TEAMCENTER

Partner Connect — Deployment and Administration

Teamcenter 2412

SIEMENS

About Siemens Digital Industries Software

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

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

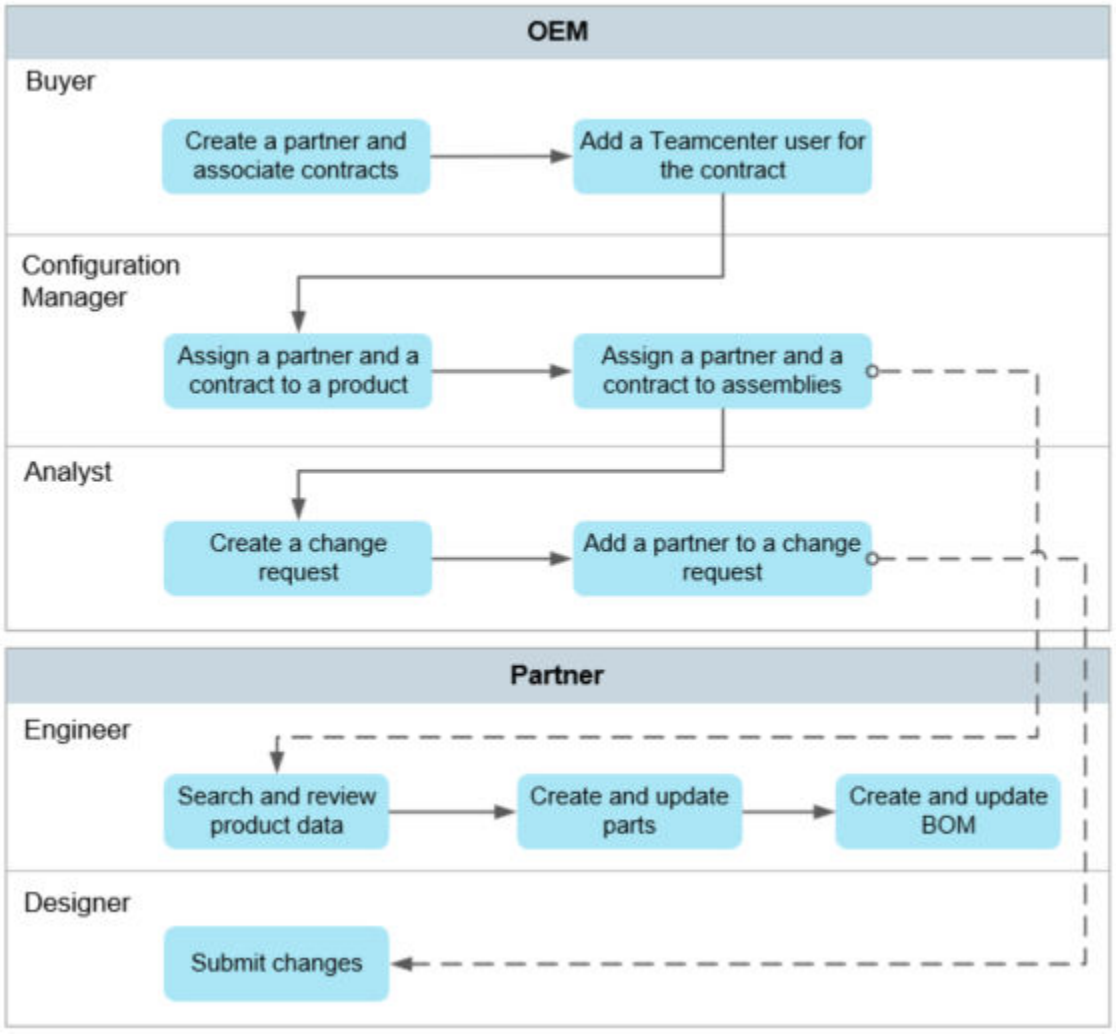
Send Feedback on Documentation: **support.sw.siemens.com/doc_feedback_form**

Contents

1. About Partner Connect

Companies looking to reduce cost and overheads rely on outsourcing. One of the business models that companies rely on when outsourcing is referred to as Contract Manufacturing. This is when a company partners with a manufacturer to deliver an entire product or significant parts of the product. The company owns the product design, while the manufacturing partner is responsible to deliver the finished goods based on contractual obligations. This process requires tight collaboration between the company and the *partner* to exchange product data, manage changes, and track related documentation. Partner Connect helps companies, especially OEMs, work closely with their partners in an integrated Teamcenter environment.

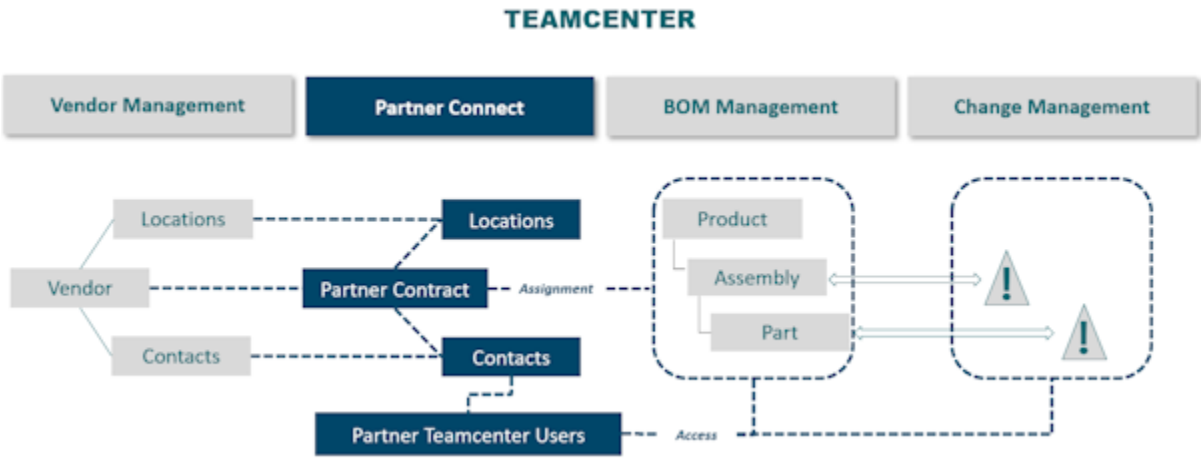
The Partner Connect process flow is as follows:



Using Partner Connect, a company can enable its partners to log on to their Teamcenter environment and access only information that is relevant to the products that they are working on. The access is controlled by using a partner contract, which allows a company to set time-bound access to specific product information in Teamcenter. In the case of a company working with multiple partners, the

partner contract also limits one partner from viewing information assigned to or managed by another partner.

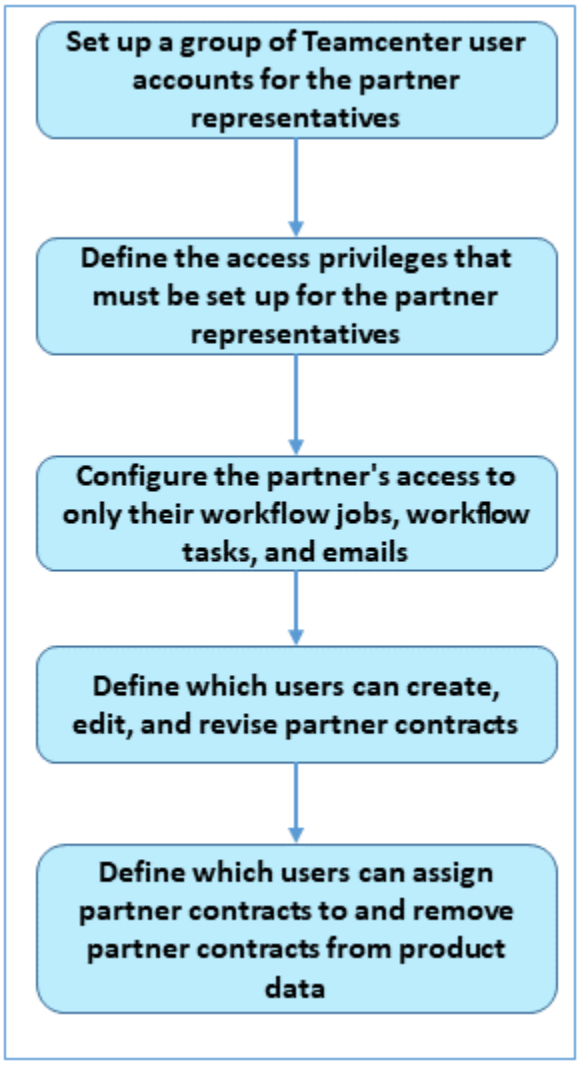
The role of Partner Connect in Teamcenter is as follows:



2. Set up Partner Connect in Teamcenter

You must install the **Vendor Management** template and the **Partner Collab** license to use Partner Connect.

The following tasks are the sequence of tasks required to set up Partner Connect in Teamcenter:



The representative of the partner assigned to a partner contract must be able to log on and access only their assigned product data, workflow jobs, workflow tasks, and emails in Teamcenter. You must define this access to ensure that your Teamcenter data is secure and that the partner representatives can access only the assigned objects. Do the following:

- **Set up a group of Teamcenter user accounts that the partner representatives will use to log on to the OEM's Teamcenter and work on their assigned product and part data.**

- **Define the access privileges that must be set up for the partner representatives by creating the required ACLs.**

- **Configure the partner's access to only their workflow jobs, workflow tasks, and emails.**

In addition to configuring the partner's access, you must also define which users can create, edit, and revise partner contracts of a vendor; and assign partner contracts to and remove partner contracts from product data. Do the following:

- **Restrict only users with the role of Buyer to create, edit, and revise partner contracts of a vendor.**

- **Restrict only users with the role of Engineer to assign and remove partner contracts from product and part data.**

3. Set up a group of Teamcenter user accounts for the partner representatives

Partner representatives need Teamcenter user accounts to log on to the OEM's Teamcenter. These accounts must be added to a group with its security set to **External** to ensure that partner representatives can access only the data assigned to them.

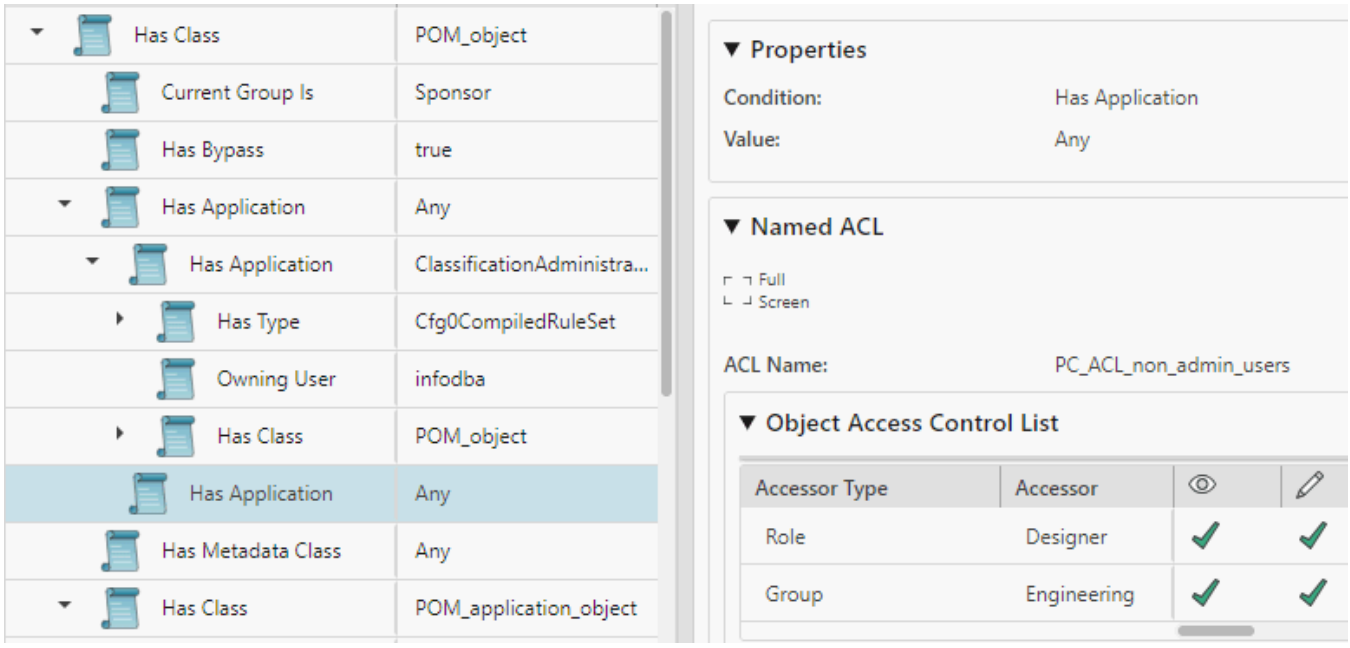
1. In Teamcenter Organization, create a group and add the required user accounts.

