

Creating and Activating a DAC Domain Policy

The procedure for creating and activating a DAC policy is described in detail in the following sections. As described earlier it is a bottom-up process.

- **Creating a Query Definition:**The Query Definition specifies the paths from the Root item through one or more Relationships to Leaf Items. It is accessed from the TOC folder *Administration/Configuration*.
- **Creating a Derived Relationship Family (DRF):**A Derived Relationship Family is an ItemType that contains a Query Definition and one or more relationship paths derived from this query known as Derived Relationships. It can also be found in the TOC folder *Administration/Configuration*.
- **Defining Derived Relationships:**Derived Relationships are related items on the “Derived Relationships” tab on the DRF. Each has a Name identifier and start/end points of the DAC Subdomain in the columns “Departure Item Type [...]” and “Destination Item Type [...]”.
- **Persisting DAC Access Control Table (*unidr_Relationships* table):**Once a Derived Relationship Family and its subdomain definitions are completed, the ‘unidr_Relationships’ table is populated with data to track all instances of the specified subdomain in the database. This may take quite a bit of time for large databases.
- **Creating a DAC Definition (Policy):**Once the Derived Relationship Family is persisted, we create a DAC Definition that applies DAC Access Rules for each DRF subdomain conditionally based on root item permissions and leaf item life cycle states.

An active DAC Definition automatically creates DAC Domains for all instances of the root item. Access is based on the Team attached to the root item and governed by the permissions set in the DAC Definition. DAC access is calculated based on the table of *unidr_Relationship* items which are automatically synchronized with the results of the Derived Relationship Family’s Query Definition.

Prerequisites

Before using Domain Access Control, you must make sure that the Agent Service is running. Ensure that the Agent Service Conversion Processing Cycles are configured in the **conversion.config** file, with the appropriate DB and User filled in:



```
<ProcessingCycles>  
  <cycle DB="{db}" User="{user}" Interval="20000" MaxBatch="5"/>  
</ProcessingCycles>
```

Restart the Agent Service after making changes to the config file.

The Agent Service also executes Conversion Tasks generated during the persisting process to create the initial unidr_Relationship items.

Important

Examples in this document use the **Project ItemType**. This **ItemType** is now an Aras Package that is installed separately. If you do not have this add-on installed, use Aras Update to install it before proceeding.



Aras Update

aras UPDATE

?

Online

Local

Aras Packages available online

Listed below are all Aras packages available online. Select a desired package version, then pick Download to download for later installation, or Install to immediately download and install. Aras subscriber-only features such as Service Packs are not listed here, to install them you must first download them from the Aras FTP site and then use the Local view on the left to add the package reference for installation.

Aras Technical Documentation

Aras Process Quality Binding

Aras Project Management

Enables organizations to manage new product development and engineer-to-order processes for complex projects that require global collaboration. By linking deliverables to project tasks, completion status and control are significantly enhanced. Projects can then combined together as Programs to manage the overall product portfolio.

[More information](#)

