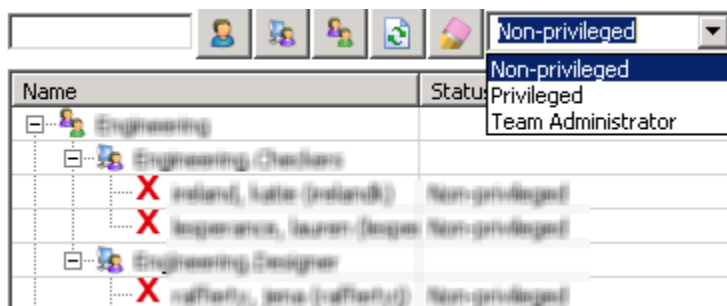
When you search for a user, the member tree displays the groups and roles to which the user belongs. In the graphic above, Jena Rafferty is a project member both in her role as a **Designer** in the **Engineering** group and as a **Designer** in the **LMD** group.

**Note:**

If your search yields no results, the tree pane will be empty. Click the **Refresh** button  or the **Clear** button  to restore the display.

## Filter user display in the project member tree

To filter the project member display, choose **Non-privileged**, **Privileged**, or **Team Administrator** from the list above the project member tree.



Click the **Refresh** button  or the **Clear** button  to restore the display.


## Creating and modifying projects and programs

### Create a project or program based on an existing project or program

You can define a new project or program by copying an existing project or program that has similar characteristics to the project or program you want to define. This allows you to leverage existing membership assignments, access rules, and associated data dictionaries (libraries).

1. In the **Definition** pane, select an existing project or program from the **Project** tree or the **Program** tree.

**Tip:**

To find a project or program in the tree, type a keyword (project or program ID or name) and click **Find** .

2. Type a new project or program ID in the **ID** box. (The maximum length for a project or program ID is 64 characters.)
3. Type a new project or program name in the **Name** box. (The maximum length for a project or program name is 32 characters.)

**Caution:**

You cannot use characters such as , % \* @, in project and program names.

4. Modify any of the project or program settings, such as team member assignments and access rules.
5. Click **Copy**.

The new project or program is created and displayed in the tree.

### Create a new project or program

1. Click the **Definition** tab.

**Tip:**

If the **Definition** tab is not displayed, click the **Project Administration** quick link.

2. Click the **Project** folder or the **Program** folder in the tree.
3. In the **Definition** pane, type a unique identifier for the project or program in the **ID** box. (The maximum length for a project or program ID is 64 characters.)

For consistency, the format of the project and program ID and name can be controlled by naming rules.

4. Type a name for the project or program in the **Name** box. The name must be unique within your site, and it cannot be the same as the name of any group at your site. (The maximum length for a project or program name is 32 characters.)

Caution:

You cannot use characters such as , % \* @, in project and program names.

5. (Optional) Type a description of the project or program in the **Description** box.
6. (Optional) Select a category from the **Project Category** drop-down list.

Project categories allow you to control access to objects in a project without using access control rules. The default categories are: **Internal**, **Partner**, and **Supplier**.

If you want to **change the listed project categories**, you can change the values in the list of values (LOV). Project categories are defined in the **Fnd0ProjectCategories** LOV found in the Business Modeler IDE.

7. Select a status for the project or program:

<b>Active</b>	Activates the project or program, which allows users to manage data within the project or program.
<b>Inactive</b>	Inactivates the project or program. Users can see the project or program, but they cannot manage data within the project or program.
<b>Inactive and Invisible</b>	Inactivates the project or program and removes it from the user's display. Only project administrators can view the data owned by an inactive and invisible project or program.

8. Select the **Program** or **Project** button.

Although Program and Project work the same, the difference between them is the program security rules. You can extend the default security rules to grant read access to program data to members of the program team, on a program-by-program basis.

9. (Optional) Select a category from the **Collaboration Categories** drop-down list.

Note:

This field is used by Mechatronics. Refer to the Mechatronics online documentation for more information.

10. **Assign individual users to a project or program** or **assign groups of users to a project or program**.

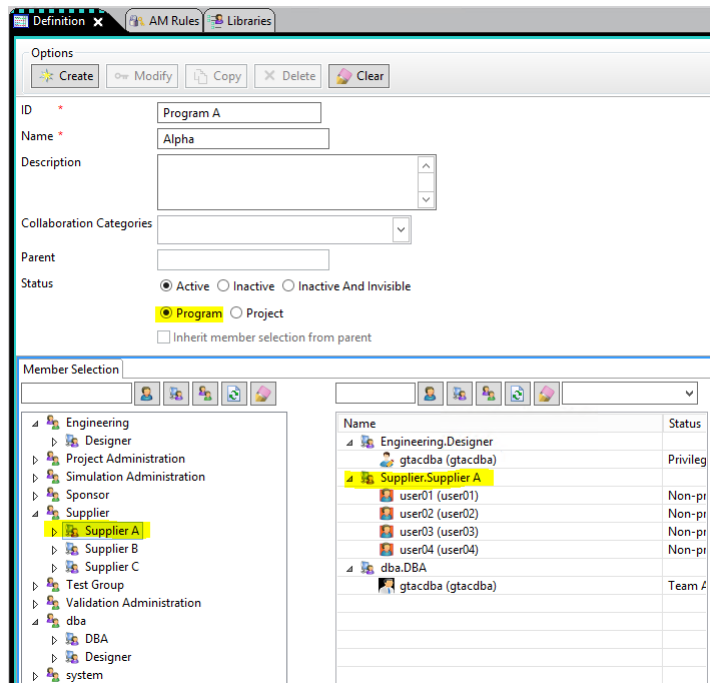
## Control data access in program and child projects

You can create hierarchical projects, which can be configured to inherit access from the parent program. In this example, the goal is to grant access to a sample user, **user01**, if an item is assigned to **Project B** and deny access to other users within the same group. We will create a **Program A** and a **Project B** to demonstrate the scenario for granting access to specific users in a project.

Typically, read access is denied from World and users are added to a *project team*, which defines read access based on the project assignment.

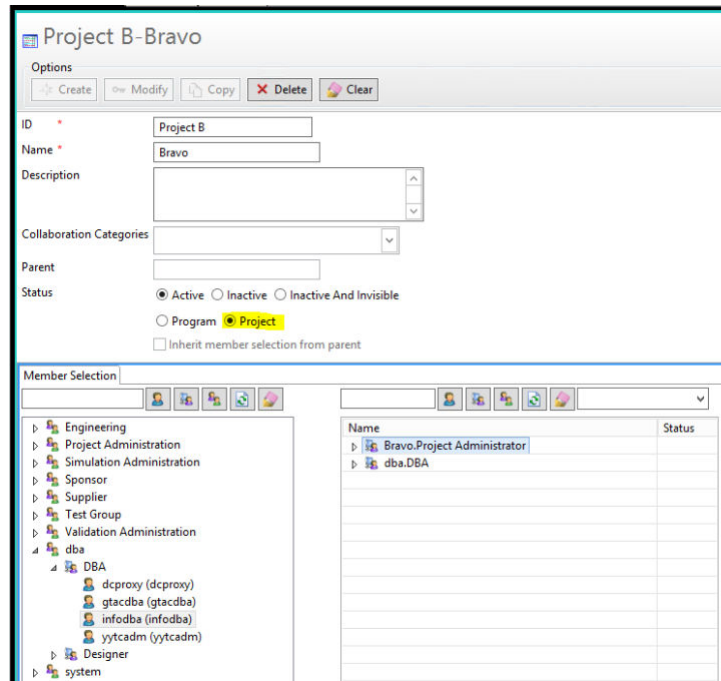
The following steps are recommended when using hierarchical projects.

1. Instead of controlling access to all members of a specific group (**Supplier A**) within the Access Manager (AM) rule tree, Siemens Digital Industries Software recommends you add **Supplier A** to a program within the Project module.
  - a. At the program level, create a new program, **Program A**:
    - Define **ID** as **Program A**.
    - Define **Name** as **Alpha**.
    - Select the **Program** button.
    - From **Member Selection**, select **Supplier A** and add to the members.



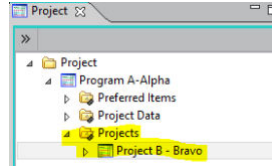
## 2. At the project level, create a new project, **Project B**:

- Define **ID** as **Project B**.
- Define **Name** as **Bravo**.
- Select the **Project** button.
- From **Member Selection**, select **dba** group and add to the members. By default, you are added as **Bravo.Project Administrator**.

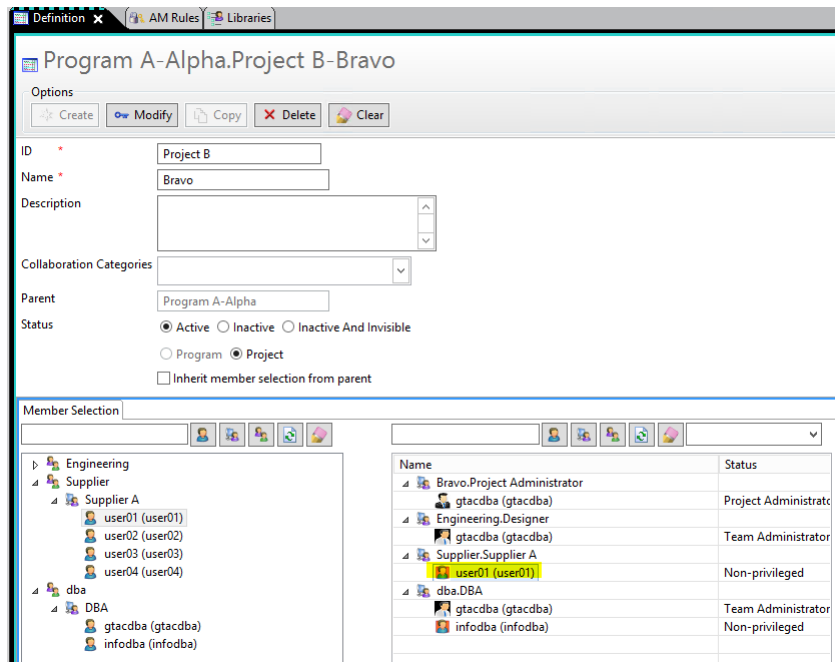


- Cut and paste **Project B** into the **Projects** folder within **Program A**.

This makes **Program A-Alpha** the parent.



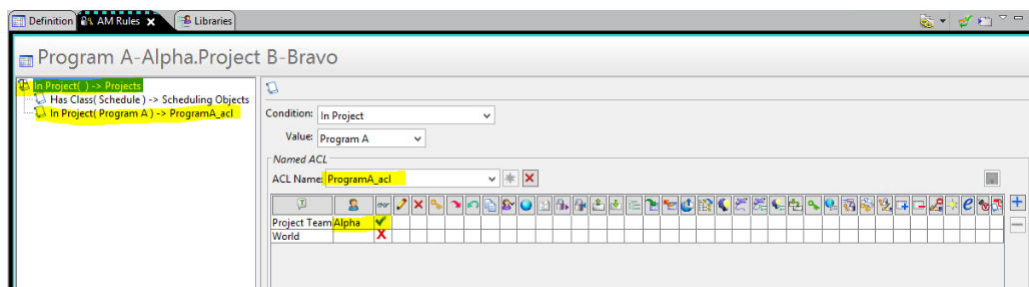
- Since **Program A-Alpha** is the parent, the left side of the **Member Selection** tab only shows members from **Program A-Alpha**. Add **user01** from **Supplier A** to **Project B**.



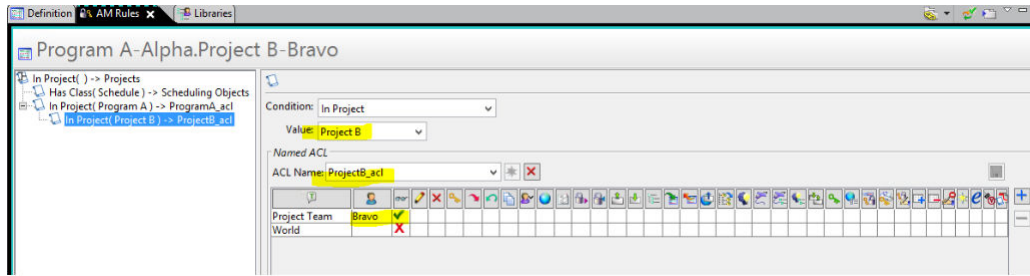
- Before creating new project-specific access control lists (ACLs), create a backup copy of your current AM rule tree from within Access Manager by performing a **File** → **Export** to save your current configuration.
- Review your Program and Project branches of your AM rule tree by using the **AM Rules** tab. Note that read access is denied to the **World** and granted to **Project Teams**.

Within the **In Project** branch, add a rule and a new ACL for **Program A**:

- Add accessor **Project Team** equals **Alpha** and grant **Read** access.
- Add accessor **World** and deny **Read** access.



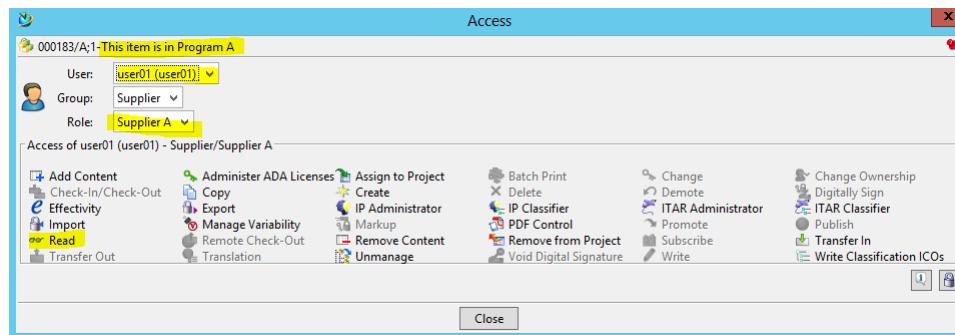
- Add a child rule for **Project B**.
  - Add accessor **Project Team** equals **Bravo** and grant **Read** access.
  - Add accessor **World** and deny **Read** access.