2. Set the security of the group to **External**.

4. Configure the approval permissions for non-administrator users

When a partner contract is sent for approval to non-administrator users, they must have the required permissions to approve or reject the partner contract. To configure this, in Teamcenter Access Manager, select the **Has Class (POM_object)** > **Has Application (Any)** node from the Access Manager (AM) rule tree, and create the **PC_ACL_non_admin_users** ACL.

Condition	Value	Accessor Type	Accessor	Privileges
Has Application	**Any**	**Role**	**Designer**	Grant these privileges: • **Read** • **Write**
Has Application	**Any**	**Group**	**Engineering**	Grant these privileges: • **Read** • **Write**



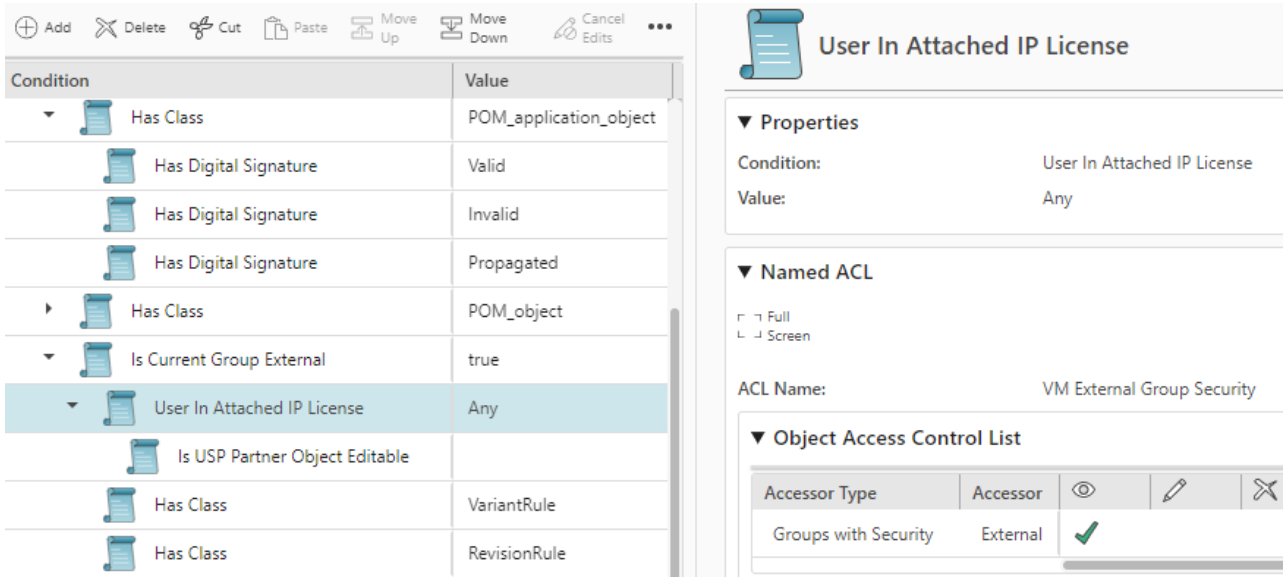
5. Define the access privileges for the partner representatives

Define access privileges for partner representatives to ensure they have the appropriate permissions to access necessary information. This helps maintain data security, compliance, efficient collaboration, and protect sensitive information.

To define the access privileges, create the following access control lists (ACLs) in Access Manager:

1. Select the **Has Class (POM Object)** > **Is Current Group External** node from the AM rule tree and create the **VM External Group Security** ACL with the following details:

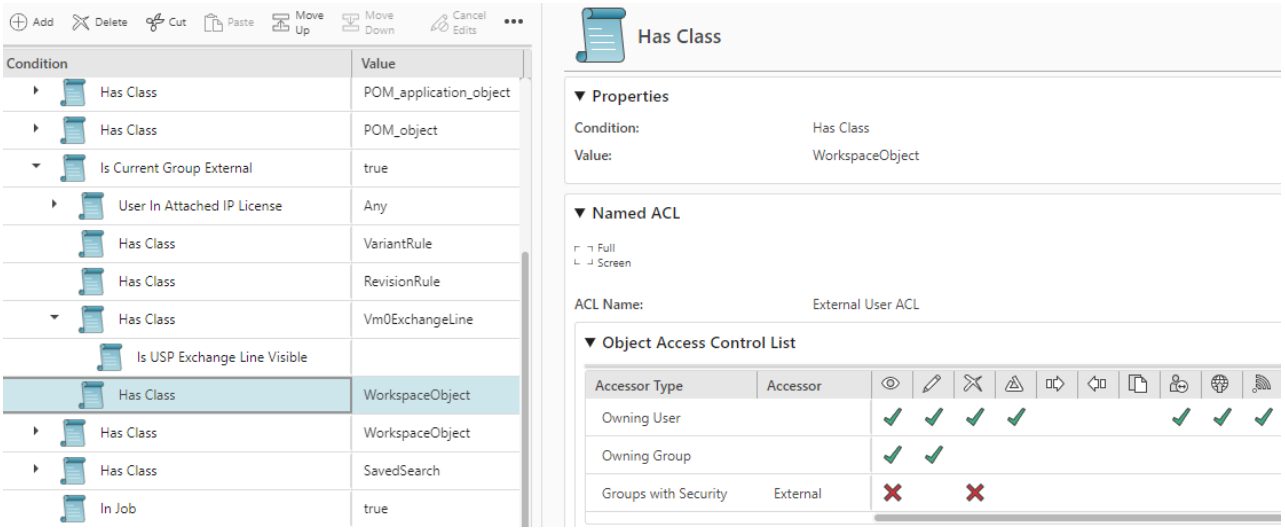
Condition	Value	Accessor Type	Accessor	Privilege
User In Attached IP License	**Any**	**Groups with Security**	**External**	Grant the **Read** privilege.



2. Select the **Has Class (POM Object)** > **Is Current Group External** node from the AM rule tree and create the **External User ACL** ACL with the following details:

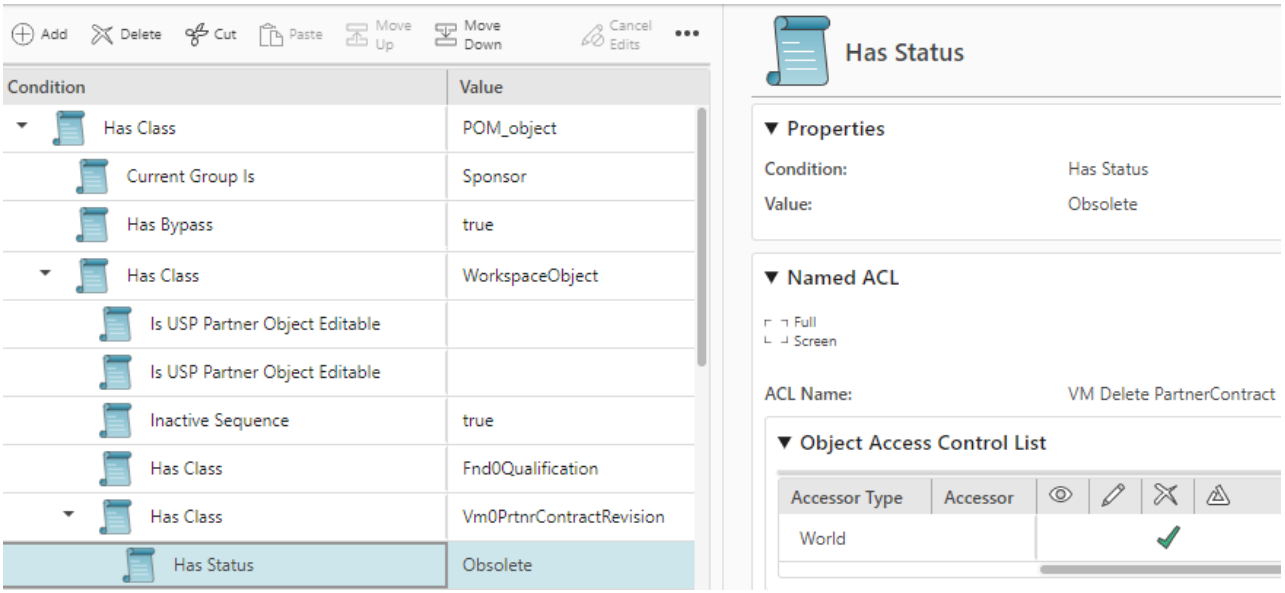
Condition	Value	Accessor Type	Accessor	Privileges
Has Class	**WorkspaceObject**	**Owning User**		Grant these privileges: • **Read** • **Write**

Condition	Value	Accessor Type	Accessor	Privileges
				• **Delete** • **Change** • **Change Ownership** • **Publish** • **Subscribe**
		Owning Group		Grant these privileges: • **Read** • **Write**
		Groups with Security	**External**	Deny these privileges: • **Read** • **Delete**



3. Select the **Has Class (POM Object)** > **Has Class (WorkspaceObject)** > **Has Class (Vm0PrtnrContractRevision)** node from the AM rule tree and create the **VM Delete Partner Contract** ACL with the following details:

Condition	Value	Accessor Type	Privilege
Has Status	**Obsolete**	**World**	Grant the **Delete** privilege.

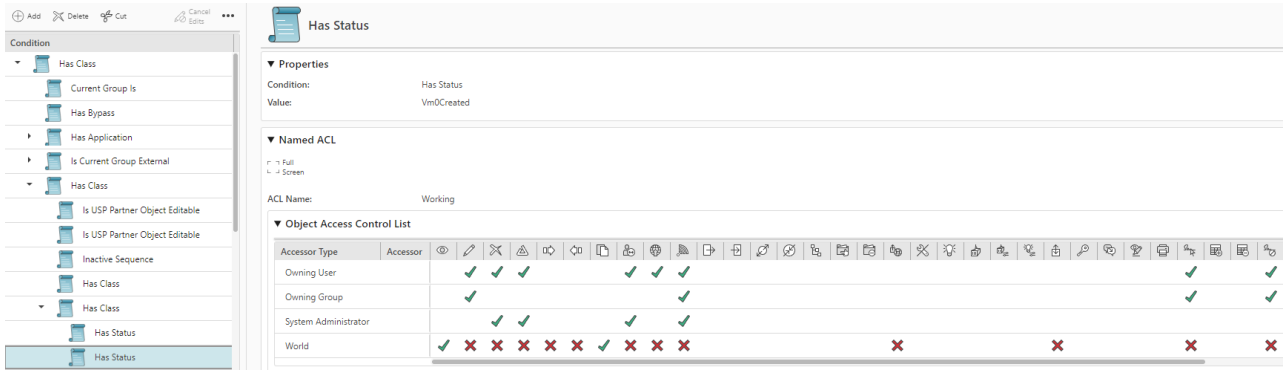


4. Select the **Has Class (POM Object)** > **Has Class (WorkspaceObject)** > **Has Class (Vm0PrtnrContractRevision)** node from the AM rule tree and create the **Working** ACL with the following details:

Condition	Value	Accessor Type	Privileges
Has Status	**Vm0Created**	**Owning User**	Grant these privileges: • **Write** • **Delete** • **Change** • **Change Ownership** • **Publish** • **Subscribe** • **Digitally Sign**

Condition	Value	Accessor Type	Privileges
			• **Void Digital Signature**
		Owning Group	Grant these privileges: • **Write** • **Subscribe** • **Digitally Sign** • **Void Digital Signature**
		System Administrator	Grant these privileges: • **Delete** • **Change** • **Change Ownership** • **Subscribe**
		World	Grant these privileges: • **Read** • **Copy** Deny these privileges: • **Write** • **Delete** • **Change** • **Promote** • **Demote** • **Change Ownership** • **Publish**

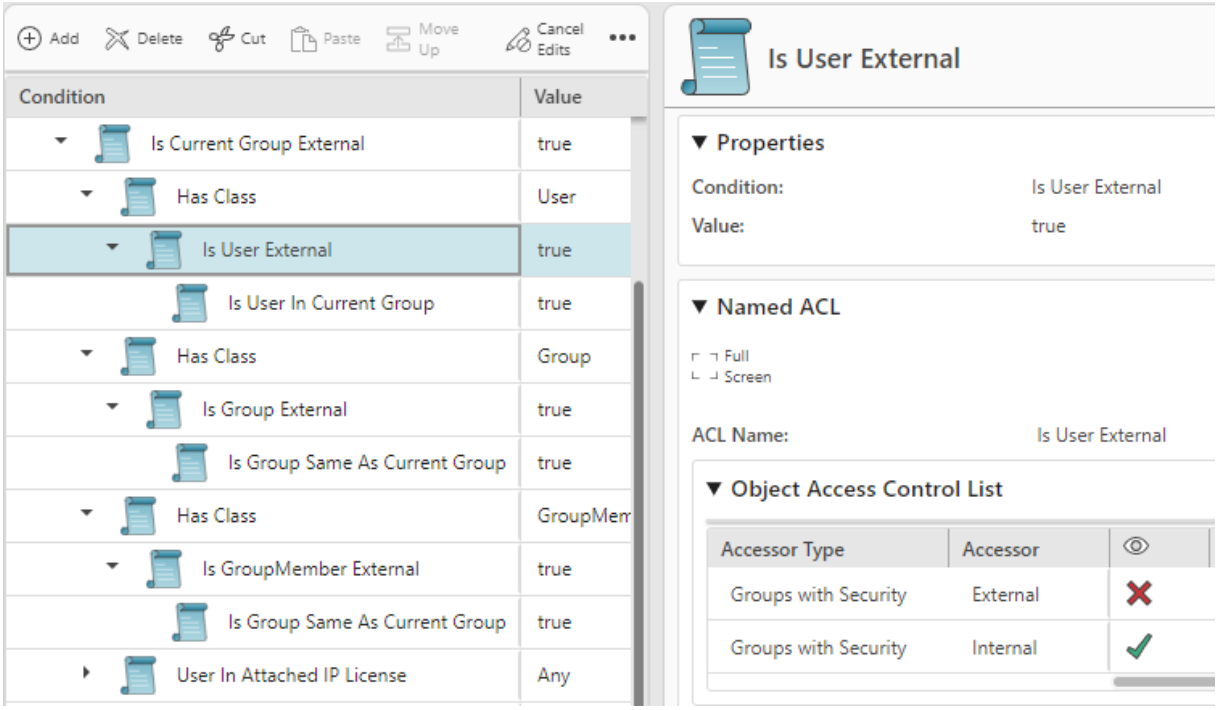
Condition	Value	Accessor Type	Privileges
			• **Subscribe** • **Remote Check-Out** • **Check-In/Check-Out** • **Digitally Sign** • **Void Digital Signature**



5. To prevent a partner from viewing the other assigned partners in an engineering change notice (ECN), create the following ACLs below **Has Bypass(true)**:

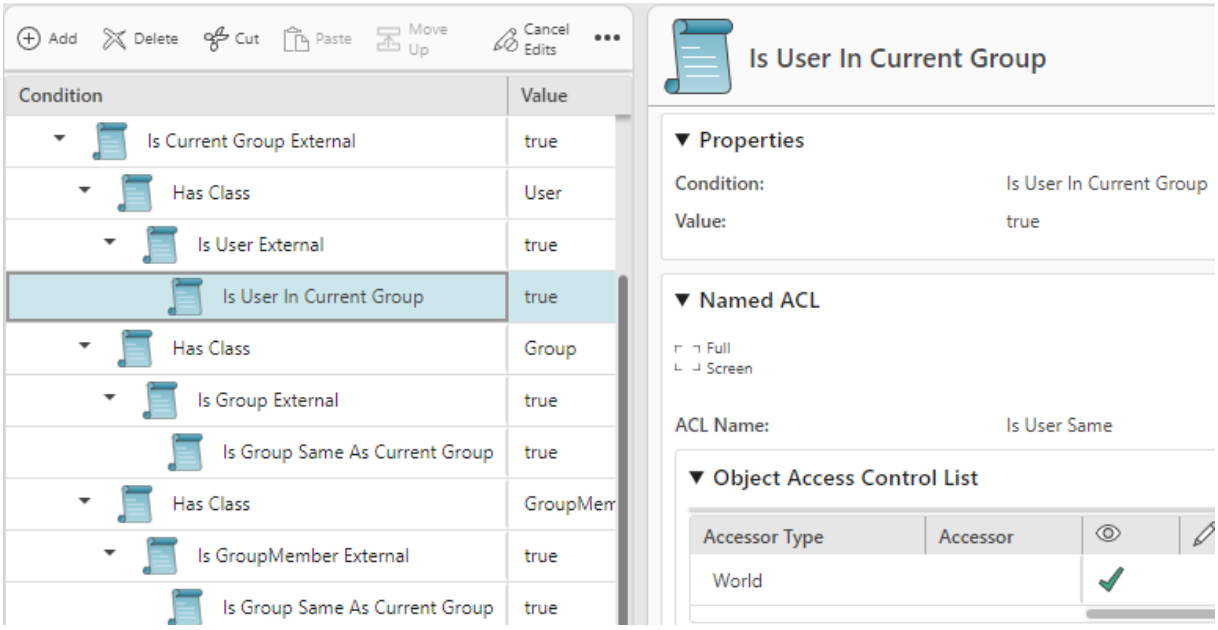
   a. In the **Is Current Group External** > **Has Class (User)** node, create the **Is User External** ACL with the following details:

Condition	Value	Accessor Type	Accessor	Privilege
Is User External	**True**	**Groups with Security**	**External**	Deny the **Read** privilege.
Is User External	**True**	**Groups with Security**	**internal**	Grant the **Read** privilege.



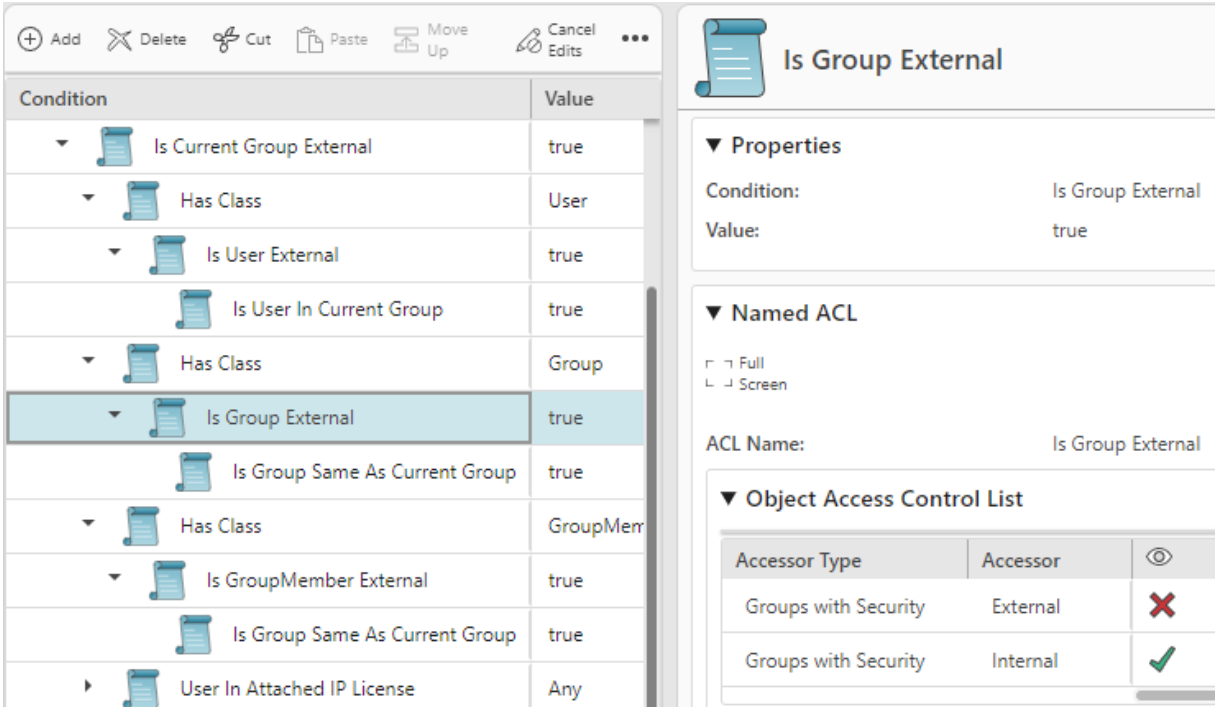
b. In the **Is Current Group External** > **Has Class (User)** > **Is User External** node, create the **Is User Same** ACL with the following details:

Condition	Value	Accessor Type	Accessor	Privilege
Is User In Current Group	**True**	**World**		Grant the **Read** privilege.



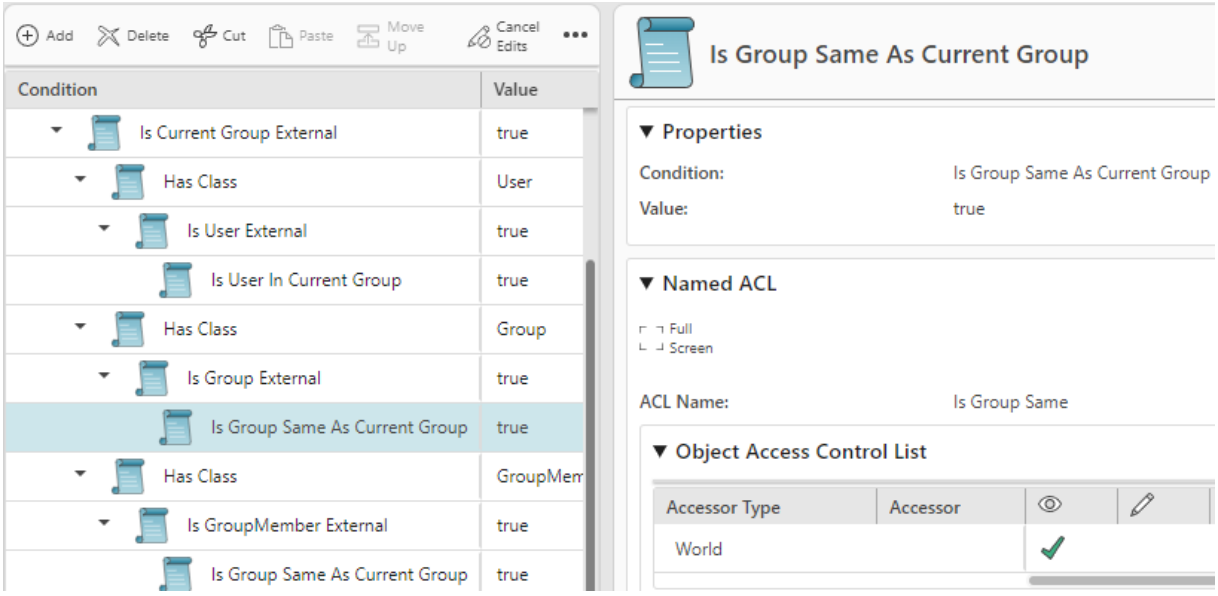
c. In the **Is Current Group External** > **Has Class (Group)** node, create the **Is Group External** ACL with the following details:

Condition	Value	Accessor Type	Accessor	Privilege
Is Group External	**True**	**Groups with Security**	**External**	Deny the **Read** privilege.
Is Group External	**True**	**Groups with Security**	**internal**	Grant the **Read** privilege.