11.0R2

11.0R3

11.0R4

11.0R5

12.0R1

Install

Download

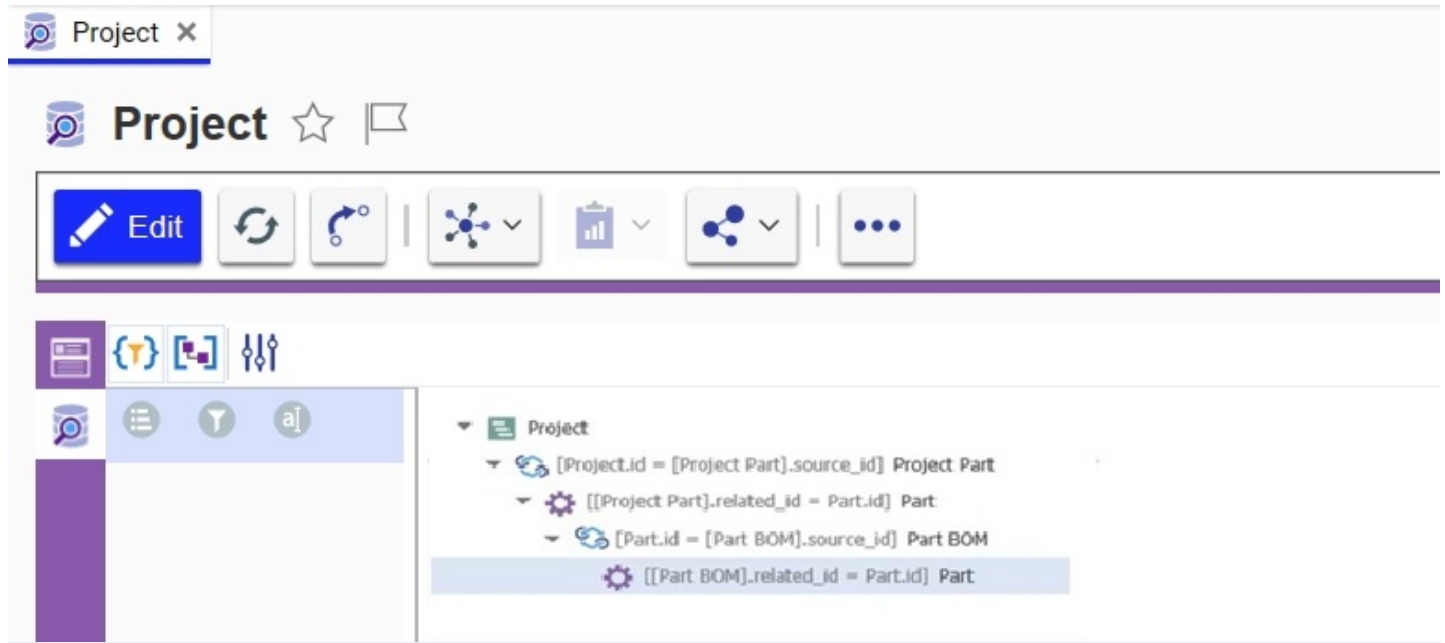
Aras Quality Management System

Creating a Query Definition

©2026 Aras Corporation All Copyrights Reserved

3

You need to create a Query Definition in order to create a Derived Relationship (DR) Family. The DR Family Definition maps DR Family elements and non-root Query Items contained in the Query Definition, which are in turn used to create DAC Subdomain Policies. For information on how to create a Query Definition, refer to the *Query Builder Guide*. In this guide we will use the following example Query Definition:



The screenshot shows the 'Project' Query Definition in the Aras Innovator interface. The top bar includes a 'Project' tab, a star icon, and a flag icon. Below this is a toolbar with buttons for 'Edit', refresh, undo, redo, and other actions. The main area displays a tree structure of the Query Definition:

- Project
 - [Project.id = [Project Part].source_id] Project Part
 - [[Project Part].related_id = Part.id] Part
 - [Part.id = [Part BOM].source_id] Part BOM
 - [[Part BOM].related_id = Part.id] Part

Below the tree structure, there is an 'Important' section with the following text:

Important
The root **Query Item** (Project in this example) should not be reused in a **Query Definition** that will be used by a **DR Family**.

Derived Relationship Families

A Derived Relationship (DR) Family describes the derived relationships between Aras Innovator items. Each Derived Relationship in a DR Family has its own destination ItemType and name. An individual DR is referred to as a DR Family Element. All DR Family elements share the same departure ItemType as determined by the Query Definition.

Creating a Derived Relationship Family

Use the following procedure:

1. In the Navigation Panel, click **Administration>Configuration>Derived Relationship Families**. The following menu appears:



2. Click **Create New Derived Relationship Family**. A screen like the following appears:

A screenshot of a web application form titled 'Derived Relationship Families'. The form has a menu bar with 'File', 'Edit', 'Views', 'Actions', 'Reports', 'Tools', and 'Help'. Below the menu bar is a toolbar with various icons. The form fields include: 'Name' (text input), 'State' (text input), 'Query Definition' (text input with a dropdown arrow), 'Departure Item Type' (text input), and 'Description' (text area). On the right side, there is a 'Persistence Status' section with 'Departure Items Persisted' (text input), 'Departure Items Total Count' (text input), 'Start' (date/time picker), and 'End' (date/time picker). Below these is an 'Errors' section with a text area. At the bottom, there is a 'Derived Relationships' section with a dropdown menu showing 'Derived Relationships' and a star icon. Below this are several icons and a 'Hidden' dropdown. At the very bottom, there is a table with columns: 'Name', 'Departure It...', 'Destination It...', and 'Description'.

3. Enter a name for the derived relationship in the **Name** field.
4. Click the ellipses in the **Query Definition** field and select the Query Definition from the dropdown list. Once you click save you may not change the assigned Query Definition.
5. Save the Derived Relationship Family. Its state is now Preliminary. The Query Definition is claimed and the Departure ItemType is updated based on the Query Definition.



6. Click the **New Derived Relationship** icon in the **Derived Relationships** section to create a Derived Relationship.