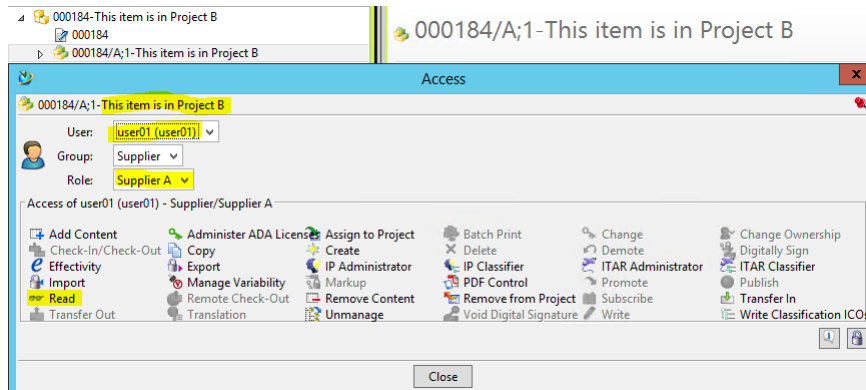
- With these settings, only **user01** from the **Supplier** group has read access to **Project B**, since **user01** was added to the project team. All members within the **Supplier A** group have access to **Program A**.

Note that **user01** has access to items assigned to *both* **Program A** and **Project B**:

- **Program A**

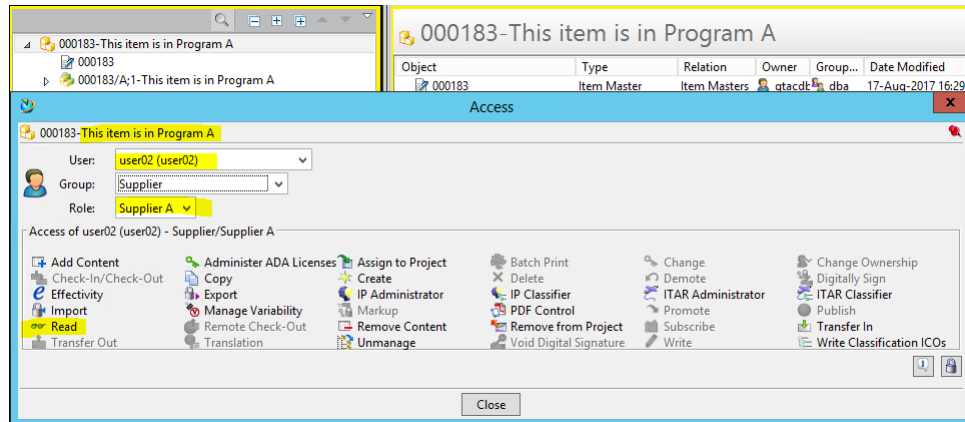


- **Project B**

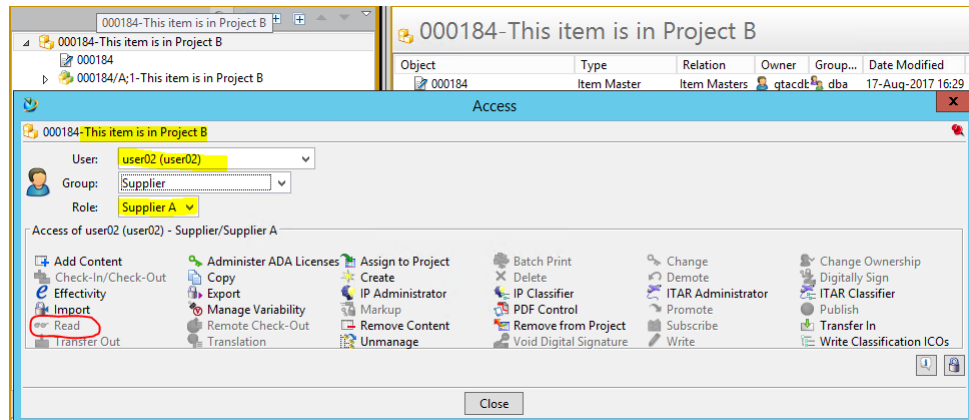


Note that **user02** has access to items assigned to **Program A**, but does *not* have access to items in **Project B**:

- **Program A**



- **Project B**



## Control data access based on a project category

*Project categories* allow you to control access to objects in a project without using access control rules. You can restrict a user from objects based on the category of the project, for example, a user assigned to the **Supplier** project category would be unable to view proprietary information. You can set different permissions for each category of a project. The default project categories are: **Internal**, **Partner**, and **Supplier**. Project categories are defined in the **FndOProjectCategories** list of values (LOV) found in the Business Modeler IDE.

1. In the **Definition** pane, select an existing project from the **Project** tree.
2. Type a new project or program ID in the **ID** box. The maximum length for a project or program ID is 64 characters.
3. Type a new project in the **Name** box. The maximum length for a project or program ID is 32 characters.
4. (Optional) Type a description of the project or program in the **Description** box.

5. Select a category from the **Project Category** drop-down list.

Project categories allow you to control access to objects in a project without using access control rules. The default categories are: **Internal**, **Partner**, and **Supplier**.

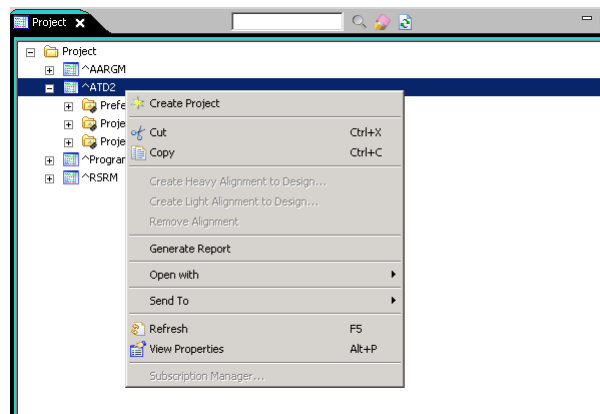
If you want to **change the listed project categories**, you can change the values in the LOV. Project categories are defined in the **FndOProjectCategories** list of values (LOV) found in the Business Modeler IDE.

6. Select the **Project** button.
7. Assign users to the project.
8. Click **Create**.

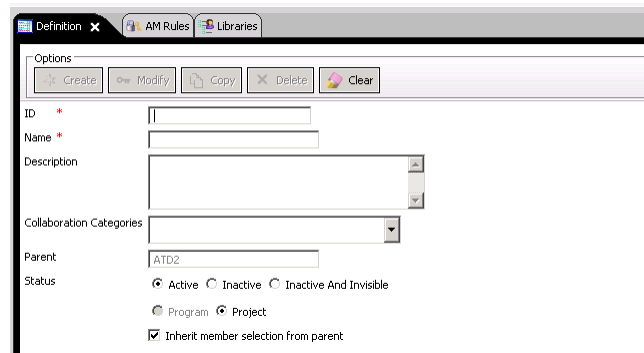
## Build hierarchy by adding new child projects under a program

You can create a new project under a program.

1. In the **Project** tree, select the program you want to create a project under.
2. Right-click the program and choose **Create Project**.



3. Select the **Inherit member selection from parent** check box to inherit the members from the program to the project.

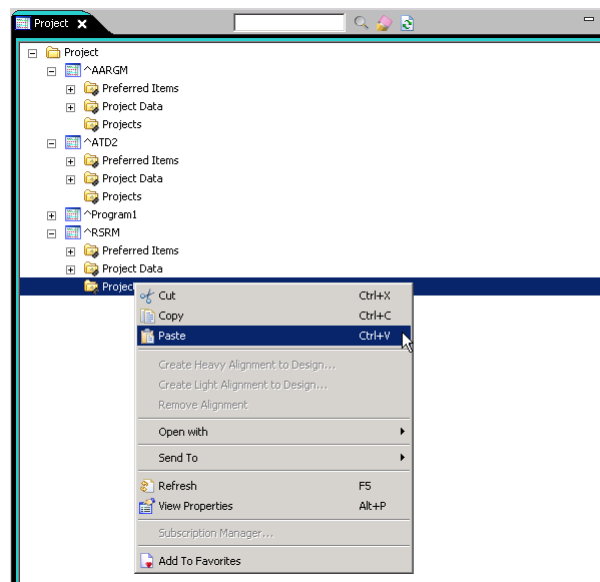


The **Parent** box automatically displays the project name, and **Project** is selected.

## Build hierarchy by adding existing projects to a new program

You can add an existing project to a new program by creating the program and using **Copy** and **Paste**.

1. Create a new program.
2. Select existing project.
3. Select the project to **Copy**.
4. Select the project folder under the program you want to add it to.
5. Right-click the project folder and choose **Paste**.



The project appears under the program.

When an existing project is pasted under a new program the project members' status changes:

- Existing project administrators on child projects are changed to project team administrators. The parent program administrators are added to child project as project administrators.
- Existing project team administrators of the child project will remain as the project team administrators.
- Privileged members of the child project are added to the parent program as privileged members.
- Team administrators of the child project are added to the parent program as privileged members.
- Regular members of the child project are added to the parent program as regular project members.

### Inherit team members from a parent program

Team members can be inherited from a parent program using the **Project Administration** window.

If the **Inherit member selection from parent** check box is selected:

- Program team members are added to the project.
- The **Create** button is enabled.
- The **Add member** button is disabled.

If **Inherit member selection from parent** is cleared:

- You must create the project team manually.
- The **Create** button is disabled until a member is added to the project.

The **Inherit member selection from parent** check box is only enabled on projects that have a parent.

### Modify projects and programs

1. Select an existing project or program from the **Project** or **Program** tree.
2. Modify project or program settings by:
  - Adding or removing team members.
  - Assigning a new team administrator.

- Assigning or removing privileged team member status.
  - Changing access rules.
3. Click **Modify**.

Modifications to the project or program information are saved to the database.

Siemens Digital Industries Software does not recommend changing the project or program name or ID after data has been assigned to an active project or program.

## Set project status

You must be a team member in the **Project Administrator** role in the **Project Administration** group to set the status of a project or program.

1. Expand the **Project** or **Program** tree and select the project or program.
2. Click the **Definition** tab.

Tip:

If the **Definition** tab is not displayed, click the **Project Administration** quick link.

3. Select one of the following status options.

Option	Description
<b>Active</b>	Project or program data is visible and can be modified by project or program team members.
<b>Inactive</b>	Project or program data is not visible to project or program team members You can add rules to make inactive program(s) data visible to program team members, but the data cannot be modified. You cannot add similar rules for inactive projects.
<b>Inactive and Invisible</b>	Project or program data is invisible and cannot be modified. Only project administrators can view the data owned by an inactive and invisible project or program.

4. Click **Modify** for an existing project or program, or click **Create** for a new project or program.

## Delete a project or program

1. Expand the **Project** or **Program** tree, and select the project or program that you want to delete.

2. Click the **Definition** tab.

Tip:

If the **Definition** tab is not displayed, click the **Project Administration** quick link.

3. Click **Delete**.

The **Confirm Delete** dialog box appears.

You can delete a project or program that has data assigned to it; however, you cannot delete a project or program that **owns data**.

4. Click **Yes**.

## Associating data dictionaries with projects

### What are data dictionaries?

Data dictionaries are central repositories for key building blocks or components of designs used in typical functional and logical design activities. These repositories make it easy to reuse components in multiple designs across projects or programs. Data dictionaries are modeled as libraries in the Classification application and can be associated with projects or programs, which allows designers who are project members to add data from the dictionary to function, logical, or physical model structures.

Tip:

Multiple data dictionaries (libraries) can be associated with a single project or program. In addition, a single data dictionary can be associated with multiple projects or programs.

### Add a data dictionary to a project or program

Data dictionaries are modeled as libraries using Classification classes. In this procedure, data dictionaries are referred to as *libraries*.

1. In the **Project Administration** window, select a project in the **Project** tree.
2. Click the **Libraries** tab.
3. In the **Hierarchy** pane, expand the tree until you locate the library that you want to associate with the selected project.
4. Select the library in the hierarchy, add it to the pane on the right side of the window, and click the **Save** button located between the panes.

The library (data dictionary) is now associated with the project. Designers who are project members can add data from the dictionary to model structures while working in Systems Engineering or Structure Manager.

## Assigning users to projects


### Assign individual users to a project or program



1. In the **Member selection** area of the **Definition** pane, locate the users in the **Organization** tree.


**Note:**

You cannot assign an inactive user to a project. However, if you make a user inactive after adding the user to a project, then the user remains assigned to the project unless removed from the project team.

**Tip:**

You can search for users by entering the user name (or partial name and wildcard characters) in the box and clicking the **Find users** button .

If your search yields no results, the tree pane will be empty. Click the **Refresh** button  or the **Clear** button  to restore the display.

2. Select the user or users and click the **Add** button .
3. After team members are assigned to the project or program, you can:

- Designate a **team administrator**.

**Note:**

Only project administrators can designate a project or program team administrator.

- Designate **privileged team members**.
4. Click the **Create** button if you've added users to a new project, or click the **Modify** button if you've added users to an existing project.

### Assigning users to a project using a workflow

You can assign users to a project using a workflow by:

- Using the **PROJ-assign-members** workflow action handler.

Using the **PROJ-assign-members** handler, either the project administrator or the project team administrator of each project receiving new members can add members to projects using handler arguments only. For example:

- The projects to receive members are named **Proj1** and **Proj2**.
- The user named **john** is to be added to both projects as a *nonprivileged member*. This user has the **Designer** role in the **Engineering** group.
- The user named **jane** is to be added to both projects as a *privileged member*. This user has the **Manager** role in the **Engineering** group.

Argument	Values
-projects	Proj1,Proj2
-members	Engineering/Designer/john
-privileged_members	Engineering/Manager/jane

- Creating a custom form using the Business Modeler IDE.

Using the Business Modeler IDE, create a custom form (for example, **Assign Project Member Form**) that can be used to assign members to a project.

The workflow administrator uses the **PROJ-assign-members** workflow handler to specify the projects and members using the properties on the custom form attached to the workflow template.

## Assign groups of users to a project or program

Hierarchical group behavior applies to the assignment of users to projects and programs. If you select a group that has subgroups, all users in the parent group and subgroups are assigned membership in the project or program.