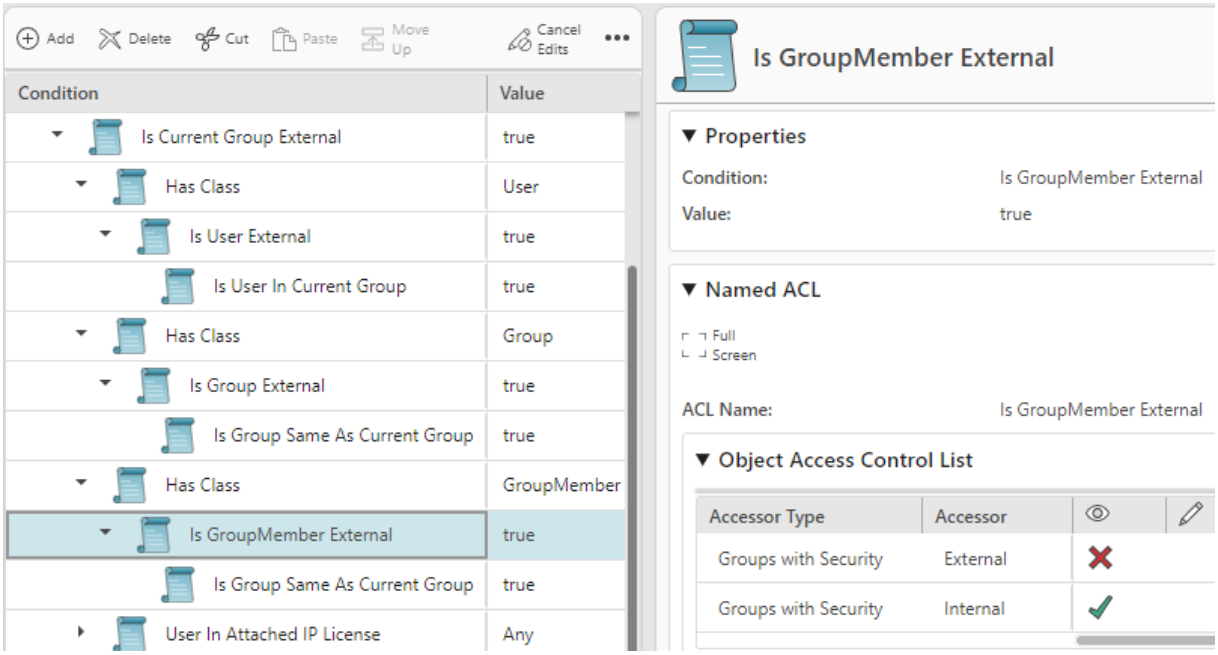
d. In the **Is Current Group External** > **Has Class (Group)** > **Is Group External** node, create the **Is Group Same** ACL with the following details:

Condition	Value	Accessor Type	Accessor	Privilege
Is Group Same As Current Group	**True**	**World**		Grant the **Read** privilege.



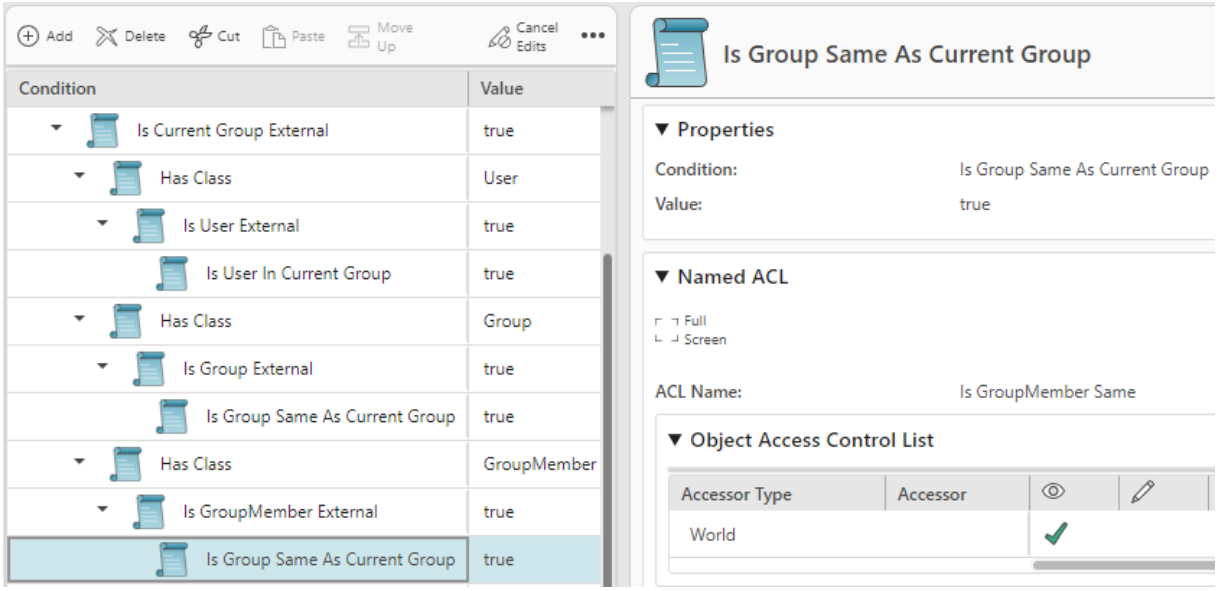
e. In the **Is Current Group External** > **Has Class (GroupMember)** node, create the **Is GroupMember External** ACL with the following details:

Condition	Value	Accessor Type	Accessor	Privilege
Is GroupMember External	**True**	**Groups with Security**	**External**	Deny the **Read** privilege.
Is GroupMember External	**True**	**Groups with Security**	**internal**	Grant the **Read** privilege.



f. In the **Is Current Group External** > **Has Class (GroupMember)** > **Is Group External** node, create the **Is GroupMember Same** ACL with the following details:

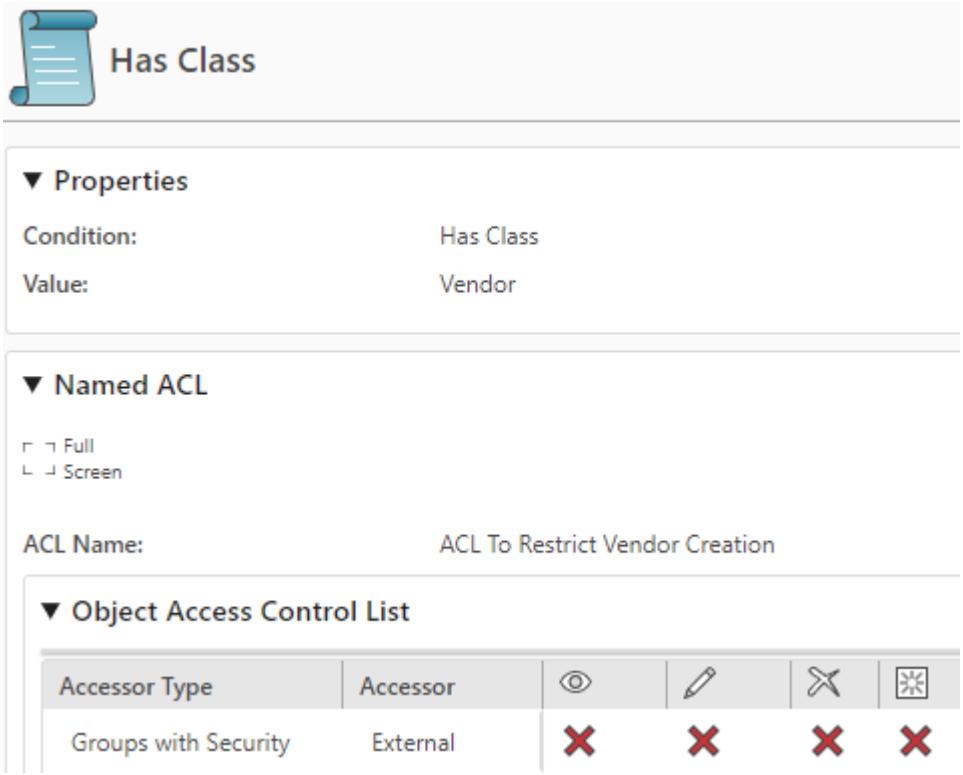
Condition	Value	Accessor Type	Accessor	Privilege
Is Group Same As Current Group	**True**	**World**		Grant the **Read** privilege.



6. To prevent a partner from creating vendors, company contacts, company locations, and partner contracts, create the following ACLs:

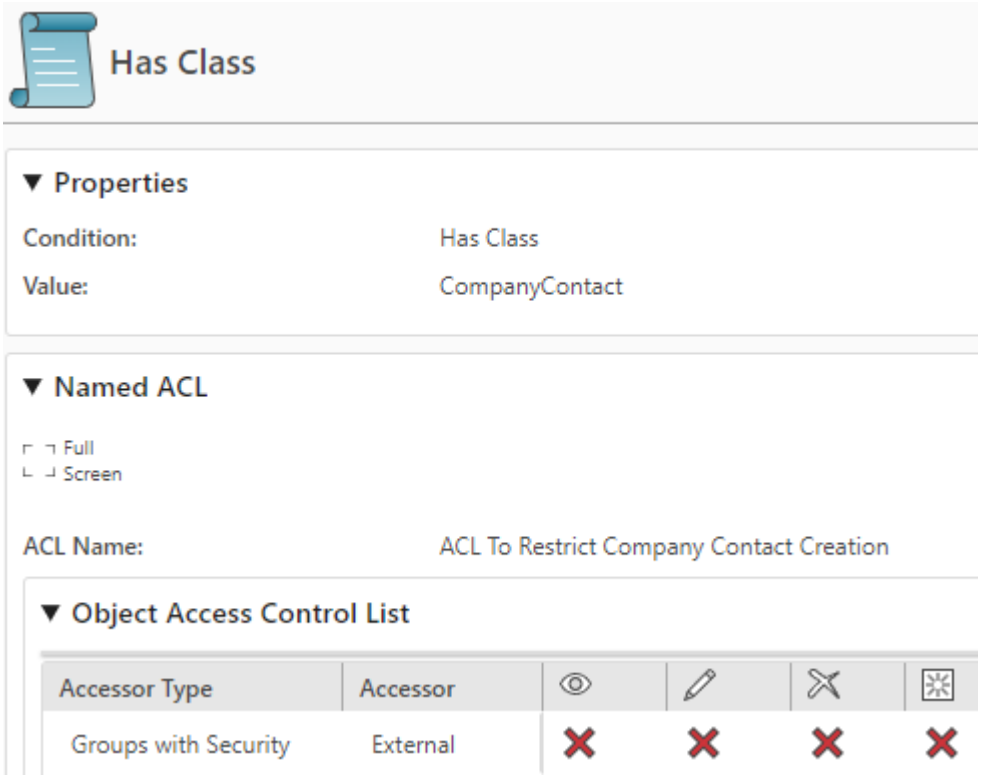
a. To prevent a partner from creating vendors, select the **Has Class (POM Object)** > **Is Current Group External** node from the AM rule tree, and create the **ACL To Restrict Vendor Creation** ACL with the following details:

Condition	Value	Accessor Type	Accessor	Privileges
Has Class	**Vendor**	**Groups with Security**	**External**	Deny these privileges: • **Read** • **Write** • **Delete** • **Create**