Name
Part Related to Project

State
Preliminary

Query Definition
[part Related to Project](#)

Departure Item Type
[Project](#)

Description

Persistence Status

Departure Items Persisted

Departure Items Total Count

Start

End

Errors

Derived Relationships

Derived Relationships ☆

Hidden

Name	Departure It...	Destination It...	Description
	Project		

The Root Item Type name (in this case Project) appears automatically. It is not editable.

7. Enter the Derived Relationship name in the **Name** field of the Derived Relationship tab.
8. Click the **Destination Item Type** field. The ItemType search dialog box appears.

Select Items



ItemTypes ▾

Simple ▾

	Name	Singular Label	Plural Label	Versio...	Depen...	Relatio...	C.	Use Src ...	Description ...	Help Url	H
▼	p*										
	Part	Part	Parts	☒	☐	☐	☐	☐		mergedProje...	^
	Part Alternate	Part Alternate	Alternates	☐	☐	☒	☐	☒			
	Part AML	Approved Manuf...	AML	☐	☐	☒	☐	☒			
	Part BOM	Part BOM	BOMs	☐	☐	☒	☐	☒			
	Part CAD			☐	☐	☒	☐	☒			
	Part Changes	Part Changes	Part Changes	☐	☐	☒	☐	☒			
	Part Document	Part Document	Documents	☐	☐	☒	☐	☒			
	Part Goal	Part Goal	Goals	☐	☐	☒	☐	☒			
	Part MultiLevel B...			☐	☐	☒	☐	☒			
	PDFWMConfigur...	PDF Collection	PDF Collections	☐	☐	☐	☐	☐			▼

< Prev Next > Page: 1 ... 25

OK Cancel

The dialog displays the ItemTypes associated with the Query Definition.

9. Select the Destination **ItemType**. In this example we select **Part**.

10. Repeat this process and add all necessary Derived Relationships.



Part Related to ... x

File Edit Views Actions Reports Tools Help

Name

Part Related to Project

State

Preliminary

Query Definition

part Related to Project

Departure Item Type

Project

Description

Persistence Status

Departure Items Persisted

Departure Items Total Count

Start

End

Errors

Derived Relationships

Derived Relationships

Hidden

Name	Departure Item Type [...]	Destination Item Type [...]	Description
Part Related to Project	Project		



11. Click the **Query Definition** icon in the sidebar to view the query definition.

Part Related to ... x

File Edit Views Actions Reports Tools Help

Name

Part Related to Project

State

Preliminary

Query Definition

[part Related to Project](#)

Departure Item Type

Project

Description

Persistence Status

Departure Items Persisted

Departure Items Total Count

Start

End

Errors

Derived Relationships

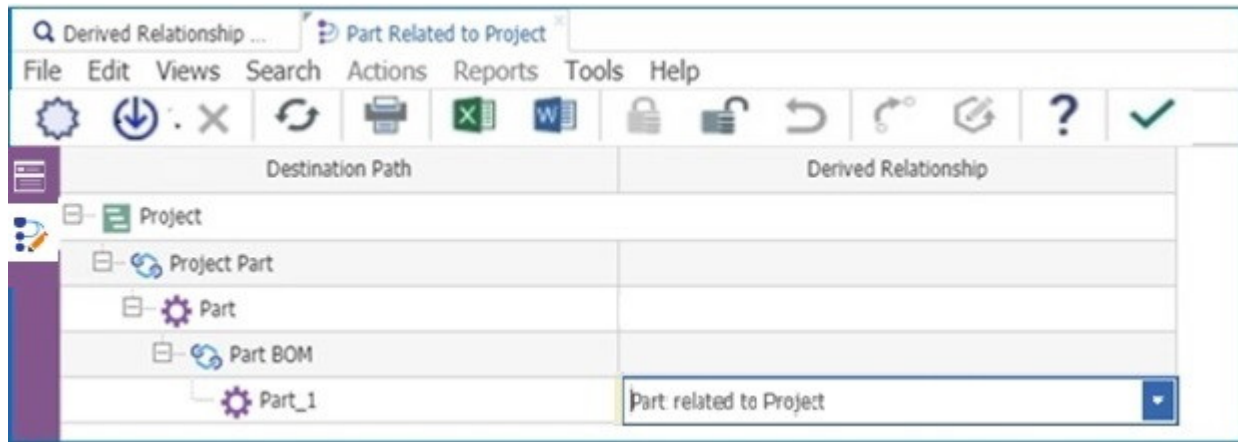
Derived Relationships

Hidden

Name	Departure Item Type [...]	Destination Item Type [...]	Description
Part Related to Project	Project		

12. Click on a **Derived Relationship** cell to map the Query Item to the Derived Relationship that you created on the previous screen.