### Note:



If a member is added to that group in the Organization application, the same user becomes a member of a project. In turn, if a member is removed from that group in the Organization application, the same user is removed from a project.


If a group is added to a project, all members in the group will be added to the project. Users in a group cannot be removed from the project/program. If a group member is explicitly added to the project, they can be removed from the project. If a role is added to the project, then all group members under the role are explicitly added to the project. Group members in a role can be removed from the project.

1. In the **Member Selection** area of the **Definition** pane, locate the group in the **Organization** tree.

**Tip:**

You can search for the group by typing the group name (or partial name and wildcard characters) in the box and clicking the **Find groups** button .

If your search yields no results, the tree pane will be empty. Click the **Refresh** button  or the **Clear** button  to restore the display.

2. Select the group or subgroup and click the **Add** button .

The group appears in the table of selected members at the right of the **Member Selection** area.

3. After team members are assigned to the project or program, you can:

- Designate a **team administrator**.

**Note:**

Only project administrators can designate a project or program team administrator.

- Designate **privileged team members**.

4. Click the **Create** button if you've added users to a new project, or click the **Modify** button if you've added users to an existing project.


## Assign all users in a role to a project

You can assign all users in a specific role to a project or program.



If a role is added to a project, you are explicitly adding those members in that role at the time of the addition.


1. In the **Member Selection** area of the **Definition** pane, locate the role in the **Organization** tree.

**Tip:**

You can search for the role by typing the role name (or partial name and wildcard characters) in the box and clicking the **Find roles** button .

Teamcenter displays all roles matching the search criteria. The roles are displayed in context of the groups to which they belong.

If your search yields no results, the tree pane will be empty. Click the **Refresh** button  or the **Clear** button  to restore the display.

2. Select the role and click the **Add** button .

The role appears in the table of selected members at the right of the **Member Selection** pane.

All users in the role are added as members of the project or program.

3. After team members are assigned to the project or program, you can:

- Designate a **team administrator**.

Note:

Only project administrators can designate a project or program team administrator.

- Designate **privileged team members**.

4. Click the **Create** button if you've added users to a new project, or click the **Modify** button if you've added users to an existing project.

## Select team administrators

Team administrators manage resources for projects and programs by assigning or removing users as resource needs change. Depending on the size of your projects/programs and the needs of your organization, you may choose to designate a single team administrator or to designate multiple team administrators.



Designating multiple team administrators allows you to spread the burden of administrative tasks, which can be considerable for large projects. For small projects you may also want to designate multiple team administrators to ensure coverage in the event that your primary team administrator is unavailable.

Note:

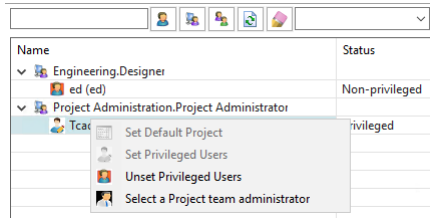
If you are logged on as the Project Team Administrator for the selected project, you are unable to change your status. This prevents unintentional removal of your Project Team Administrator status. Your Project Team Administrator status can only be changed by another Project Team Administrator.

1. Select a project from the tree in the **Project** view.
2. Go to the **Definition** view and locate the team members (in the project member tree in the lower-right of the view) who will be team administrators.


You can search for users, groups, or roles within the project member tree.

If your search yields no results, the tree pane will be empty. Click the **Refresh** button  or the **Clear** button  to restore the display.

3. Select the team members, right click, and choose **Select a Project team administrator**.



You can select individual users from different groups and roles or you can select all users in a group or all users in a specific role.



The project member tree is updated, the user icon  is displayed next to the user’s name, and **Team Administrator** is displayed in the **Status** column of the tree table.

### Designate privileged team members

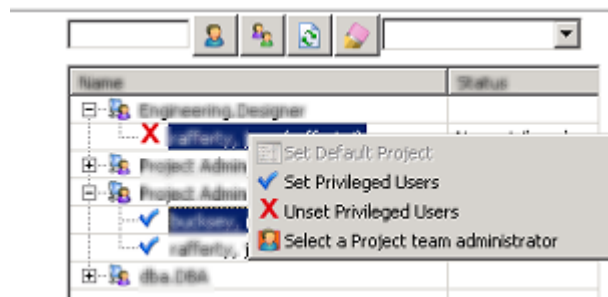
Designating team members as *privileged* is one step in the process of **granting access** to users to allow them to assign data to and remove data from projects and programs.

1. Select a project from the tree in the **Project** view.
2. Go to the **Definition** view and locate the team members (in the project member tree in the lower-right of the view) who will be designated as privileged users.


You can search for users, groups, or roles within the project member tree.

If your search yields no results, the tree pane will be empty. Click the **Refresh** button  or the **Clear** button  to restore the display.

3. Select the team members, right click, and choose **Set Privileged Users**.



You can select individual users from different groups and roles or you can select all users in a group or all users in a specific role.


The project member tree is updated, the check mark  icon is displayed next to the user's name, and **Privileged** is displayed in the **Status** column of the tree table.

## Applying security rules to projects and programs

### Overview of applying project and program security (Access Manager) rules

Project administrators with **DBA** privileges can extend the default security rules, which grant read access to project or program data to members of the project or program team, on a project-by-project or program-by-program basis.

Note:

Project administrators with **DBA** privileges only have access to the  **In Project() -> Projects** branch of the rule tree.

Using the Project branch in the rule tree, you can:

- Grant or deny access to a particular group of users by applying the **Owning Group** condition.
- Grant or deny access to groups of users based on the group's categorization as internal (OEM) or external (supplier) by applying the **Owning Group Has Security** condition.
- Grant access to data assigned to projects by applying the **In Project** condition.

Note:

This rule is applied by default to any object assigned to an active project.

- Grant or deny access to users based on their membership in a project by applying the **Is Project Member** condition.
- Grant or deny access to users based on their membership in a program by applying the **Is Program Member** condition.
- Deny users access to data if the owning program is not the active program in the user's session by applying the **In Current Program** condition.
- Deny users access to data if the owning program is inactive by applying the **In Inactive Program** condition.
- Deny users access to data if the owning program is invisible by applying the **In Invisible Program** condition.

- Grant or deny access to program data by applying the **Is Owned By Program** condition.


## Default security rules for projects and programs

### Access rules for projects and programs

The Access Manager rule tree delivered as part of the standard Teamcenter installation includes the following rules related to programs and projects:

- **Is Project Member(true) -> Project Objects**
- **In Project() -> Projects**
- **Is Program Member(true) -> Not Program Member**
- **In Current Program(false) -> Not Current Program**
- **In Inactive Program(true) -> Inactive Program**
- **In Invisible Program(true) -> Invisible Program**
- **Is Owned By Program() -> Projects**

### Is Project Member rule


























The  **Is Project Member(true) -> Project Objects** rule specifies whether the user's membership in the project is evaluated. This condition is true only when the user is a current member of the project.

Note:

The **Is Project Member(true) -> Project Objects** rule can only be modified by an administrator using the Access Manager application. It cannot be modified from the Project application.

The **Project Objects** ACL grants project administrators and project team administrators privileges to modify *projects in which they are members*. These privileges apply to the project metadata, not to the data assigned to projects.


The ACL is defined as follows.

						
Accessor Type	Accessor ID	Read	Write	Delete	Change	Change Ownership
Owning User						
Regular project member						
Administrator project member						
Team Admin project member						
Privileged project member						
World						














**Note:**

You can modify the **Project Objects** ACL to meet the project access requirements at your site.

## In Project rule

The  **In Project()** → **Projects** rule grants access to data assigned to projects. This default rule is applied to any object that is assigned to an active project.

The **Projects** ACL grants read privileges to the data in a project, as follows.

											
		Read									
Project Teams											


The **Project Teams** accessor gives all team members read privileges to the data in a project. For example, if the **Design**, **Validation**, and **Documentation** groups are selected as a project team, the **Project Teams** accessor grants privileges to all members of each group; therefore it is not necessary to use the **Group** accessor to grant privileges to each group individually.

The project administrator can create or modify project security rules to meet the requirements of a specific project by creating a new named ACL for the project or by adding rules under the **In Project** condition in the rule tree. Project administrators can modify rules using Project.














Note:

Only an administrator with privileges to use Access Manager can change the placement of the **In Project** rule in the AM rule tree. They can also modify the order of the child nodes of the **In Project** branch of the rule tree.

## Is Program Member rule

The  **Is Program Member(false) -> Not Program Member** rule denies read access to users if the user is not a member of the owning program or shared program.

The **Not Program Member** ACL denies read access to the data in the program, as follows.


		 Read									
World											

The **World** accessor denies read access to users if the user is not the member of the owning program or the shared programs.

Note:

This rule evaluates to true when the object is owned by a program and the session user is member of the object owning program. Program is basically a project with “program level security” turned on. When the **autoAssignToProject** extension is configured and a program is set in the session, then any object created in that session is owned by that session program. This makes the **Is Program Member** condition apply to the object. It evaluates to true if a given session user is a member of the program.

## In Current Program rule


The  **In Current Program(false) -> Not Current Program** rule denies write, delete, change, and export privileges to users if the owning program of the data is not the active program for the user's session.

The **Not Current Program** ACL denies the privileges to the data in a program, as follows.












		 Read				
World						

The **World** accessor denies write, delete, change and export privileges to users if the owning program of the data is not the active program for the user's session.

### In Inactive Program rule


The  **In Inactive Program (true)** → **Inactive Program** rule denies write, delete, change and export privileges to users if the owning program of the data is in the inactive state.

The **Inactive Program** ACL denies write, delete, change and export privileges to the data in the program, as follows.





		 Read				
World						

The **World** accessor denies write, delete, change and export privileges to users if the owning program of the data is in the inactive state.

### In Invisible Program rule


The  **Is Invisible Program (true)** → **Invisible Program** rule denies read access to users if the owning program of the data is in the invisible state.

The **Invisible Program** ACL denies read access to the data in the program, as follows.














		 Read									
World											

The **World** accessor denies read access to program data if the program is in the invisible state.

### Is Owned by Program rule

The  **Is Owned By Program()** → **Projects** rule grants or denies access to data based on program or project ownership.

The **Projects** ACL grants read access to the data in the program, as follows.

											
		Read									
Project Teams											

The **Is Owned By Program** rule can be configured to Enable the exchange of Aerospace and Defense program data between databases. Exchanging program data requires that the user initiating the import or export is a member of the program to which the objects being exchanged are assigned.

## Configure evaluation of rules that include the Role in Project and Role in Projects of Object accessors

You can set the **AM\_PROJECT\_MODE** preference to configure how the system considers the user's roles in active projects when evaluating access rules that use the **Role in Project** or **Role in Projects of Object** accessors.

The system either:

- Evaluates the user's roles in the context of all active projects.
- Evaluates the user's roles in the context of the current project. The user's current project is the project specified for the active Teamcenter session.

Note:

To view a user's current project, choose **Edit→User Setting**.

### Role in project of object for roles under the sub-group in projects

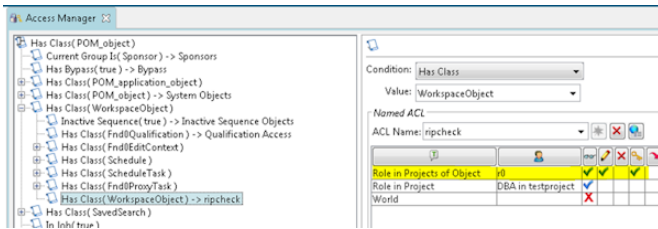
This preference enables the AM evaluation of **Role in Project of Object** accessors to be effective on the **Roles** that are in the sub-group level in **Projects**.

The prerequisite is the corresponding **Role In Project** accessor object exists (In order to have the **Role In Project** accessor object, at least there has to have either a **Group** that contains such **Role** added to the **Project Team** directly, or a **Group Member** with such **Role** directly added to the **Project Team**).

To enable this feature turn the **AM\_enable\_RIP\_for\_child\_groups** preference to **True**.