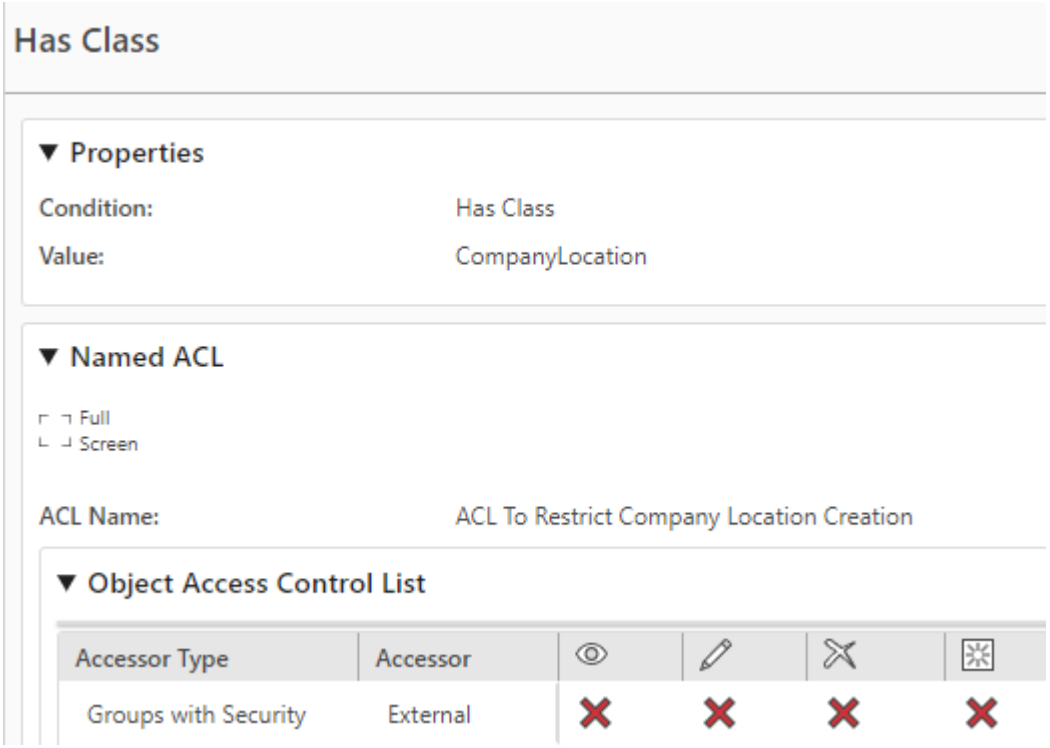
b. To prevent a partner from creating company contacts, select the **Has Class (POM Object)** > **Is Current Group External** node from the AM rule tree, and create the **ACL To Restrict Company Contact Creation** ACL with the following details:

Condition	Value	Accessor Type	Accessor	Privileges
Has Class	**CompanyContact**	**Groups with Security**	**External**	Deny these privileges: • **Read** • **Write** • **Delete** • **Create**



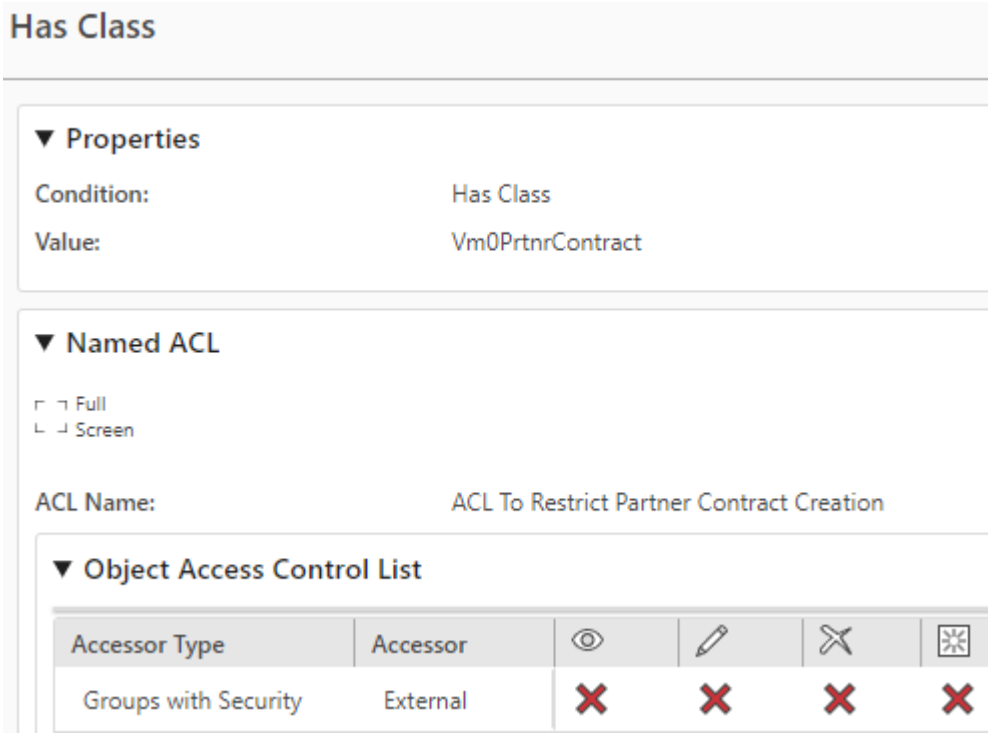
c. To prevent a partner from creating company locations, select the **Has Class (POM Object) > Is Current Group External** node from the AM rule tree, and create the **ACL To Restrict Company Location Creation** ACL with the following details:

Condition	Value	Accessor Type	Accessor	Privileges
Has Class	**CompanyLocation**	**Groups with Security**	**External**	Deny these privileges: • **Read** • **Write** • **Delete** • **Create**



d. To prevent a partner from creating partner contracts, select the **Has Class (POM Object)** > **Is Current Group External** node from the AM rule tree, and create the **ACL To Restrict Partner Contract Creation** ACL with the following details:

Condition	Value	Accessor Type	Accessor	Privileges
Has Class	**VM0PrtnrContract**	**Groups with Security**	**External**	Deny these privileges: • **Read** • **Write** • **Delete** • **Create**

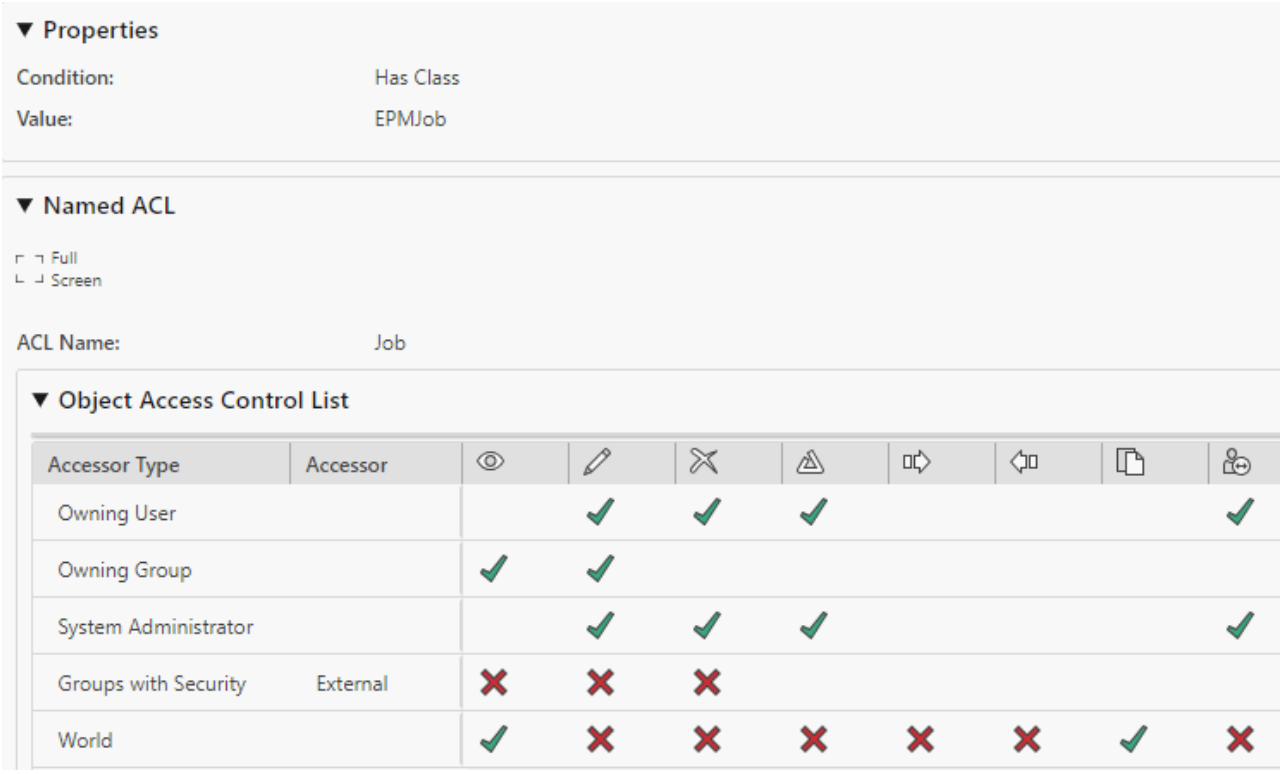


6. Configure a partner's access to only their workflow jobs, workflow tasks, and emails

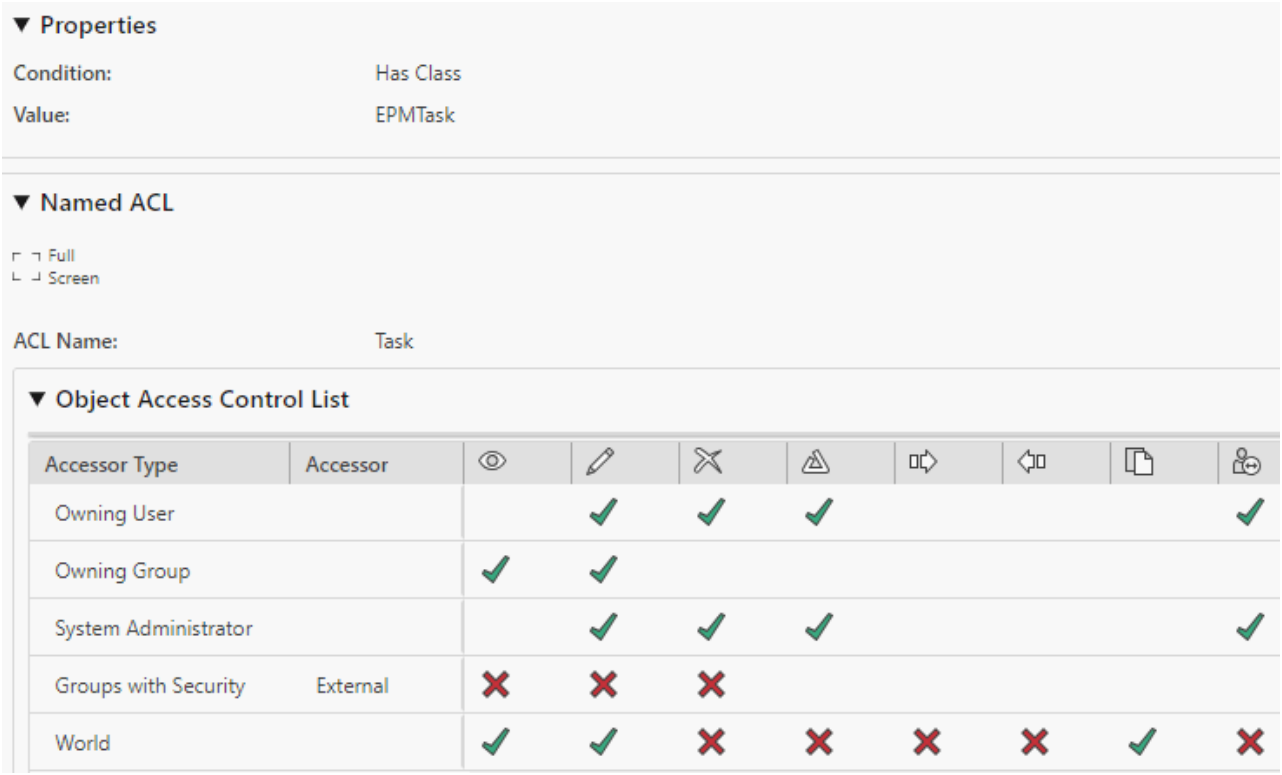
Configure a partner's access to ensure they can only view and interact with their specific workflow jobs, workflow tasks, and emails. This configuration enhances security, protects sensitive information, and ensures that partners focus solely on their assigned responsibilities.

To configure a partner's access to only their assigned objects, configure the following ACLs as follows:

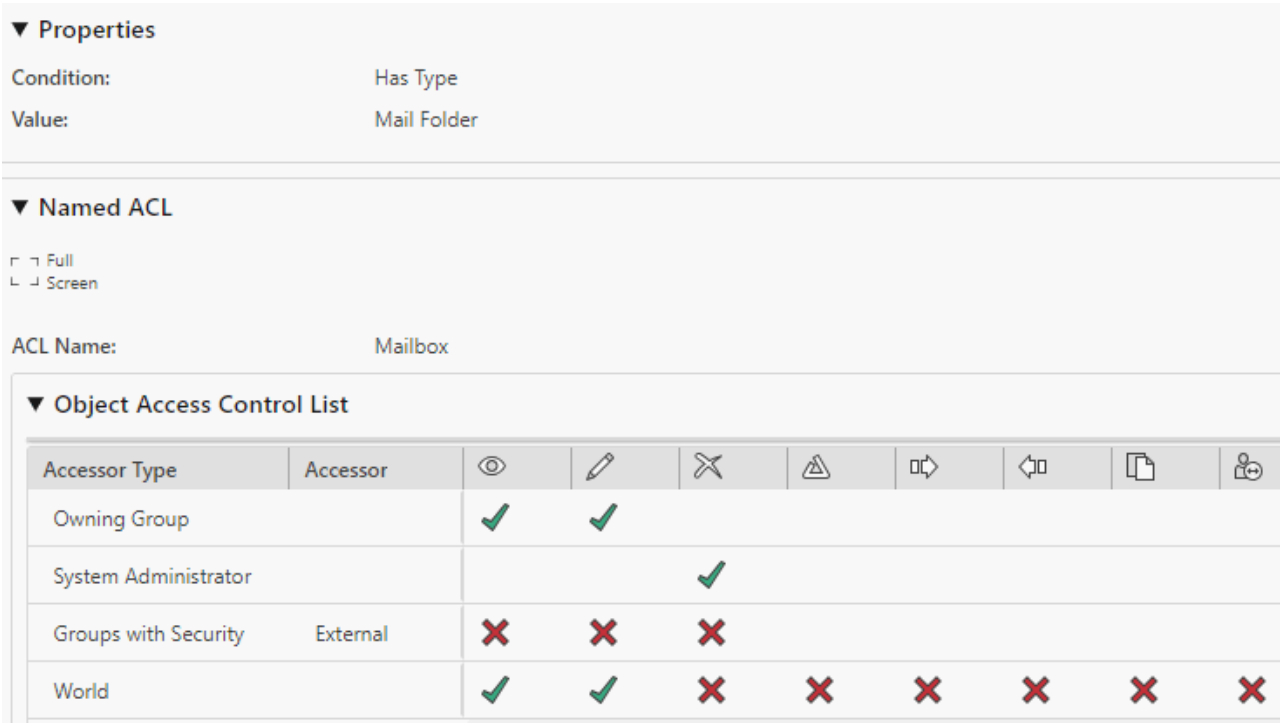
1. To allow partners access to the workflow jobs of their group and to restrict their access to only these jobs, in the **Job** ACL of the **EPMJob** Workflow object, grant **Read** and **Write** privileges to **Owning Group**, and deny **Read**, **Write**, and **Delete** privileges from **Groups with Security** as **External**.



2. To allow partner representatives access to the workflow tasks of their group and to restrict their access to only these tasks, in the **Task** ACL of the **EPMTask** Workflow object, grant **Read** and **Write** privileges to **Owning Group**, and deny **Read**, **Write**, and **Delete** privileges from **Groups with Security** as **External**.



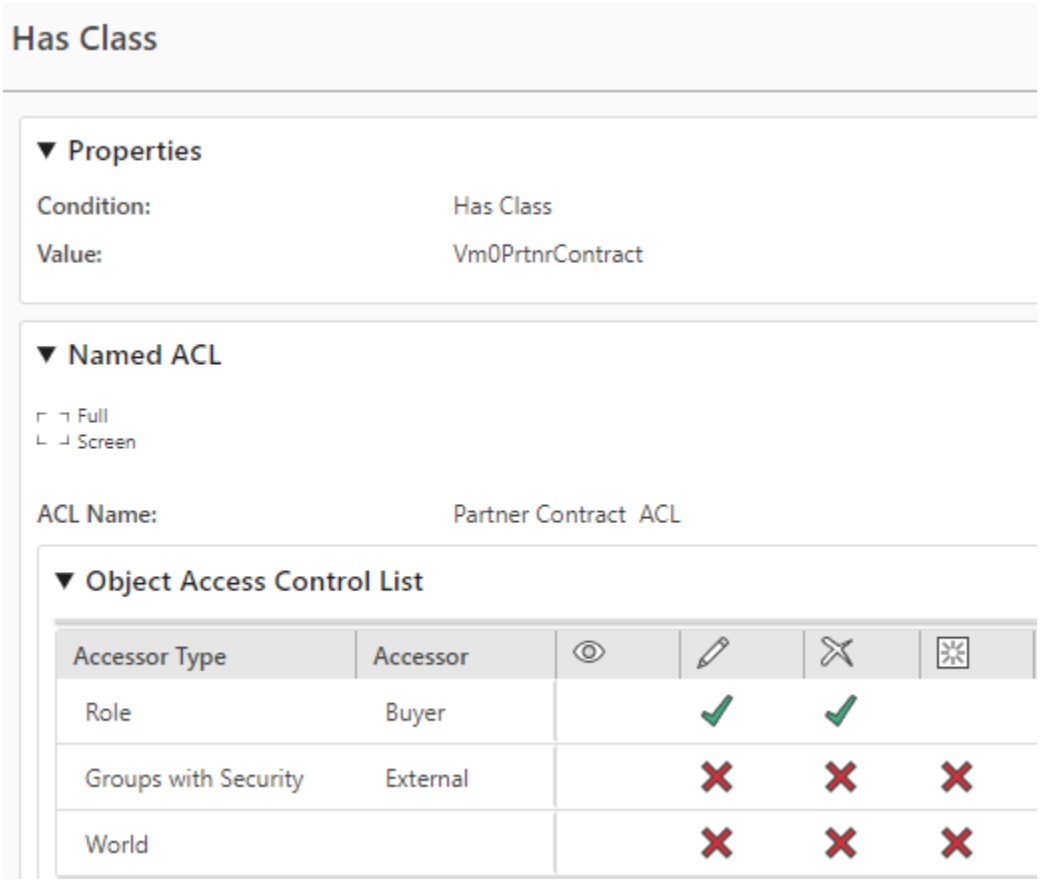
3. To allow partner representatives access to the emails of their group and to restrict their access to only these emails, in the **Mailbox** ACL of the **Mailbox** object, grant **Read** and **Write** privileges to **Owning Group**, and deny **Read**, **Write**, and **Delete** privileges from **Groups with Security** as **External**.



7. Define which users can create, edit, and revise the partner contracts of a vendor

To restrict only users with the role of **Buyer** to create, edit, and revise partner contracts, you must create the **Partner Contract ACL** access rule for the **Partner Contract** object (**VM0PrtnrContract**). To configure this, in Teamcenter Access Manager, select the **Has Class (POM_object)** > **Has Class** node from the Access Manager (AM) rule tree, and create the **Partner Contract ACL** ACL.

Condition	Value	Accessor Type	Accessor	Privileges
Has Class	**VM0PrtnrContract**	**Role**	**Buyer**	Grant these privileges: • **Write** • **Delete**
Has Class	**VM0PrtnrContract**	**Groups with Security**	**External**	Deny these privileges: • **Write** • **Delete** • **Create**
Has Class	**VM0PrtnrContract**	**World**		Deny these privileges: • **Write** • **Delete** • **Create**



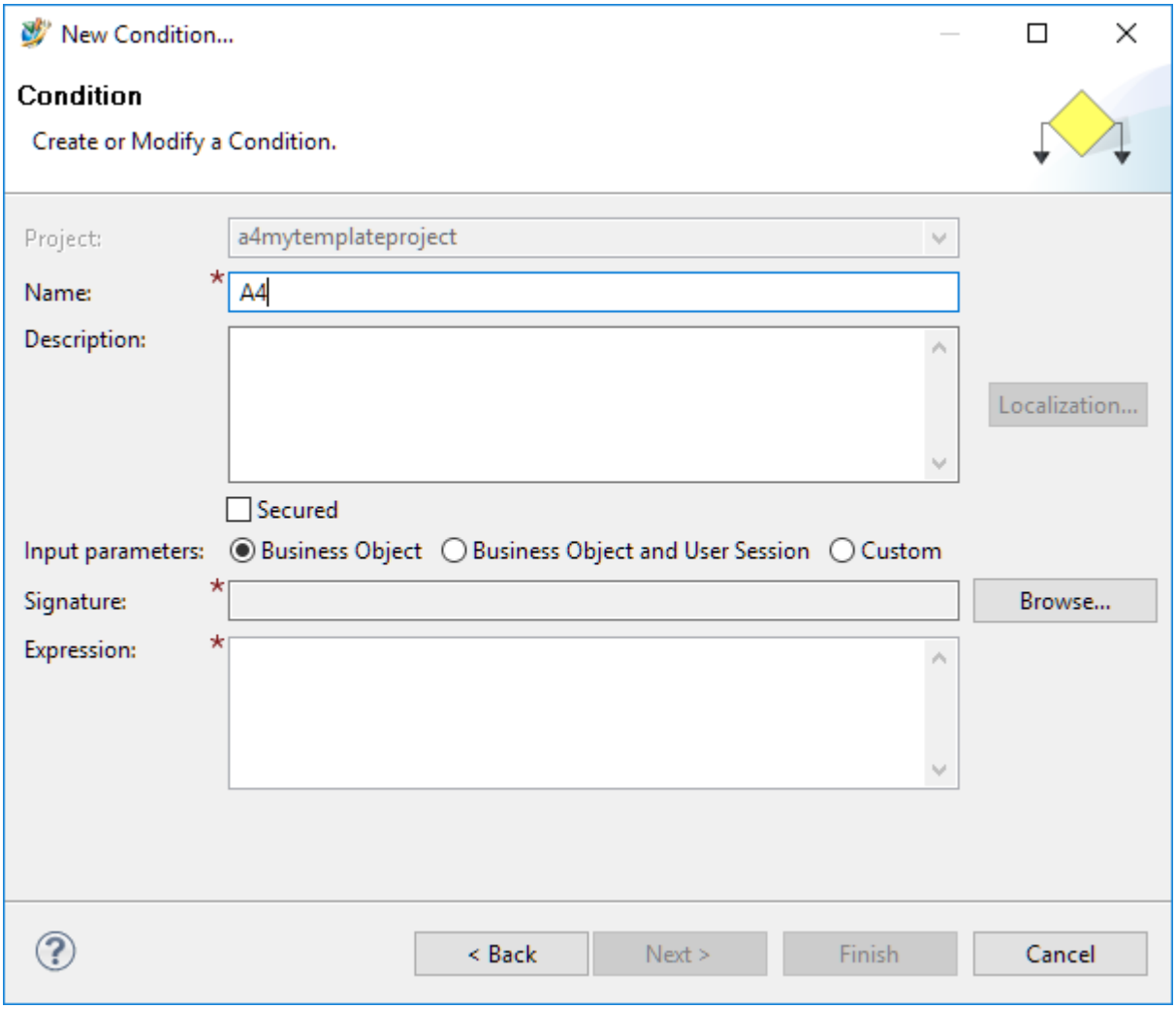
8. Define which users can assign partner contracts to and remove partner contracts from product data

To restrict only users with the role of **Engineer** to assign or remove partner contracts, in BMIDE, create the condition that defines when to display the **Assign** and **Remove** buttons to assign and remove partner contracts, respectively.