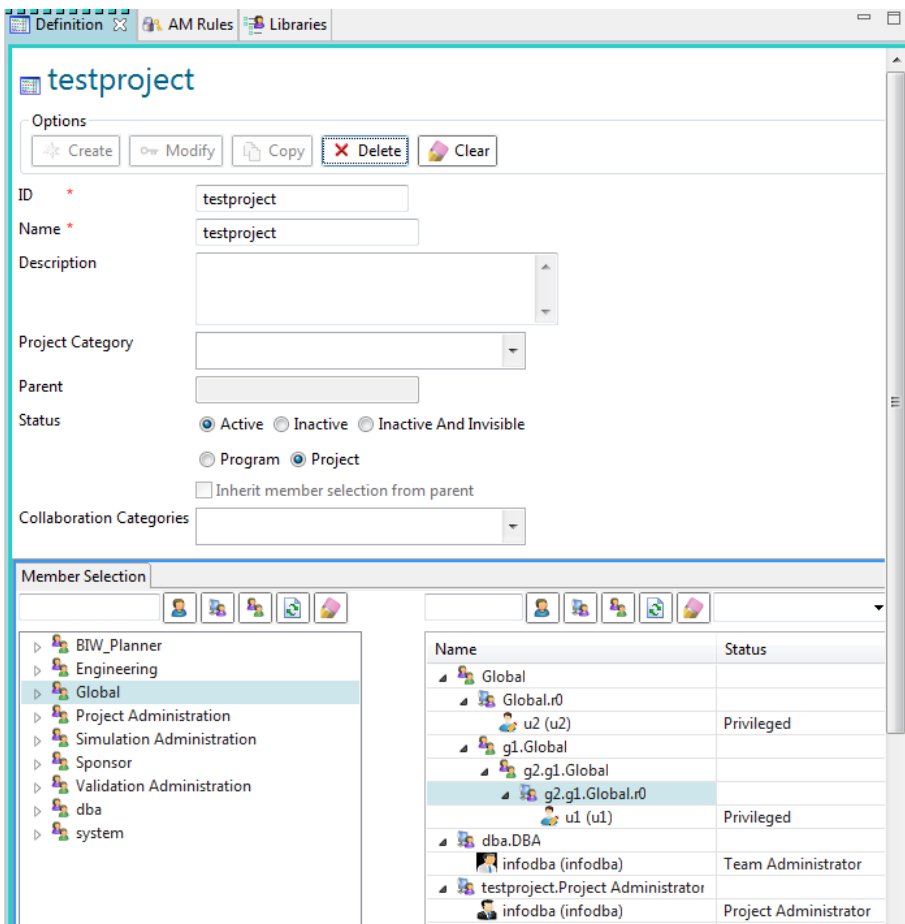
Example

The AM rule definition



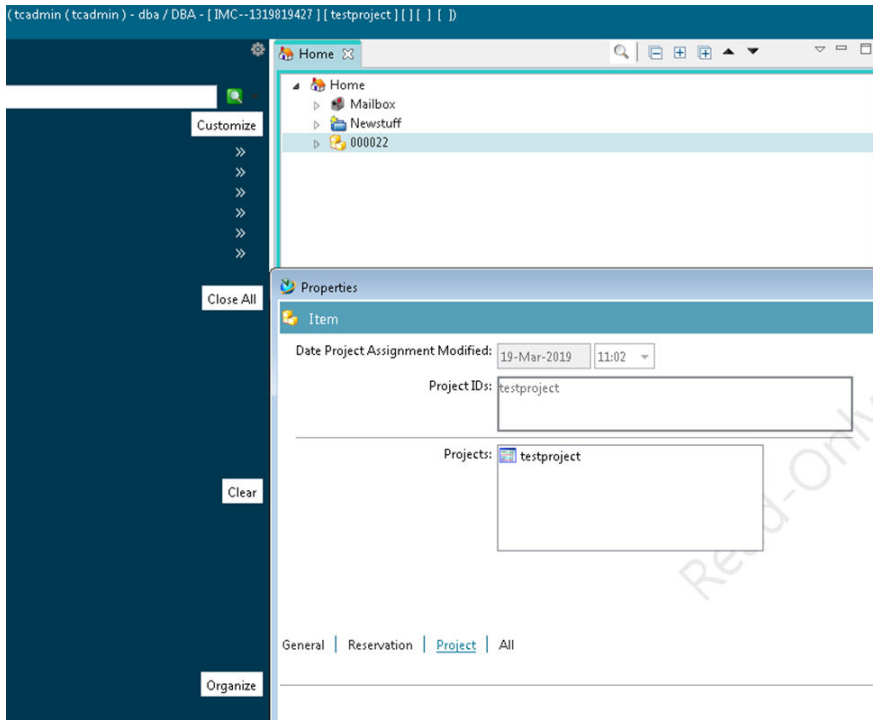
## Project team structure

It is required to have the **Role In Project** accessor object created for "r0" in "testproject". The object is created either by adding the **Group** which contains the **Role** directly to the **Project Team**, or by adding any **Group Member** which has the **Role** to the **Project Team**. In this case, the **Group** "global" contains the role "r0", which is directly added to the **Project Team**.

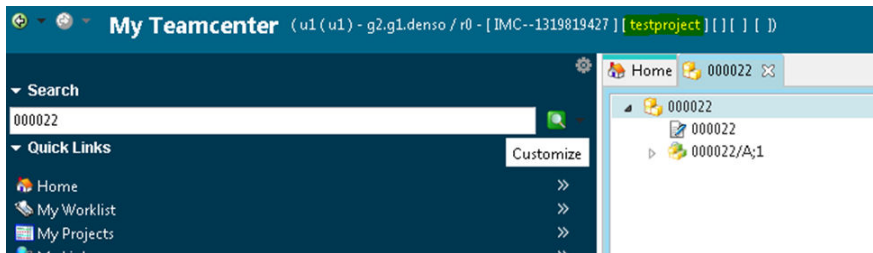


We are expecting the "u1" to be able to access the data that's assigned to the **Project** "testproject".

Item 000022 has been created by the **tcadmin** user and assigned to "testproject".



To confirm access, log on as **User "u1"** and search "000022".



### Note:

The session **Project** of "u1" needs to be set as "testproject" (or any **Project** that's assigned to the object the **User** is expecting to have access to).

The access is granted based on the AM rules defined for **Role in Projects of Object "r0"**.

Privilege	Verdict	Named ACL	Accessor	Rule Path
Read	✓	ripcheck	Role in Projects of Object	Has Class(WorkspaceObject) / Has Class(WorkspaceObject) / Has Class(POM_object)
Write	✓	ripcheck	Role in Projects of Object	Has Class(WorkspaceObject) / Has Class(WorkspaceObject) / Has Class(POM_object)
Delete	✗	Working	World	Has Class(POM_application_object) / Has Class(POM_object)
Change	✗	ripcheck	Role in Projects of Object	Has Class(WorkspaceObject) / Has Class(WorkspaceObject) / Has Class(POM_object)
Promote	✗	Working	World	Has Class(POM_application_object) / Has Class(POM_object)
Demote	✗	Working	World	Has Class(POM_application_object) / Has Class(POM_object)
Copy	✓	Working	World	Has Class(POM_application_object) / Has Class(POM_object)
Change Ownership	✗	Working	World	Has Class(POM_application_object) / Has Class(POM_object)
Publish	✗	Working	World	Has Class(POM_application_object) / Has Class(POM_object)
Subscribe	✗	Working	World	Has Class(POM_application_object) / Has Class(POM_object)
Export	✓	Import/Export	World	Has Class(POM_application_object) / Has Class(POM_object)
Import	✓	Import/Export	World	Has Class(POM_application_object) / Has Class(POM_object)
Transfer Out	✗	Import/Export	World	Has Class(POM_application_object) / Has Class(POM_object)
Transfer In	✓	Import/Export	World	Has Class(POM_application_object) / Has Class(POM_object)
Write Classification ICOs	✓			A.M.(NO RULES APPLY)
Assign to Project	✓			A.M.(NO RULES APPLY)
Remove from Project	✓			A.M.(NO RULES APPLY)
Remote Check-Out	✗	Working	World	Has Class(POM_application_object) / Has Class(POM_object)
Unmanage	✓			A.M.(NO RULES APPLY)
IP Administrator	✓			A.M.(NO RULES APPLY)
ITAR Administrator	✓			A.M.(NO RULES APPLY)
ITAR Classifier	✓			A.M.(NO RULES APPLY)
IP Classifier	✓			A.M.(NO RULES APPLY)
Check-In/Check-Out	✗	Working	World	Has Class(POM_application_object) / Has Class(POM_object)

## Configure evaluation of the ASSIGN\_TO\_PROJECT and REMOVE\_FROM\_PROJECT privileges

You can set the **TC\_project\_validate\_conditions** preference to specify how the **ASSIGN\_TO\_PROJECT** and **REMOVE\_FROM\_PROJECT** access privileges are validated in conjunction with privileged project membership.

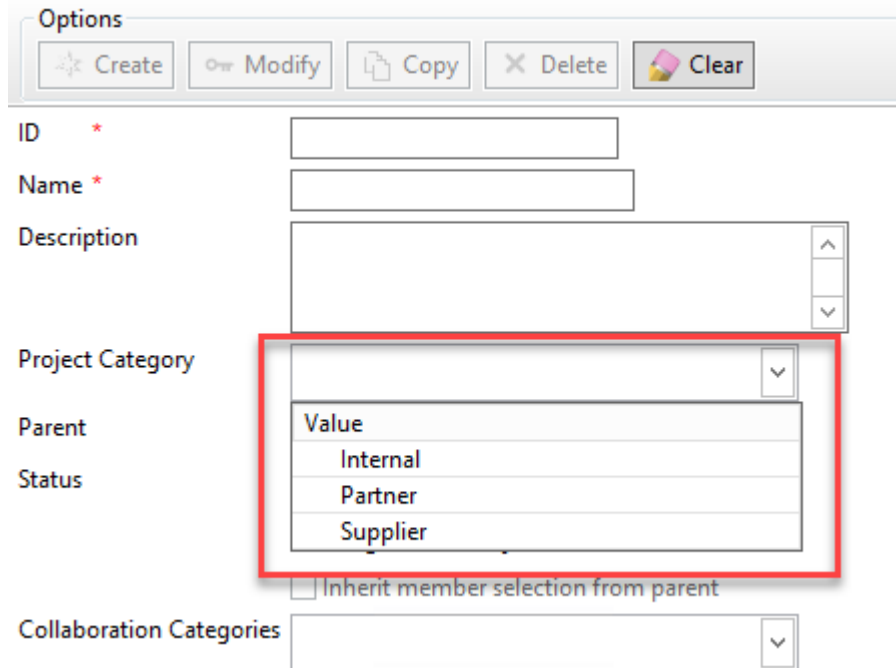
You can configure the following project access scenarios:

- Users who are privileged project members can assign objects to or remove objects from projects.
- Users who are either privileged project team members or who have specific **ASSIGN\_TO\_PROJECT** and **REMOVE\_FROM\_PROJECT** privileges can assign objects to and remove objects from projects.
- Users who are members of the project, but who are not privileged team members, must have **ASSIGN\_TO\_PROJECT** and **REMOVE\_FROM\_PROJECT** privileges to assign objects to and remove objects from projects.
- Users who are privileged team members and also have **ASSIGN\_TO\_PROJECT** and **REMOVE\_FROM\_PROJECT** privileges can assign objects to or remove objects from projects.
- This preference is not included in your Teamcenter installation. You must add it to the database.
- The **TC\_project\_validate\_conditions** preference does not apply to the assignment of replicated objects during an import operation. The **TC\_multi\_site\_project\_member\_bypass** controls the privileges required to assign replicated objects to projects upon import.
- The **TC\_project\_validate\_conditions** preference has no effect on which projects are displayed in the Teamcenter interface. Therefore, it is possible that users will see projects displayed in the interface, for example in the **User Settings** dialog box, that they do not have access to.

## Add project categories

*Project categories* allow you to control access to objects in a project without using access control rules. Objects can be added to a category and then the administrator can define a rule for that specific category. Changing the category of a project is a small and easy change that affects all project objects without changing the AM rule tree.

1. To see the available default categories in the Project application, click the arrow in the **Project Category** box.



The screenshot shows the 'Options' dialog box with the following fields and controls:

- Options** header with buttons: Create, Modify, Copy, Delete, Clear.
- ID \***: Text input field.
- Name \***: Text input field.
- Description**: Text area with up/down arrows.
- Project Category**: Dropdown menu (highlighted with a red box) showing a list of values: Internal, Partner, Supplier.
- Parent**: Text input field.
- Status**: Text input field.
- Inherit member selection from parent
- Collaboration Categories**: Dropdown menu.

**Note:**

These categories are defined in the **Fnd0ProjectCategories** list of values (LOV) found in the Business Modeler IDE. If you want to change the listed categories, you can change the values in the LOV.

2. In the Business Modeler IDE, open the **Extensions\LOV** folders and double-click the **ProjectCategories** list of values.

Fnd0ProjectCategories x TC\_Project

LOV : Fnd0ProjectCategories

**Details**

Project: foundation

Name: Fnd0ProjectCategories

Description\*: The list of category values to categorize projects. Using this LOV, user can configure user-defined category values based on business requirement.

Type: ListOfValuesString

Usage:  Exhaustive  Suggestive  Range

LOV Value Management

Enter values using BMIDE and store values in my template

Supply values directly to Teamcenter database using "bmide\_manage\_batch\_lovs" command line utility

Reference:

Show Cascading View

Value	Description	Condition	COTS	Template
<input checked="" type="radio"/> Internal		<input checked="" type="checkbox"/> isTrue		foundation
<input checked="" type="radio"/> Partner		<input checked="" type="checkbox"/> isTrue		foundation
<input checked="" type="radio"/> Supplier		<input checked="" type="checkbox"/> isTrue		foundation

- To add a new category, click the **Add** button to the right of the table and type the new value. (You can also remove the default values and add completely new values, if desired.)

The screenshot shows a dialog box titled "Add LOV Value". At the top, there is a "Create" section with a red error icon and the message: "Invalid 'Value' field. LOV value 'Internal' is duplicate." Below this, there are four input fields: "Value" (containing "Internal"), "Value Display Name" (containing "Internal"), "Description" (containing "Internal Description Here"), and "Condition" (containing "isTrue"). A "Browse..." button is located to the right of the "Condition" field. At the bottom of the dialog, there are three buttons: "Finish", "Cancel", and "Apply".

4. Click **Finish**.

The new value is displayed on the table of values.

Fnd0ProjectCategories x TC\_Project

LOV : Fnd0ProjectCategories

**Details**

Project: foundation

Name: Fnd0ProjectCategories

Description\*: The list of category values to categorize projects. Using this LOV, user can configure user-defined category values based on business requirement.