1. In the Business Modeler IDE, start the new condition wizard in one of these ways:

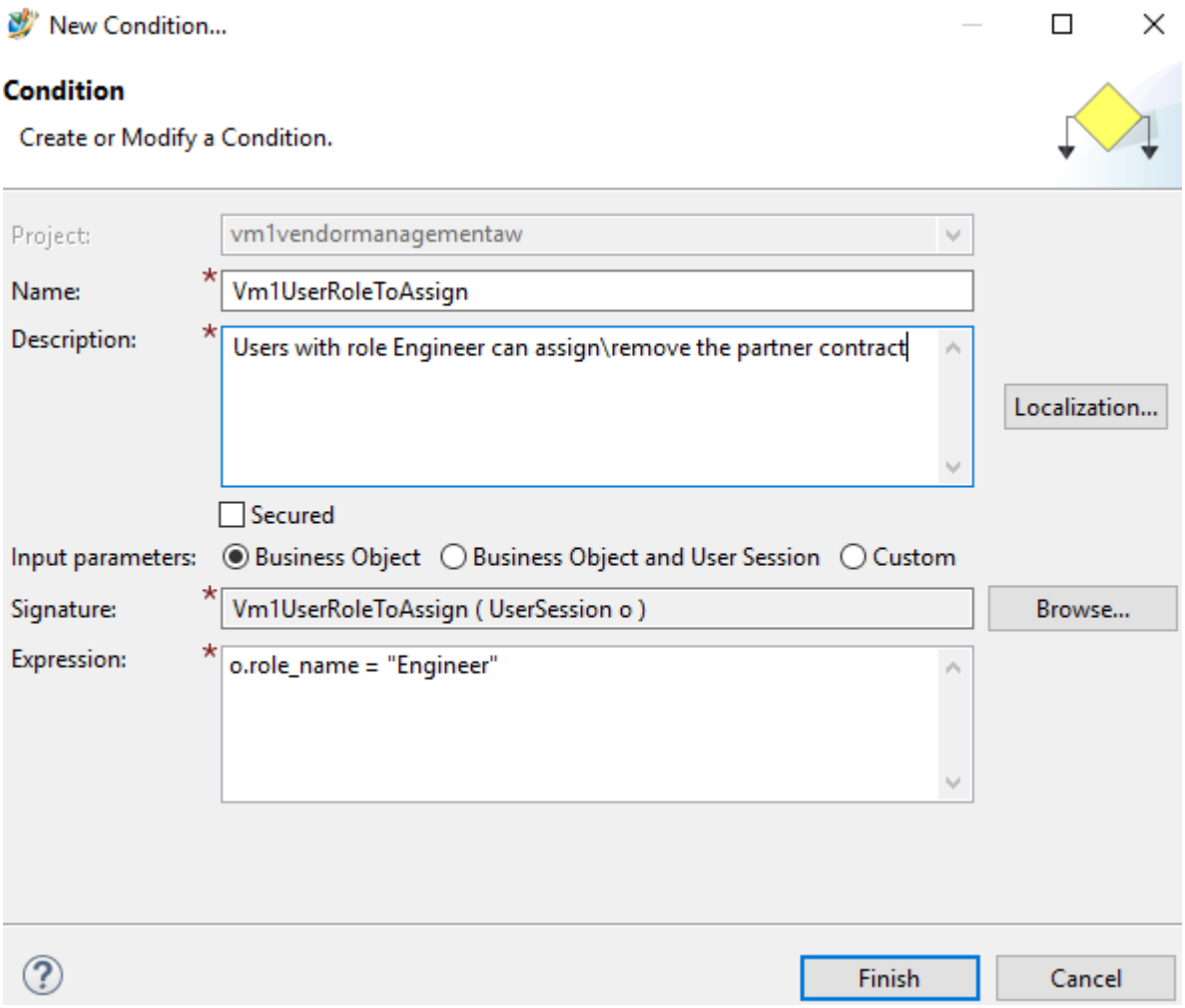
   - On the menu bar, choose **BMIDE→New Model Element**, in the **Wizards** box type **condition**, and click **Next**.

   - Open the **Extensions\Rules** folders, right-click the **Conditions** folder, and choose **New Condition**.



2. In the **Condition** dialog box, define the condition parameters.

For this parameter	Do this
Name	Type **Vm1UserRoleToAssign** as the name that you want to assign to the condition in the database.
Description	Type a description of the condition.
Input parameters	Select **Business Object** to apply the condition to a business object.
Expression	Type `o.role_name = "Engineer"`.

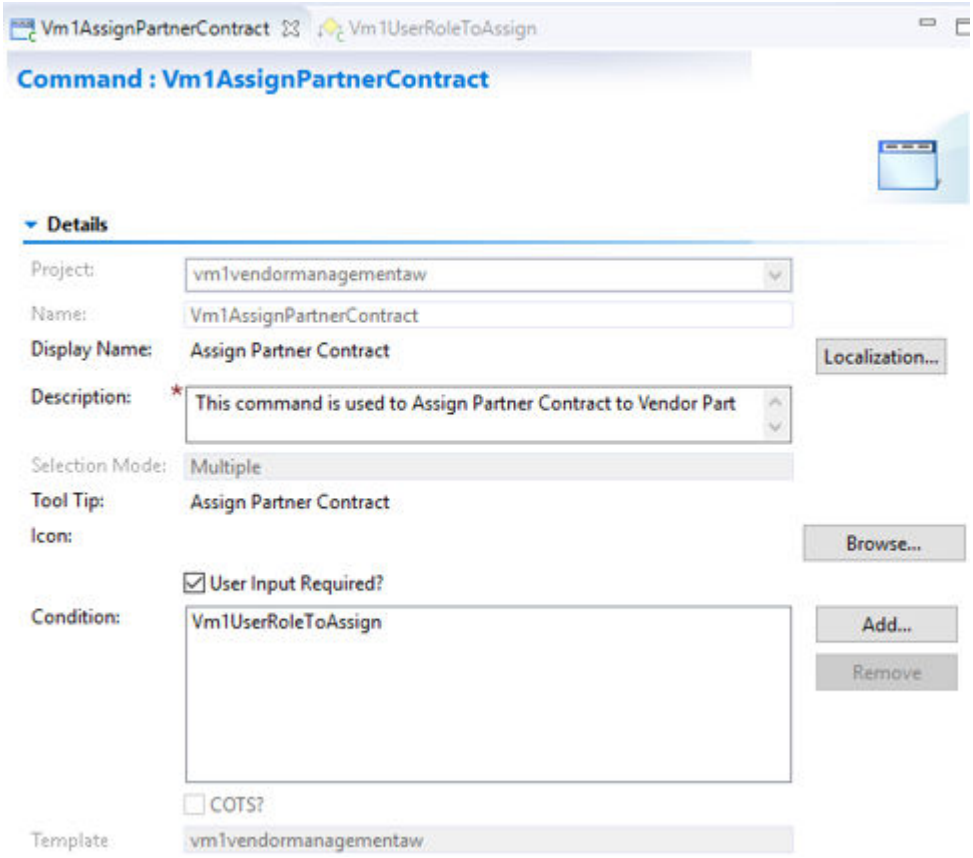


3. Click **Finish**.

4. Open the **VM1AssignPartnerContract** command.

5. In the **Condition** box, select the **isTrue** condition and click **Remove**.

6. Click **Add** and search for and add the **Vm1UserRoleToAssign** condition to the **VM1AssignPartnerContract** command.



7. To save the changes to the data model, choose **BMIDE→Save Data Model**, or click the **Save Data Model** button on the main toolbar.

8. On the menu bar, choose **BMIDE→Deploy Template**. Type the password, click the **Connect** button, and when a connection is established, select the **Generate Server Cache?** check box and click **Finish**.

9. When deployment is done, check the status in the **Console** view.

9. About Partner Connect workflows

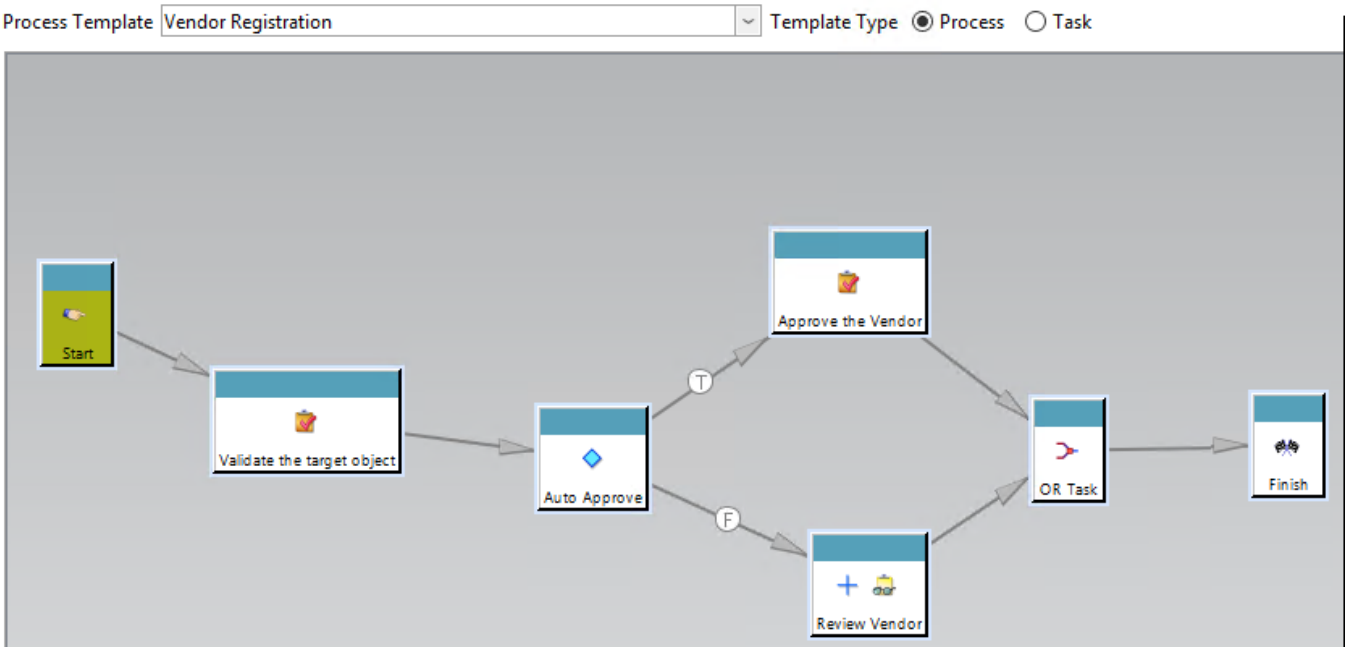
You can customize the Partner Connect workflows or create new ones to meet the specific needs of your organization and partners. However, you must use the existing workflow handlers. If you create new workflows, ensure you update the appropriate preferences to include the names of the new workflows

The following are the default workflows used by Partner Connect:

- Vendor Registration
- Partner Contract Qualification
- Retire Partner Contract

Vendor Registration workflow

This workflow is used for the individual tasks and the sequence required to register a vendor. By default, the **Vendor_default_workflow_template** preference is set to the name of this workflow. When you select a vendor and click **Manage** > **Submit to Workflow**, the default workflow from the **Vendor_default_workflow_template** preference is selected.



The following table describes the tasks and action handlers in the **Vendor Registration** workflow:

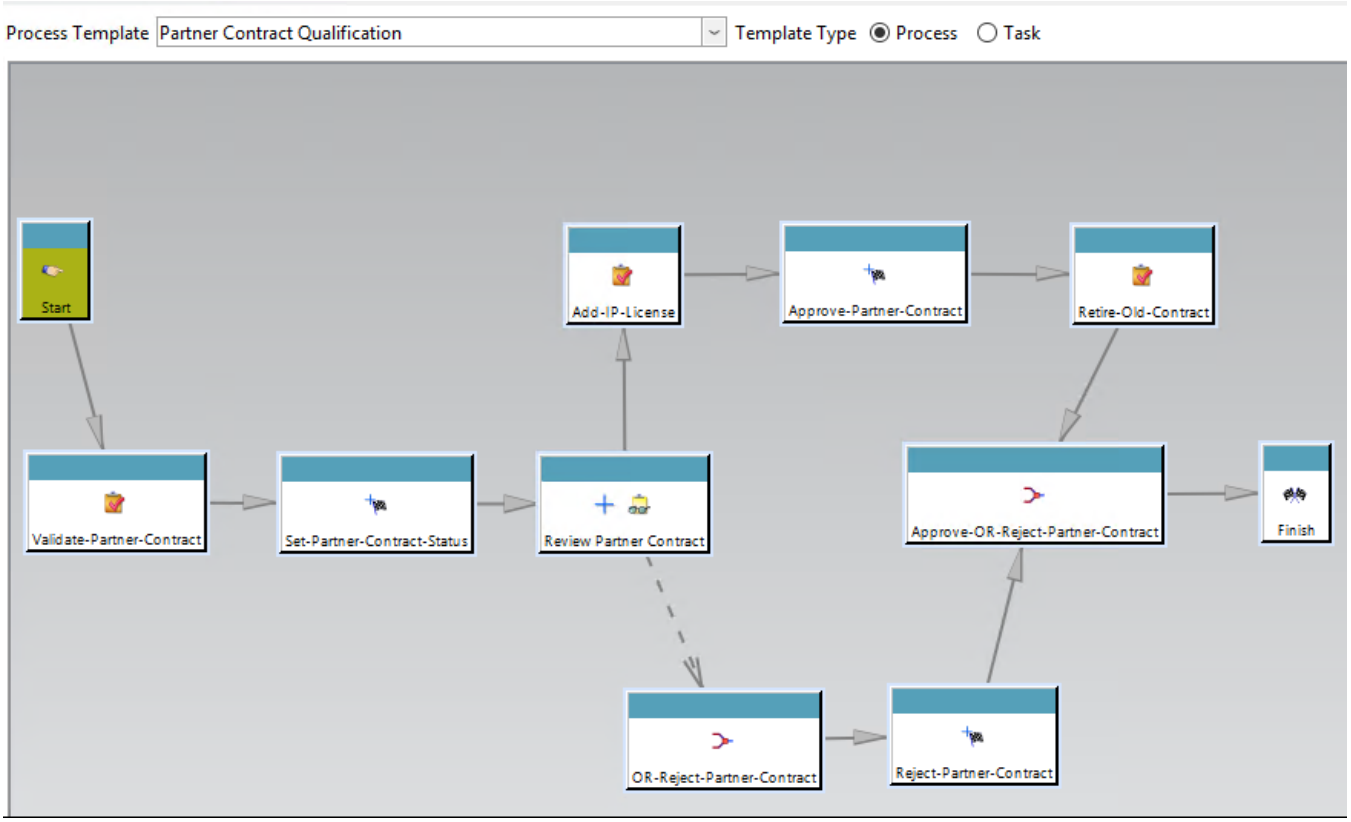
Task: Action handler	Description	Arguments	Placement	Restrictions
VM-Validate-Vendor	Validates the attached target object. A valid target object is a vendor without the registration status of **Approved**.	None	This action handler is attached to the **Start** action of the **Validate the target object** task.	The **Target** object for this action handler is **Vendor**.
VM-is-auto-approve-vendor-pref-on	If the **VM_auto_approve_vendor_registration** preference is **True**, this action handler sets the task result to **True**.	None	This action handler is attached to the **Start** action of the **Auto Approve** task.	
VM-auto-approve-vendor-registration	Approves the vendor. The registration status of the target **Vendor** object is set to **Approved**.	None	This action handler is attached to the **Start** action of **Approve the Vendor**.	
VM-Review-Vendor	Reviews the vendor attached as a target object to the workflow. The registration status of the target **Vendor** object is set to **Approved** if the reviewers approve the target object. The registration status of the target **Vendor** object is set to **Rejected** if the reviewers reject the target object.	None	This action handler is attached to the **Perform** action of the **perform-signoffs** handler of the **Review Vendor** review task.	

Partner Contract Qualification workflow

This workflow is used for the individual tasks and the sequence required to approve or reject a partner contract. By default, the **Vm0PrtnrContractRevision_default_workflow_template** preference is set to the name of this workflow. When you select a partner contract revision and click **Manage** > **Submit**

to Workflow, the default workflow from the **Vm0PrtnrContractRevision_default_workflow_template** preference is selected.

> Note:
>
> These workflow handlers are part of the Partner Connect solution and require the Partner Connect License to use them.



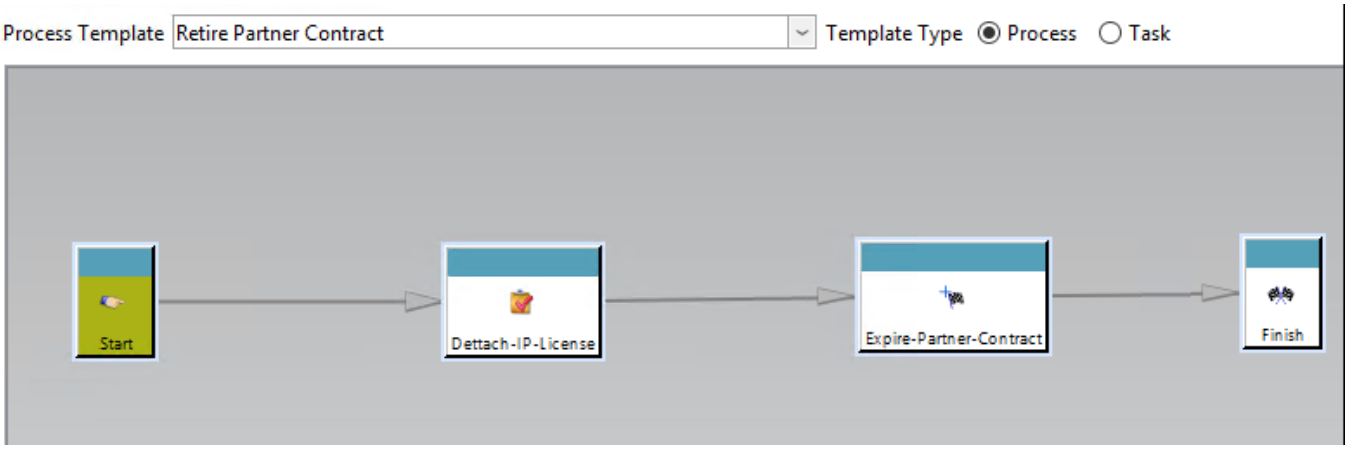
The following table describes the tasks and action handlers in the **Partner Contract Qualification** workflow:

Task: Action handler	Description	Arguments	Placement	Restrictions
VM-Validate-PartnerContract	Validates the attached target object. A valid target object is a partner contract revision with a release status of **Created** and its associated vendor's	None	This action handler is attached to the **Start** action of the **Validate-Partner-Contract** task.	The **Target** object for this action handler is **Vendor**.

Task: Action handler	Description	Arguments	Placement	Restrictions
	registration status is **Approved**.			
VM-Add-PartnerContract-IPLicense	Creates a new IP license for the partner contract revision if the license does not exist and attaches the IP license to the partner contract's **Item**, **Revision**, **Contacts**, **Locations**, and **Vendor**. Creates the **Vm0UsesIPLicense** relation between the partner contract **Item** and IP license.	None	This action handler is attached to the **Start** action of the **Add-IP-License** task.	
VM-Retire-Old-PartnerContract	Initiates the **Retire Partner Contract** workflow for all previous partner contract revisions with the release status of **Approved**.	None	This action handler is attached to the **Start** action of **Retire-Old-Contract**.	

This workflow is used for the individual tasks and the sequence required to discontinue a partner contract by retiring it.

> Note:
>
> These workflow handlers are part of the Partner Connect solution and require the Partner Connect License to use them.



Retire Partner Contract workflow

The following table describes the tasks and action handlers in the **Retire Partner Contract** workflow:

Task: Action handler	Description	Arguments	Placement	Restrictions
VM-Detach-PartnerContract-IPLicense	Removes the IP license from the partner users of a partner contract revision. The IP license is removed if no partner contract revision exists with the state of **Created**, **Approved**, or **Approval Pending**. After removing the IP license, the partner users cannot access the objects associated with the IP license. If a partner contract revision exists with the state of **Created**, **Approved**, or **Approval Pending**, the handler does not remove the IP license of the partner users. If a target partner contract revision is already part of the **Partner Contract**	None	This action handler is attached to the **Start** action of the **Detach-IP-License** task.	The **Target** object for this action handler is **Partner Contract Revision**.

Task: Action handler	Description	Arguments	Placement	Restrictions
	Qualification workflow, the handler aborts the **Partner Contract Qualification** workflow.			

10. Ensuring partner access to all related datasets

In an OEM Teamcenter, when a configuration manager assigns a part, sub-assembly, or assembly to a partner contract, Partner Connect automatically creates a dedicated intellectual property (IP) license. This IP license is attached to the assigned part, sub-assembly, or assembly, and the associated partner is also assigned to this IP license. This setup grants the partner access to the specified part, sub-assembly, or assembly.

In some cases, partners may not have access to all necessary datasets related to the shared part, sub-assembly, or assembly. This issue arises because IP licenses are propagated from items to datasets using propagation rules, and the default propagation rules might not cover all related datasets. For example, the default rules do not propagate the IP license to a **DirectModel (JT)** dataset attached to an **ItemRevision** with the **IMAN_Rendering** relation. To ensure partner access to all related datasets, you must configure custom propagation rules for any datasets not covered by the default rules.

11. Configure how to manually assign partner contracts to vendor parts

By default, when a Configuration Buyer assigns a partner contract to an assembly, the partner contract is assigned to its child items, subassemblies, and commercial parts. However, you, as the administrator, can configure whether the Configuration Buyer must manually assign the partner contract to the vendor parts in the commercial part. To do this, set the **CM_AllowVendorPartContractAssignment** preference to **False**.

Now, after the Configuration Buyer assigns a partner contract to a commercial part, they must open the partner contract of the commercial part, and manually assign the required vendor parts to the partner contract.

12. Delete obsolete partner contracts

A retired partner contract is one that is no longer in use. You can delete all such retired contracts from the database by running the **vm_delete_obsolete_contract_revisions** utility. You require administrator privileges to run this utility.

> Note:
>
> You must remove the partner contract assignment from the part or assembly in a product and submit it to the **Retire Partner Contract** workflow before running this utility.

While running this utility, you can specify the ID of the vendor whose obsolete partner contracts must be deleted or the specific ID of the obsolete partner contract to be deleted. If you do not specify a vendor ID or a partner contract ID, the utility deletes all obsolete partner contracts.

> Caution:
>
> After the partner contracts are deleted, they cannot be recovered. Therefore, if you prefer to maintain a history of partner contracts, you must run this utility only after due diligence.

Run the **vm_delete_obsolete_contract_revisions** utility with the following arguments:

vm_delete_obsolete_contract_revisions -u=*user-id* {**-p=***password*} [**-g=***group*] [**-v=***vendor ID*] [**-c=***partner contract ID*] [**-output=***path to the log file*]

-u

Specifies the user ID.

This is generally a user with administration privileges.

-p

Specifies the password.

-g

Specifies the group associated with the user.

If used without a value, the user's default group is considered.

-v

Specifies the ID of the vendor whose obsolete partner contracts must be deleted.

-c

Specifies the ID of the obsolete partner contract to be deleted.

-output

Specifies the absolute path to the generated log file. If no path is specified, the report is generated in the current working directory of the application.

-h

Displays the help for this utility.