Type: ListOfValuesString

Usage:  Exhaustive  Suggestive  Range

LOV Value Management

Enter values using BMIDE and store values in my template

Supply values directly to Teamcenter database using "bmide\_manage\_batch\_lovs" command line utility

Reference:

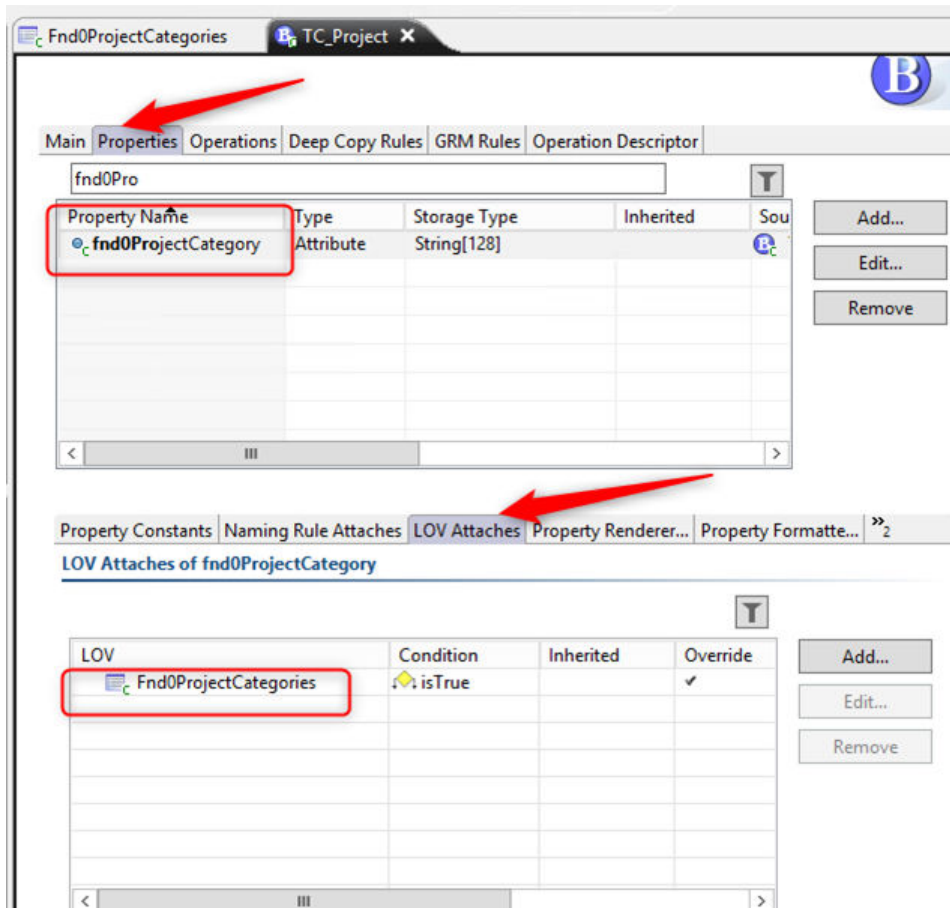
Show Cascading View

Value	Description	Condition	COTS	Template
<input checked="" type="radio"/> Internal		<input checked="" type="checkbox"/> isTrue		foundation
<input checked="" type="radio"/> Partner		<input checked="" type="checkbox"/> isTrue		foundation
<input checked="" type="radio"/> Supplier		<input checked="" type="checkbox"/> isTrue		foundation

5. The **Project Category** box in the Project interface is defined by the **fnd0ProjectCategory** property. To see how the categories list of values is attached to the **Project Category** box, open the **TC\_Project** business object, click the **Properties** tab, and scroll to the **fnd0ProjectCategory** property in the table. The **Fnd0ProjectCategories** list of values is shown as attached to this property in the **LOV Attaches** table.

Note:

If you want to create your own set of project categories in another list of values, you can attach your custom LOV to the **fnd0ProjectCategory** property and use it instead of the **Fnd0ProjectCategories** LOV.



6. To save your changes to the template, on the menu bar choose **BMIDE**→**Save Data Model**.

Then package the template and use Teamcenter Environment Manager to install the packaged template to your server.

7. To see the new category in the Project application, click the arrow in the **Project Category** box.

## Example of defining Access Manager rules to restrict access based on user and project category

The following example demonstrates how ABC Company configures access privileges that vary for a single item object based on the project category and the user in the current session. ABC Company configures one set of privileges for users who are listed in the Supplier category and configures another set of privileges for users who are listed in the Internal category.

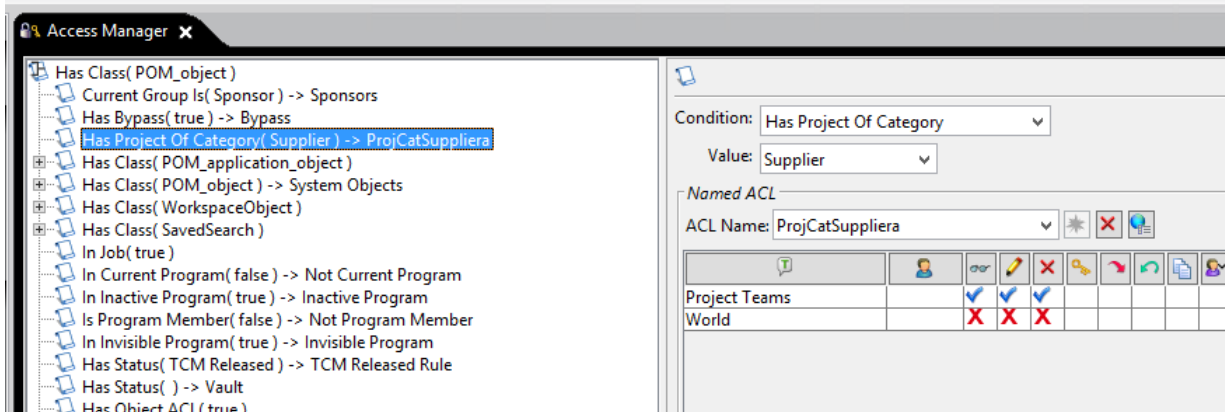
### Project category configuration in Business Modeler IDE

The administrator of ABC Company uses the Business Modeler IDE to set up the desired categories for the projects:

- Review the categories defined in the **Find** list of values (LOV).
- Add more categories, if needed.

### Rule configuration in Access Manager

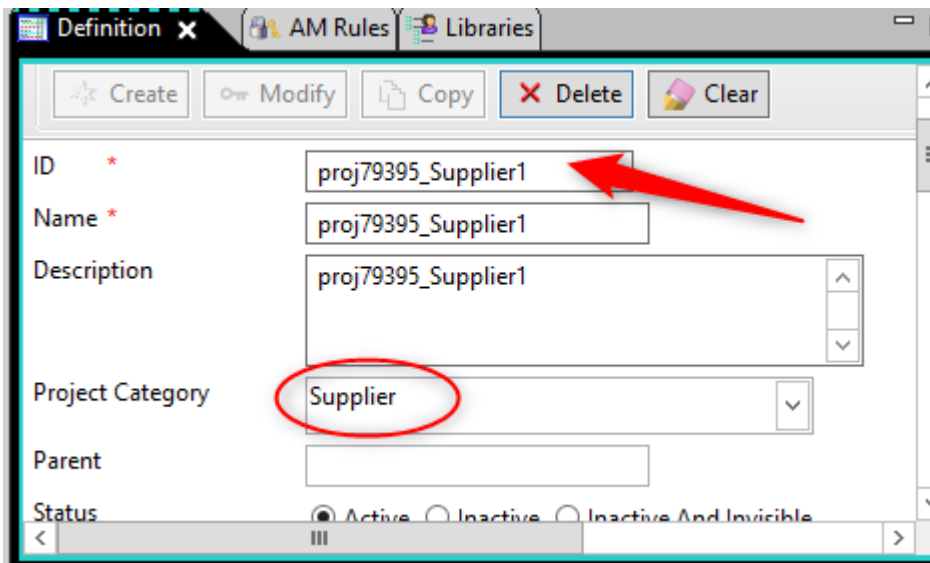
The administrator then uses the **Has Project Of Category** condition to create the following rules to check access. By calling out this condition, the administrator can search for this specific condition and see where to apply it.



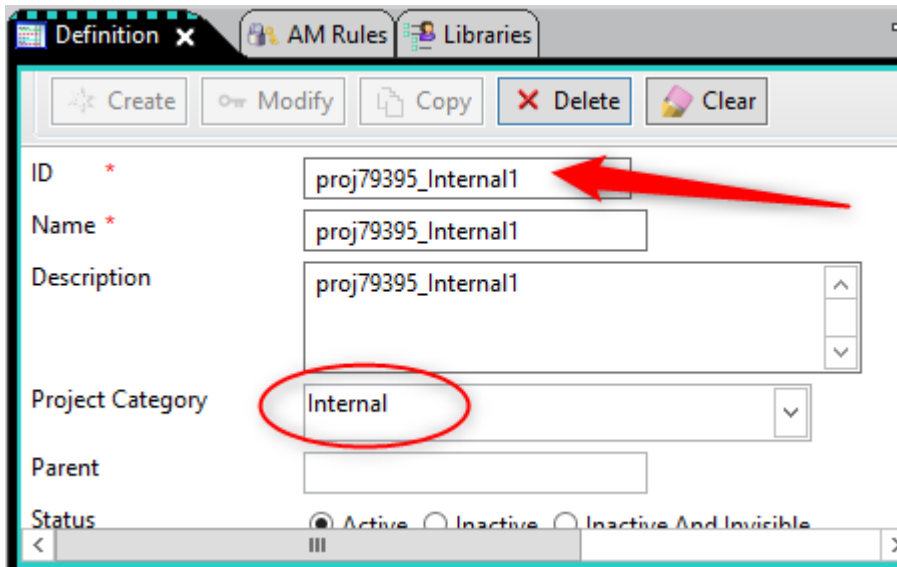
### Example of assigning categories to a Project

The following example shows how you can categorize a project under category and give access permission to that category.

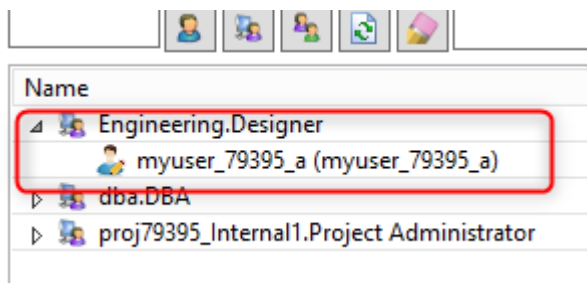
1. Create a project with category as Supplier.



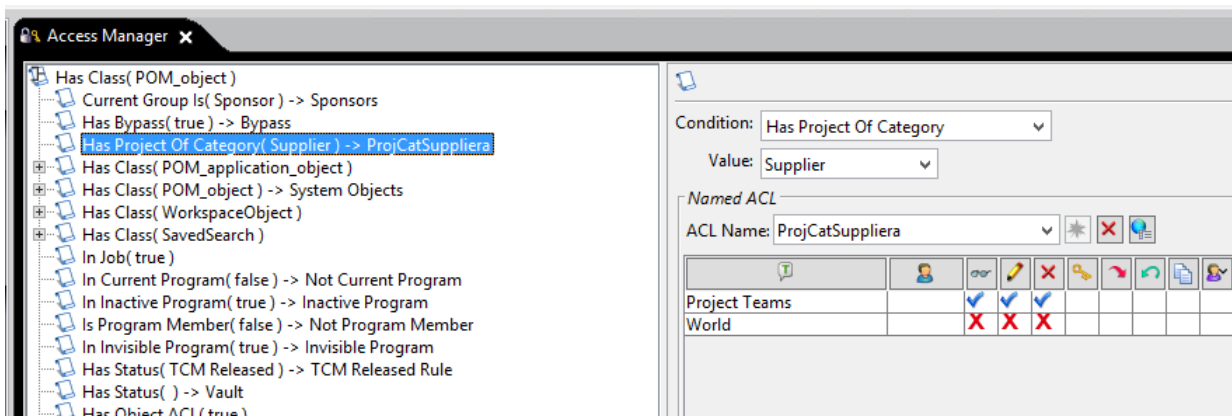
2. Create a project with category as Internal.



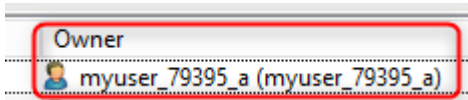
3. Assign user myuser\_a to these projects.



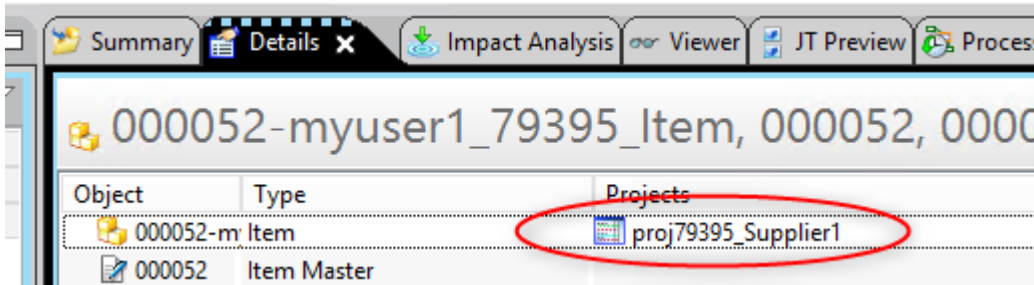
4. Create the rule in Access Manager.



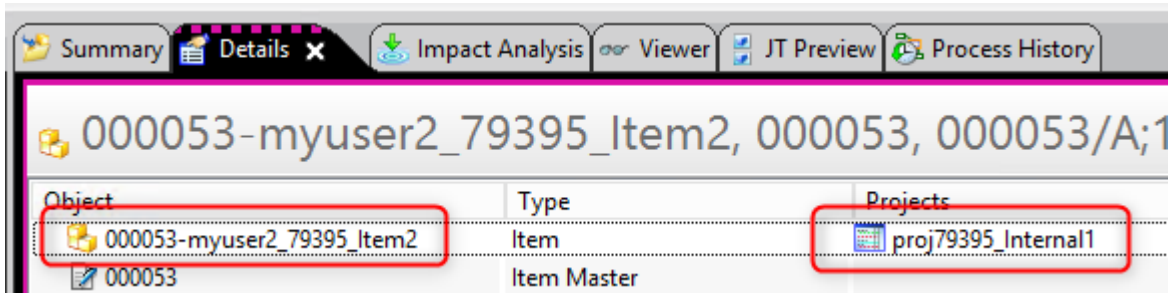
5. Log on as myuser\_a and create two items.



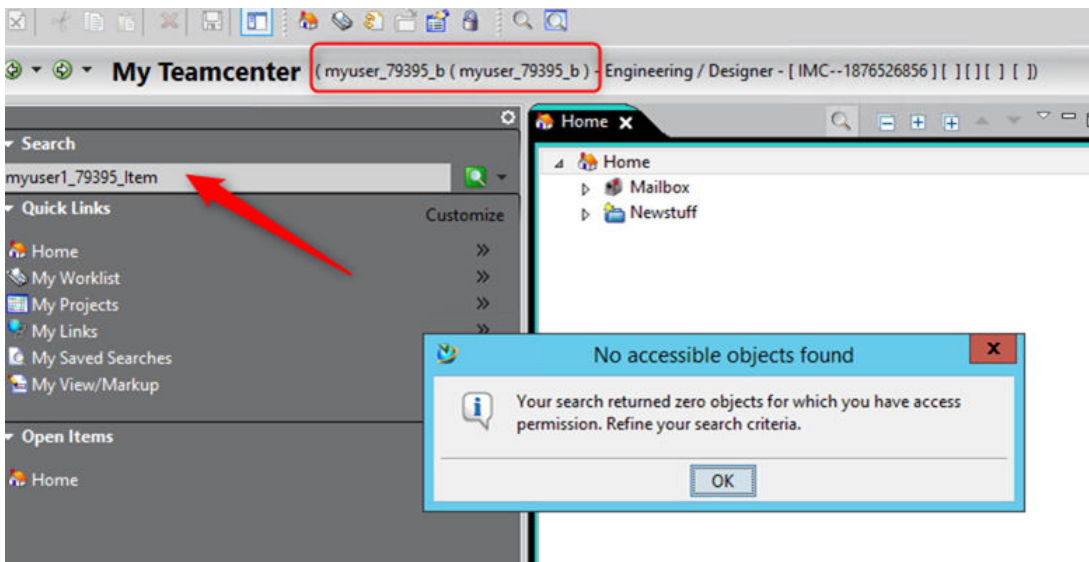
- Assign project with supplier category to item 1.



- Assign project with internal category to item 2.



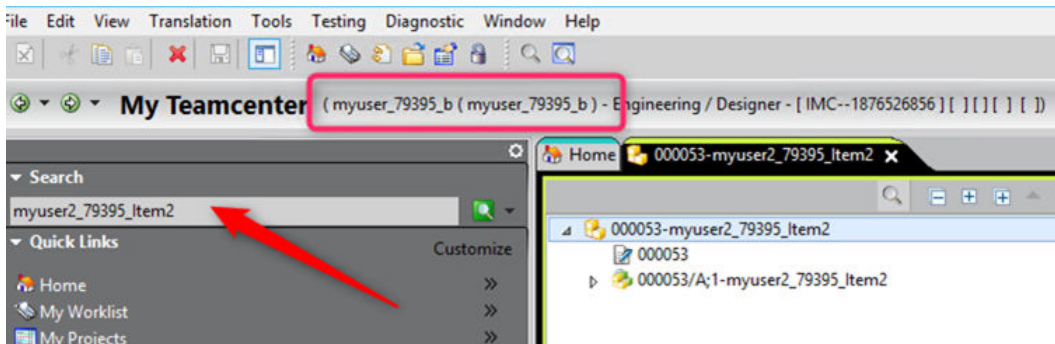
- Log on as myuser\_2
- Search for the item that has a project category of supplier.



Note:

Search will return zero objects because it was restricted by Access Manager.

- Search for item that has a project category of internal.



Note:

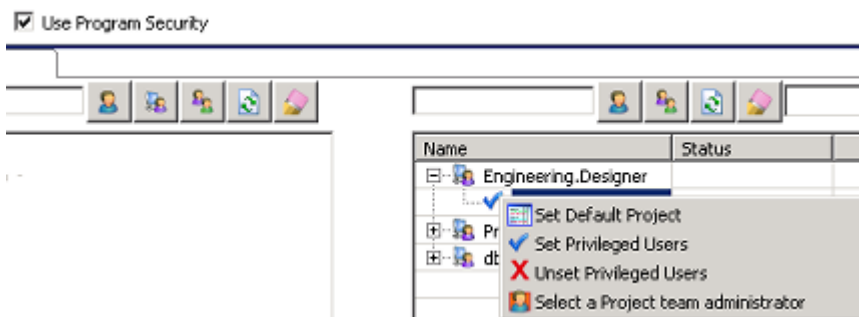
Search will return item because it is accessible to myuser\_2.

## Set a user's default project or program

Note:

- You must select the **Use Program Security** option in the project definition to be able to set default projects.
- Users must be assigned to more than one project or program before you can set a default project or program. If users are only assigned to one project or program, that project or program is their default project or program.

- In the **Project** or **Program** tree, select a default project or program for the user.
- In the **Member selection** area of the **Definition** pane, right-click the user in the **Selected Members** tree, and choose **Set Default Project** or **Set Default Program**.



## Considerations for importing and exporting project or program data in a Multi-Site Collaboration environment

The following points should be considered when importing and exporting project or program data in a Multi-Site Collaboration environment:

- Project and program IDs are case sensitive and must match exactly across sites. Siemens Digital Industries Software recommends that you duplicate projects and programs across sites before attempting to share project or program data.
- Users, groups, projects, and programs must be created at all sites that share project or program data. You can use the **dsa\_util** utility to replicate users and groups. Projects and programs must either be created individually, using the Project or Program application or using the **create\_project** utility, which enables you to create projects and programs individually based on command-line input or in batch using input from a text file.

The project must already exist on the remote site to automatically assign objects to the project. If no matching project or program ID is found at the importing site, the import operation proceeds.

- When an object that is assigned to a project or program is exported, the explicitly assigned ID of the project or program is exported along with the other data associated with the object. When that object (replica) is imported at the receiving site, it is assigned to the project or program with the same ID. If an imported object is assigned to multiple projects or programs, it is assigned to the corresponding projects or programs at the importing site.

Note:

If no matching project or program ID is found at the importing site, the import operation proceeds.

Also, to import data with **owning\_project** assigned, the project and program IDs must match exactly across sites. Otherwise, Teamcenter issues an error message.

- When a replicated object is imported to a remote site by a user who is not a privileged team member, the value of the **TC\_multi\_site\_project\_member\_bypass** preference determines whether the object is assigned to the project or program. If the value of the preference is **true**, the replicated object is assigned to the defined project or program upon import. If the value of the preference is **false**, the replicated object is not assigned to the project or program upon import.
- Use the **TC\_sync\_projects\_with\_owning\_site** preference to synchronize the projects between the master and the replica objects.



## 2. Configuring projects and programs

### Project and program configuration tasks

Configuring projects and programs includes the following tasks:

- Configuring naming rules to control the ID and name format when projects are created.
- Setting preferences that control different aspects of project and program behavior.
- Configuring how the assignment of secondary objects is handled when a primary object is assigned to a project or program.
- Using the **update\_project\_bom** utility to assign BOM data to a project.
- Configuring the **autoAssignToProject** extension (using the Business Modeler IDE) to automatically assign objects to projects when they are created.
- Configuring the list of values displayed for the property of an object based on the project that the object is assigned to.
- Restricting the creation of data to within the context of a project or program by setting the **CreateDataWithNoActiveProgram** global constant.
- Applying access rules to project and program data.

### Default naming rules for projects

Naming rules control the format of project names and IDs, allowing you to apply custom naming conventions to your projects. While naming rules are not required to use projects to organize and secure your data; they ensure that projects are identified in a consistent manner.

Naming rules are created and attached to project properties using BMIDE for Data Model Design.

### Configuring assignment of related secondary objects

#### Assigning secondary objects to projects and programs

Teamcenter administrators can configure Teamcenter to implicitly assign related objects to a project or program when the primary object is assigned to a project or program.

For example, a propagation rule can be defined to assign datasets (secondary objects) to a project or program when it has a specific relationship to an item revision (primary object) assigned to the project or program. This is *level-one* propagation.

**Note:**

Level-one propagation implicitly propagates forms when the dataset in which they are contained is assigned to a project.

## Secondary object propagation rules for projects and programs

The secondary object propagation rules are defined by relation type and apply to any secondary object associated with a primary object by one of the included relation types. By default, the following relation types are included for propagation:

**Specifications**  
**Manifestations**  
**AltReps**  
**Affected Item**  
**Attaches**  
**Solution Item**

The propagation rules are valid when the following actions take place in Teamcenter:

- When an item or item revision is assigned to a project or program, the related objects are also assigned.
- When a related object is cut from the object assigned to the project or program, the related object is removed from the project or program.
- When an attachment (related object) is pasted to an object, the attachment is assigned to the project or program to which the primary object is assigned.
- When an item is imported in a Multi-Site Collaboration environment.
- When an imported object (replica) is assigned to a project or program, the associated secondary objects are also assigned to the project or program according to the propagation rules in effect at the receiving site.

Generally, you propagate secondary items that are related to the primary item by a **Specifications**, **Manifestations**, **Attaches**, or **AltReps** relationship. Secondary items related to primary items by **Requirements** and **References** relationships generally do not need to be implicitly propagated to the project or program.

## Assigning BOM data to projects and programs

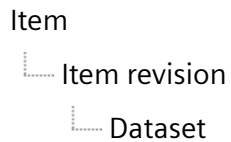
Objects contained in a product or manufacturing structure can be assigned to or removed from projects or programs in batch mode by running the **update\_project\_bom** utility.

**Note:**

Only objects that were added to the project or program using the utility can be removed using the utility.

The utility can be used to assign or remove objects at either the level-one or level-two propagation level.

*Level-one propagation* processes BOM objects and their attached datasets. The following structure (BOM) represents the focus of level-one propagation:

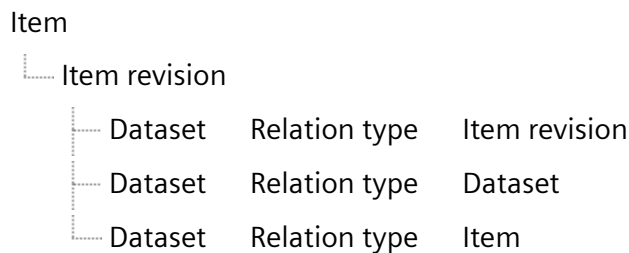


This structure can be nested at multiple levels within a BOM. The depth to which the utility processes the structure is determined by the **-depth** argument. In addition, the utility provides arguments that allow you to specify the type of object, either item or item revision, to be used for level-one propagation, the revision rule by which the BOM should be configured, and the unit number, effectivity date, and ID of the end item associated with the revision rule. Variant rules can also be applied to the structure before propagation.

**Note:**

Objects assigned to a project or program as part of level-one propagation are *implicitly* assigned to the project or program. Therefore, any change to the relationship between the primary and secondary objects is reflected in the project or program.

*Level-two propagation* processes dependent objects related to the datasets associated with level-one objects. The following data model is supported for level-two propagation:

**Note:**

Dependent objects assigned through level-two propagation are *explicitly* assigned to the project or program. Objects that are explicitly assigned to a project or program must also be explicitly

removed. If the BOM object to which the secondary object is related is removed from the project or program, the dependent object remains assigned to the project or program.

## Configuring the automatic assignment of objects

### Assign projects to objects using workflow

You can assign projects to objects using a workflow by:

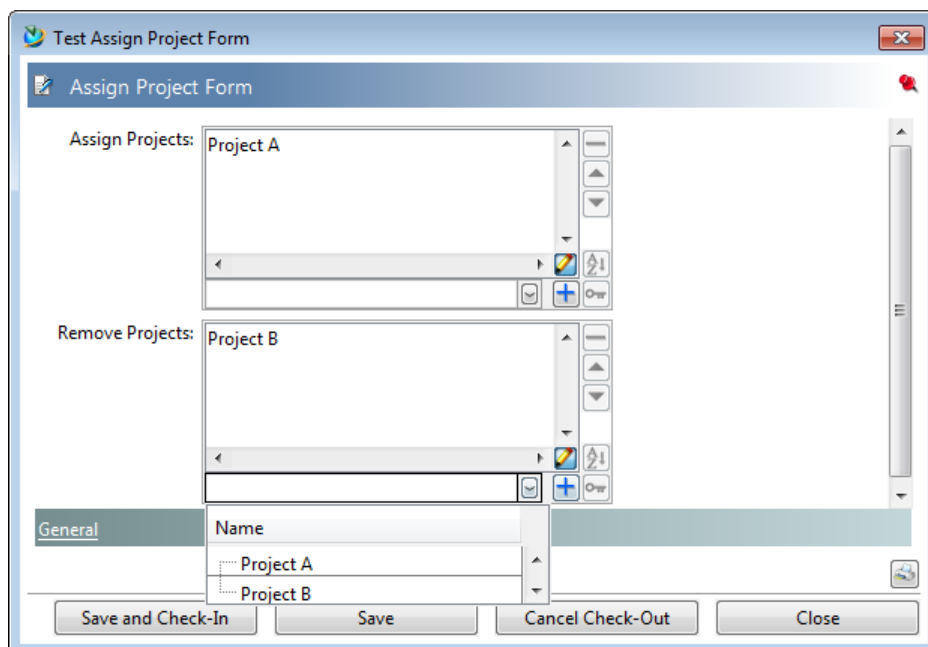
- Using the **PROJ-update-assigned-projects** workflow action handler.

Using the **PROJ-update-assigned-projects** handler, you can assign and remove projects from the target objects using handler arguments only. In the following example, assume that the projects to be assigned are **Proj1** and **Proj2** and that the projects to be removed are **Proj3** and **Proj4**.

Argument	Values
-assign_projects	Proj1,Proj2
-remove_projects	Proj3,Proj4

- Creating a custom form using the Business Modeler IDE.

Using the Business Modeler IDE, create a custom form (for example, **Assign Project Form**) that can be used to assign projects using the **PROJ-update-assigned-projects** handler.



The workflow administrator uses the **PROJ-update-assigned-projects** workflow handler to specify the projects using the properties on the custom form attached to the workflow template.

## Automatically assigning objects to projects and programs

Teamcenter administrators can configure Teamcenter to automatically assign certain types of objects to a project or program when the specified objects are created by privileged team members.

The following object types can be configured for automatic assignment:

- Item and item revision subtypes, such as engineering changes and documents
- Forms
- Datasets

For example, Teamcenter can be configured to assign new item revisions to the *current project or program* of the user who creates the new item revision.

Note:

Your current project or program is defined in the **User Settings** dialog box. You can choose **Edit**→**User Setting** to change your current project or program.

Use the Business Modeler IDE to configure the **autoAssignToProject** extension. This extension defines the type of objects that are automatically assigned to the user's current project or program when the specified object is created. When you assign an object to a project automatically, the project becomes the **Owning Project**.

Note:

When importing or exporting project or program data in a Multi-Site Collaboration environment, you must first create the project on both the export site and the import site prior to using the **autoAssignToProject** extension.

Note:

To be able to remove an object from the owning project, add the **TC\_allow\_remove\_owning\_project** preference and set it to **true**, which allows you to right-click the object and choose **Project**→**Remove**.

Because the **owning\_project** attribute is assigned only during object creation, you cannot set or change the **owning\_project** attribute on objects that are already created. To assign the owning project for newly created objects, you must set the following:

- Configure the **autoAssignToProject** postaction extension on the create operation of the business object.
- Set the session project in the Teamcenter client.

To apply this rule to an object, you must:

- Set the owning project of the object.
- Enable the program context on the owning project.
- Make certain the user is a member of any of the object's projects, not necessarily associated with the owning project.

Also, you can create propagation rules in the Business Modeler IDE to:

- Automatically assign objects to projects. This rule copies property values from one object type to another object type using a relationship or a property reference when a certain operation occurs (for example, at checkin).
- Control the propagation of security data (project assignments, ADA licenses, and Classification attributes).

## Implications of the autoAssignToProject extension on propagation rules

An administrator can attach the **autoAssignToProject** extension to a business object operation **PostAction** extension point. When the action occurs, the extension automatically assigns the user's current project defined by the work context or user settings.

Configuring the **autoAssignToProject** internal extension for a business object has implications on the project propagation rules. Project propagation rules determine which secondary objects are assigned to a project when a primary business object is assigned. When there is a conflict between a propagation rule and the execution of the **autoAssignToProject** extension, the extension takes precedence.

## Improving performance for bulk operations

As of Teamcenter release 14.2, enhancements have been made to improve performance in the case of bulk operations. When the **autoAssignToProject** extension is attached to the type-specific object and object-set operations listed in column **Attach to the following operations**, then the project assignment extension is called only once per object type and its subtypes. If the extension is already attached to the operations listed in column **Detach from following operations**, then detach the extension from these operations to gain the improved performance.

Type (and all its subtypes)	Attach to the following operations	Detach from following operations
Item	IMAN_save	IMAN_import ITEM_create

Type (and all its subtypes)	Attach to the following operations	Detach from following operations
	fnd0Save	ITEM_create_from_rev
Item revision	IMAN_save fnd0Save	ITEM_baseline_rev ITEM_copy_rev ITEM_copy_rev_to_existing ITEM_create_rev
Dataset	AE_save_dataset fnd0Save	
Form	IMAN_save fnd0Save	

## Interactions with propagation rules

The following scenarios illustrate the relationship between extensions and propagation rules when assigning objects to projects.

### Note:

- If a current project is not specified for the user, this extension is ignored and the object is not automatically assigned. In addition, when the **autoAssignToProject** extension is configured for an item or ECO, the project name is preselected in the **Assign to Projects** page of the item or ECO create, revise, and save as dialog boxes.
- If you want users to be able to remove objects from an owning project, you must create the **TC\_allow\_remove\_owning\_project** preference before using the **autoAssignToProject** extension. If this preference is not set, objects assigned to owning projects cannot be removed using the **Project→Remove** command.

The following points must be considered when implementing the **autoAssignToProject** extension:

- The **autoAssignToProject** extension applies only to newly created objects; whereas, propagation of related objects to projects occurs whenever a relation between two objects is created, modified, or deleted.
- The **autoAssignToProject** extension explicitly assigns objects to projects; therefore, the objects can only be removed from the project by explicitly right-clicking the object in the Teamcenter rich client and choosing **Project→Remove**.
- Propagation rules implicitly assign secondary objects to projects. Therefore, when the primary object is explicitly removed from the project, the secondary object is also removed from the project.

- When importing or exporting project or program data in a Multi-Site Collaboration environment, you must first create the project on both the export and the import site prior to using the **autoAssignToProject** extension.

Scenario	Project assignment behavior
The <b>autoAssignToProject</b> extension is configured for types <b>P</b> (primary object) and types <b>S</b> (secondary object). A user creates an object of type <b>P</b> and an object of type <b>S</b> related by the <b>Requirements</b> relation.	Both objects are automatically assigned to the current project, regardless of whether the <b>Requirements</b> relation is specified in the propagation rule list.
The <b>autoAssignToProject</b> extension is configured for types <b>P</b> (primary object), but not for types <b>S</b> (secondary object). A user creates an object of type <b>P</b> and an object of type <b>S</b> related by the <b>Requirements</b> relation.	The object of type <b>P</b> is automatically assigned to the current project based on the <b>autoAssignToProject</b> extension. If the <b>Requirements</b> relation is specified in the propagation rule list, the type <b>S</b> object is also assigned to the project. If the <b>Requirements</b> relation is not specified in the propagation rule list, the secondary object is not assigned to the project.
The <b>autoAssignToProject</b> extension is configured for types <b>P</b> (primary object) and types <b>S</b> (secondary object). In addition, the <b>Requirements</b> relation is defined as a propagation rule. The user creates an object of type <b>P</b> and an object of type <b>S</b> . After creating the objects, the user attaches the secondary object to the primary object using the <b>Requirements</b> relationship.	Both the primary and secondary object are automatically assigned to the project based on the configuration of the extension, resulting in an explicit assignment rather than the implicit assignment that occurs when an object is assigned to a project based on propagation rules.

## Restricting data creation to within the context of a project or program

Set the **CreateDataWithNoActiveProgram** global constant to **false** to restrict data creation to within the context of a program. When this constant is set to **false**, users can only create data when:

- An active program is set for the user session.
- **Program-level security** is enabled for that program.

This feature restricts creation of the following data types (and subtypes of these data types):

- Item
- Item revision
- Dataset

- Collaboration context
- Identifier

Other data types, such as folders and forms, are not restricted by the **CreateDataWithNoActiveProgram** global constant.

To allow data creation outside the context of a program, set the global constant **CreateDataWithNoActiveProgram** to **true**.

## Controlling assignment of replicated (imported) data to projects

By default, replicated objects are assigned to the defined projects even if the importing user is not the member of the projects because the **TC\_multi\_site\_project\_member\_bypass** preference bypass is set to **true**. If you want to restrict the project assignment during import, set this preference to **false**.

## Setting the current project for new Teamcenter sessions

You can determine whether users' current project settings persist from one Teamcenter session to the next by setting the **TC\_set\_current\_project** preference. The default behavior is to save the user's current project setting and apply it the next time they log on to Teamcenter.

## Selecting different object property values based on project assignment

As a Teamcenter administrator, you can configure the system so that users see different lists of property values based on the project that the object is assigned to. For example, if an item is assigned to project A, the list of values attached to the **Color** property shows one set of colors to choose from, and if the item is assigned to project B, the list of values attached to the **Color** property shows another set of colors.

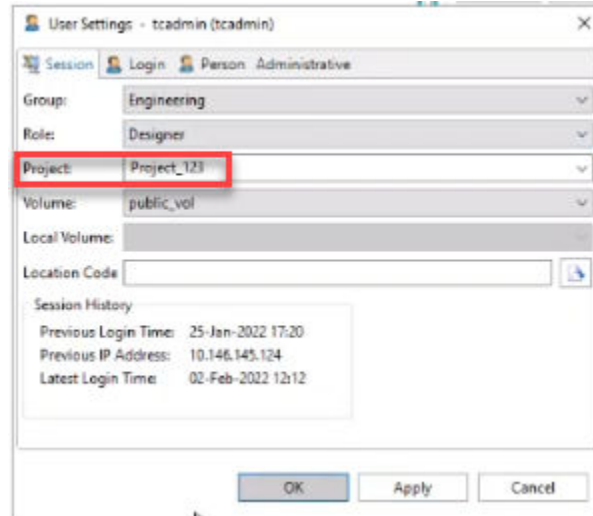
These conditional lists of values (LOVs) are defined in the Business Modeler IDE. The conditions can only be defined using the **owning\_project**, **project\_list**, **owning\_user**, and **owning\_group** project properties.

## Displaying projects to users based on current group and role

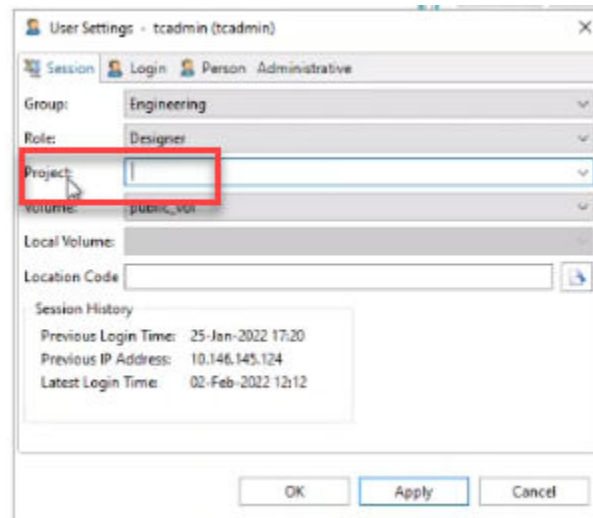
You can limit which projects are displayed to users throughout the Teamcenter interface, for example, in the **New Item** dialog box or the **Assign an Object to a Project** dialog box. By default, all active projects in which the user is a privileged team member are displayed. You can set the **TC\_show\_all\_user\_projects** preference to **false** to limit the display to only the projects that correspond to the user's current group and role.

## Example

When you set the **TC\_show\_all\_user\_projects** preference to **true**, the group and role are not considered and the user's project is displayed in the **User Settings** dialog.



However, when you set the **TC\_show\_all\_user\_projects** preference to **false**, if the user is not a member of the project that is set in the user session with the group/role combination, the **Project** field on the **User Settings** dialog is empty.



## Using pseudofolders to organize the display of objects in projects

You can specify properties to be displayed as pseudofolders under a node of the **TC\_Project** business object. After you specify a pseudofolder, all objects defined in that property are shown in the folder under the **TC\_Project** object.

# 3. Configuring smart folder filtering

## What are project smart folders?

Propagation of related data to projects can result in projects that contain thousands of objects, many of which are not relevant to project team members (users). You can use *smart folders* to filter project data.

Smart folders:

- Are only available in the rich client.
- Provide a mechanism for configuring arbitrary subdivisions of data within a project based on functional and subfunctional units.
- Allow you to filter project data to only the subset of data that is relevant to the project members.
- Maintain the context of the project family and program to which the individual projects belong.

Note:

The project administrator must be a member of the **dba** group to be able to administrate project smart folders.

**Filtering** is applied using the Project application, and the filtered **view of project data** is presented to users in the **My Projects** pane in My Teamcenter.

## Types of smart folders

There are two types of smart folders: *abstract smart folders* and *data-driven smart folders*.

Abstract folders:

- Serve as funnels through which project data is filtered.
- Can be nested.
- Can be placed anywhere above or below the project folder in the hierarchy.
- Are used to configure the top-level project-specific smart folder hierarchy.

Data-driven smart folders:

- Are the leaf nodes in the project hierarchy.

- Display the data objects assigned to the project.

Note:

Smart folders are pseudofolders, they are not physical folder objects in Teamcenter.

## Understanding the project smart folder hierarchy

The smart folder hierarchy is created and maintained using Business Modeler IDE and Project. It consists of three distinct components:

- Top-level smart folder hierarchy.
- Project smart folder hierarchy.
- Properties by which objects (items) in projects are associated with folders in the hierarchy.

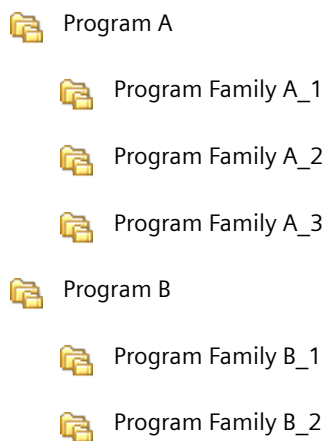
## Top-level smart folder hierarchy

The top-level smart folder hierarchy:

- Represents high-level functional units, such as programs and associated program families.
- Provides context and is visible to all users who are members of projects within the hierarchy.

## Top-level hierarchy example

The following figure illustrates a top-level folder.



Each branch in the hierarchy is represented by a list of values (LOV) that specifies subbranches in the hierarchy. The **Project Top Level Smart Folders** LOV acts as the root of the hierarchy. It defines the branches in the hierarchy by specifying the LOV associated with each branch.

To create the project top-level smart folder hierarchy shown above, you must:

- Create three exhaustive string LOVs, beginning with the LOVs representing the branches.
- Specify the **Project Top Level Smart Folders** LOV as the value for the **ProjectTopLevelSmartFolders** global constant.

This enables the display of top-level smart folders in Project.

- Associate data objects with smart folders.

Note:

The LOV limit is 32 characters for LOV names.

## Creating the example project top-level hierarchy

The following process is used to create the top-level smart folder hierarchy shown in *Top-level hierarchy example*:

1. Create the **Program A** list of values (LOV), which represents the **Program A** folder in the hierarchy.
2. Add the following values to the **Program A** LOV:

**Program Family A\_1**  
**Program Family A\_2**  
**Program Family A\_3**

These values represent the folders that are sub-branches of **Program A**.

3. Create the **Program B** list of values (LOV), which represents the **Program B** folder in the hierarchy.
4. Add the following values to the **Program B** LOV:

**Program Family B\_1**  
**Program Family B\_2**

These values represent the folders that are sub-branches of **Program B**.

5. Create the **Project Top Level Smart Folders** list of values (LOV), which represents the root of the hierarchy.

6. Add the following values to the **Project Top Level Smart Folders** LOV:

**Program A**  
**Program B**

These values connect the **Program A** and **Program B** branches to the root of the hierarchy.

7. Specify the **Project Top Level Smart Folders** as the value of the **ProjectTopLevelSmartFolders** global constant.

## Associating data objects with smart folders

Item properties are used to filter data objects that are assigned to projects.

The properties used as filters:

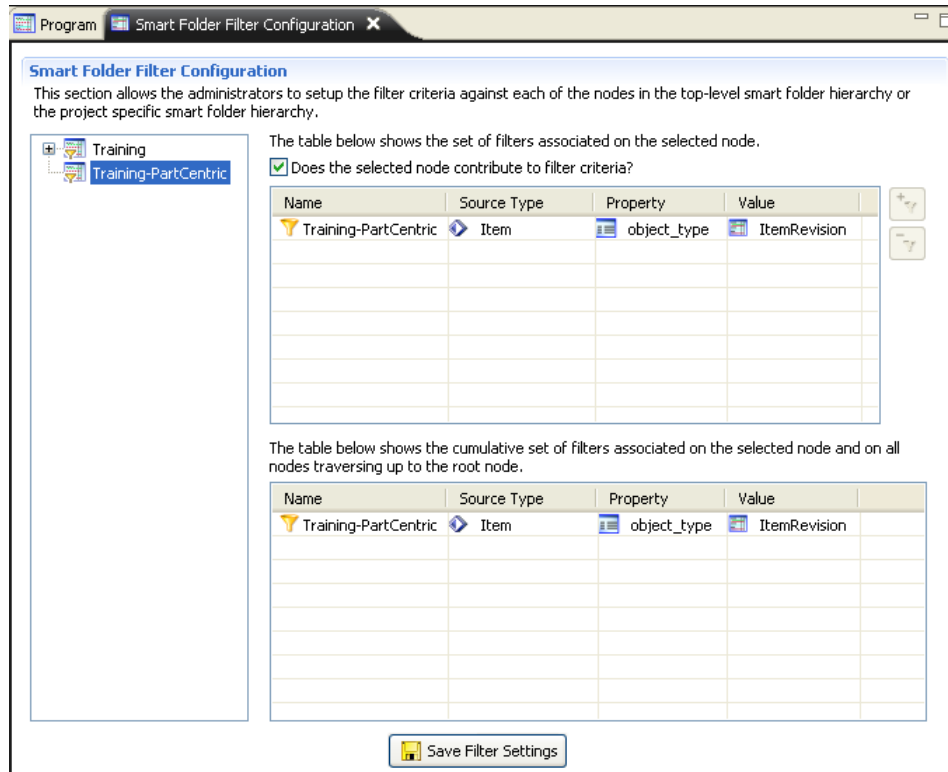
- Can be persistent properties.
- Can be compound properties.
- Must be available on the respective object types.
- Are generally custom properties defined by creating a new form and adding a compound property rule to display the properties from the form on the item object.

## Using properties to filter data

In the figure shown in *Top-level hierarchy example*, the top-level smart folder hierarchy consists of programs and program families.

The following process is used to associate items with a program smart folder or a program family smart folder:

1. Create custom attributes named **program** and **program family**.
2. Expose the **program** and **program family** attributes on the item object.
3. Assign values to the properties of the items that you want to filter.
4. Configure filtering in the Project application.

**Note:**

Filtering is based on a combination of smart folder, source type, property, and value. Multiple filter criteria can be applied to data-driven driven smart folders; however, only one filter criteria can be applied to an abstract smart folder.


## Project smart folder hierarchy


The *project smart folder hierarchy* represents subdivisions of the data contained in a project.

The project smart folder hierarchy can be used in the context of a top-level smart folder hierarchy or independently of a top-level hierarchy.


## Project smart folder hierarchy example


The following figure illustrates a smart folder hierarchy that represents the projects available at a Teamcenter site.


 T-Upper Body (*Project*)


 T467-Side Fittings (*Abstract folder representing a subfunctional unit within the project.*)

 T0247-Frame Backlit Glass (*Data-driven folder that displays filtered project data.*)

 T0248–Frame Frontlit Glass (*Data-driven folder that displays filtered project data.*)

 L-Platform Body (*Project*)

 L678–Side Fittings (*Abstract folder Abstract folder representing a subfunctional unit within the project.*)

 L0395–System Fuel Feed (*Data-driven folder that displays filtered project data.*)

 L0396–Trunk Lid (*Data-driven folder that displays filtered project data.*)

---

To create the project smart folder hierarchy shown above, you must:

- Create an exhaustive string LOV with values derived from the **TC\_Project.project\_id** reference class.
- Specify the LOV as the value of the **ProjectSmartFolders** global constant.
- Associate data objects with smart folders.

#### Example

This example illustrates:

- Setting up projects using smart folders.
  - Referencing the **project\_id** when creating LOVs.
1. Create two projects: **ProjA** and **ProjB**.
  2. In BMIDE, create a template: **SF9test**.
  3. Add three classic LOVs:
    - **SF9ProjA** LOV and add two values: **SmartFolder1** and **SmartFolder2**.
    - **SF9ProjB** LOV and add two values: **SmartFolder3** and **SmartFolder4**.
    - **SF9ProjectSF** LOV.
  4. Click the **Browse** button in the **Reference** section and select **TC\_project.project\_id**.
  5. Click the **Load** button in the **LOV Values** section.

Note:

User name and password are required to log on to the server.

6. To create cascading LOVs:
  - a. Select the **Show Cascading View** check box.
  - b. Select **ProjA** from the list, click the **Add Sub LOV** button, and select the **SF9ProjA** LOV.
  - c. Select **ProjB** from the list, click the **Add Sub LOV** button, and select the **SF9ProB** LOV.
  - d. Click **Finish**.
7. Open the **Global Constants Editor**. Select the **ProjectSmartFolders** global constant and set the value to **SF9ProjectSF**.
8. Save the data model.
9. Deploy the template.

## Apply smart folder filtering

Smart folders provide a mechanism for configuring arbitrary subdivisions of data within a project or program based on functional and subfunctional units. Smart folders are pseudofolders configured in a hierarchical structure; they are not physical folder objects in Teamcenter.

A project or program hierarchy that defines the abstract and data-driven folders used to filter project or program data based on object type and attribute is created in Business Modeler IDE. This filtering controls what is displayed when users access their project or program in the **My Projects** view, and it enables them to easily locate data in large projects and programs.

After the folder hierarchy is defined in Business Modeler IDE, you can choose object types, properties, and property values to filter data for display.

You can also filter project or program data in the absence of a smart folder hierarchy by applying filters within a project or program.


**Note:**

The ability to use smart folders to structure projects is available in the rich client environment only.

1. In Project or Program, click the **Smart Folder Administration** quick link.
2. Expand the hierarchy tree to display the smart folders.

**Note:**

If a smart folder hierarchy is not defined, only the project or program objects are displayed in the tree.

3. Select a smart folder  that you want to use to apply filtering criteria.

4. Select the **Contribute to Filter Criteria** check box.

The smart folder name appears in the filter table beneath the check box.

5. Select an object type from the **Type** list.

6. From the **Property** list, select a property corresponding to the object type.

When the project or program hierarchy is evaluated, the smart folder funnels project or program data of the specified type and property value to the user's project or program display.

7. Continue down the tree, selecting folders and applying object types and properties as filtering criteria.

The cumulative filter criteria is displayed in the table at the bottom of the window.

8. When you reach the leaf node (data driven folder) that displays the project or program data, you can define additional filter criteria as follows:


a. Select the data driven folder in the tree.

b. Select the **Contribute to Filter Criteria** check box.

The folder name appears in the filter table beneath the check box.

c. Select an object type from the **Type** list.

d. From the **Property** list, select a property corresponding to the object type.

e. Click **Add a filter** button  to add a new row to the filter table that is used to define additional filters for data-driven folders.

**Note:**

Additional filtering criteria can only be added to data-driven folders. Abstract folders in the hierarchy only accept a single **filter criterion**.

- f. In the new row of the table, select the object type and property.
- g. Enter a value for the property in the **Value** column.

When setting values to filter for user and group properties, you can filter project data for the logged-on user or group by setting the value to **\$USER** or **\$GROUP**. In addition, you can filter data based on properties that have a null value by entering **\$NULL**.

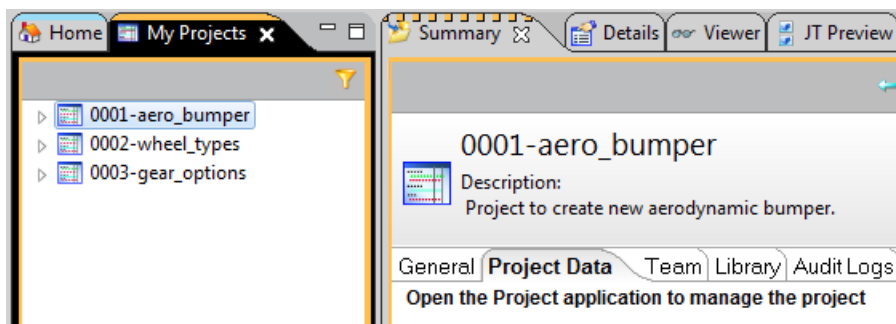
9. Click **Save Filter Settings**.

The filter settings are applied to the project. In the **My Projects** view, the data is displayed according to this filter criteria.

## Viewing project data


### Viewing project data in My Teamcenter

You can click the **My Projects** link in the navigation pane to view your projects and associated data. This view is determined by the project administrator.



The **My Projects** view is similar to My Teamcenter views but with the following differences:

- The **Preferred Items** folder is not displayed in the **My Projects** view.
- **Project Smart Folder Hierarchy**

Displays the smart folder hierarchy as configured by the administrator and provides a filter button  to display the **Apply filters** dialog box.

Note:

The **Apply filters** dialog box lets you apply filtering based on source types, properties, and values.

You select a project from the hierarchy to display information in the other **My Projects** panes.

- **Summary**

The summary view consists of two tabs:

- **Project Data**

Displays the filtered objects assigned to the selected project.

- **Team**

Displays the project team associated with the selected project.

Note:

The content of the **My Projects** view varies when you install CPG.

## Project smart folder filtering

The following considerations apply to smart folder project filtering:

- Only one filter criteria can be applied to an abstract smart folder 📁.
- Multiple filter criteria can be applied to a data-driven smart folder 📁. When the multiple filter criteria is applied, all data that meets the criteria appears under the project or program.
- The **TC\_UI\_SmartFolder\_JoinOperator** system preference applies the filter criteria between different filters on data-driven smart folders. The **TC\_UI\_SmartFolder\_JoinOperator** preference is not provided OOTB but can be created if required. If you do not create a preference, the default filter criteria applied between multiple filters is **OR**.

By default, **AND** is applied between the filters of abstract folders and data driven folders. Following is an example of how smart folder filters are applied:

```
Folder1 (Abstract folder)-> Filter1
Folder2 (Data driven folder) -> Filter2, Filter3
```

If the preference is set to **AND**, then the resulting filter is:

```
Filter1 AND (Filter2 AND Filter3)
```

If the preference is set to **OR**, then the resulting filter is:

```
Filter1 AND (Filter2 OR Filter3)
```

By default the filter criteria for any non-string, runtime and compound properties is **AND**. If one of the filters defined contains a non-string/runtime/compound property, the filter criteria is always **AND**.

```
Folder1 (abstract folder) -> Filter1
Filter1 = source_type=ItemRevision, Property = object_type,
Value=<Any value>
Folder2 (data driven folder) ->Filter2, Filter3
Filter2 = source_type=ItemRevision, Property = object_type,
Value=<Any value>
Filter3 = source_type=ItemRevision, Property = ics_subclass_name,
Value=<Any value>
```

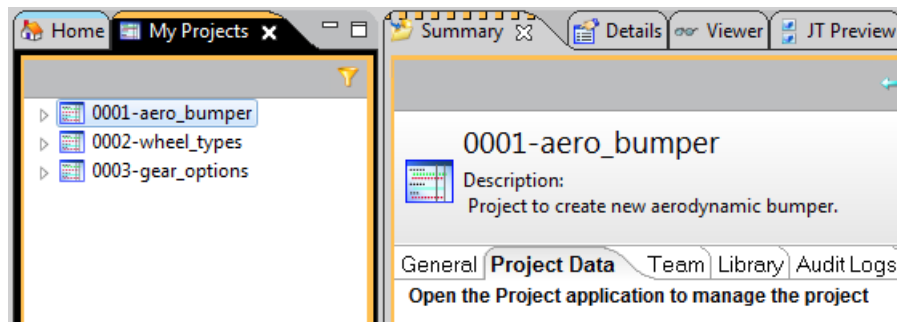
If a run time property is defined in the filter, the filter criteria applied is **ALWAYS**.

```
Filter1 AND (Filter2 AND Filter3)
```

## Apply project smart folder filtering

1. Click **My Projects** in the **Quick Links** section of the navigation pane in My Teamcenter.

The **My Projects** view appears.



2. In the **My Projects** view, expand a project object  to view the data associated with the project.


**Note:**

Objects are displayed according to the project smart folder hierarchy configuration defined by the project administrator and/or by any filters that you apply.


If a smart folder hierarchy is configured for your project, you see folders when you expand your project. Expand these folders to reveal your project data.

3. To apply filters, select the project object in the **My Projects** view.

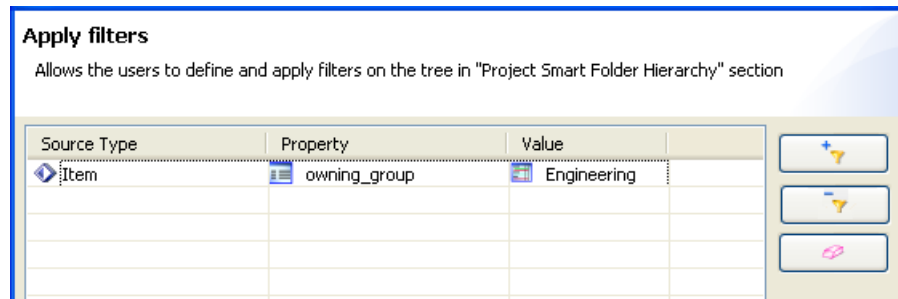
The **Summary** view shows details of the project object. The contents of the **Summary** view vary according to the type of object selected.

- Click the filter button  to apply a filter to refine the list of objects for the selected project.



The **Apply filters** dialog box appears.

- Click the **Add a filter** button  to populate a row in the table.
- Choose filter criteria by clicking the **Source Type**, **Property**, and/or **Value** cells and selecting a value from the list.

For example, you can add a filter to display the project data owned by the **Engineering** group.



Note:

You can remove rows from the filter table by clicking the **Remove the selected filter** button . Click the **Clear the filters** button  to remove all values from the table.

- Click **OK** to apply the filter.

The list of objects in the **My Projects** view is updated to reflect the applied filters.

Note:

The filters you apply remain until you remove them. However, you cannot save a filter configuration and retrieve it for future use.