13. Select the corresponding DR from the dropdown list.



Important

You can only specify Derived Relationships which have been created on the previous screen.

14. Once all Derived Relationships are mapped click **save** and then **unclaim** the DR Family.
15. From the **Actions Menu**, select the **Activate** action to promote the DR Family to the **Active** state. The status of the Derived Relationship changes to **Persisting** and the initial persistence of the DR Family begins.

Conversion Tasks are created which will persist the DRs for each QD root item in the database. By default, 10 Departure Items (Can be configured by modifying the `dr_DepartureItemsPerConvTask` system variable) are processed per Conversion Task. When the last Conversion Task is finished, the DR Family is activated

Important

During this process you should not change data in any way that may affect the outcome of the DR persistence operation. The initial DR persistence process may take some time depending on the size of the database.

Information relevant to the Persistence process can be viewed in the Persistence section of the Derived Relationship Family Form. Departure Items Persisted, % displays the percent of Departure Items that have been persisted. Departure Items Total Count provides the total number of Departure Items in the system. If there are errors during the persistence process, the Errors box is populated with information pertaining to the errors.

You can only deactivate a DR Family if it is not in active use. To deactivate a DR Family, select the Deactivate action from the Actions menu. Upon deactivation of a DR Family all the persisted Family DRs are deleted. The observers that have been activated for DR Family “participating ItemTypes” are deactivated. Deactivated DR Family can be deactivated and then activated again.

Part Related to ...

File Edit Views Actions Reports Tools Help

Name
Part Related to Project

State
Preliminary

Query Definition
part Related to Project

Departure Item Type
[Project](#)

Description

Persistence Status

Departure Items Persisted	Errors
Departure Items Total Count	
Start	
End	

Derived Relationships

Derived Relationships ▾ ☆

+ - 🔍 ⌕ Hidden ▾ 📷 🖨️ 🔗 ▾

Name	Departure Item Type [...]	Destination Item Type [...]	Description
Part Related to Project	Project		

Derived Relationship Family Remediation

DRF remediation is a process of resolving denormalized data when the denormalized data gets out of sync with the primary data. This process can be run either manually from the DRF action menu in Aras Innovator, or automatically at certain times of the day. Both manual and automatic DRF remediation assumes that the primary DRF data is not modified by end users during the duration of the DRF remediation process. (If this assumption is incorrect, DRF remediation process cannot be run.) Both manual and automatic remediation record the result of the undertaken remedial action in



the Aras Innovator log files, and in the Conversion Tasks that are used to drive the DRF remediation process. **How to do manual DRF Remediation?**

Admin can do the Remediation for a specific DRF via ellipsis icon and **Remediate** action any time when it's needed:

The screenshot displays the 'User Identities' configuration page in Aras Innovator. The top navigation bar shows 'Derived Relationship Family' and 'User Identities'. The main content area is titled 'User Identities' and includes a sidebar with 'Edit', 'Refresh', 'Share', and 'More' icons. The 'Derived Relationship Family' section shows details for 'User Identities', including its name, state (Preliminary), query definition, departure item type (User), and a description. A dropdown menu is open, showing actions: 'Create New Derived Relationship Family', 'Properties', 'Permissions', 'Delete', 'Activate', 'Deactivate', and 'Remediate'. The 'Remediate' option is highlighted. Below the dropdown, there are input fields for 'Start' and 'End' dates. To the right, an 'Errors' section is visible. The bottom section shows 'Derived Relationships' and 'Event Handlers' tabs, with a table listing relationships.

Name	Departure It...	Destination It...	Description
------	-----------------	-------------------	-------------

How to set up automatic DRF Remediation This functionality enables an Admin to configure the system to automatically recalculate and resolve issues with the Derived Relationship Families on a scheduled basis. Administrators should create a daemon process that will periodically check correctness of the system DRFs and perform required repairs when needed: The Agent Service should be set up as a prerequisite to this process (details are provided in the **Prerequisites** section).



Step 1 – Activate target DRF, that you need to Remediate on a regular (automatic) basis.

Step 2 – Install scheduler: run **InstallService.bat** from Scheduler folder on CD Image. For more details you can also refer to [Aras Innovator 40 - Scheduler Service](#) .pdf of the latest released build.

Step 3 – Create the method with the following code where target DRF ID and departure_id is replaced with the correct values.

```
var schedServ = this.newItem("dr_RelationshipFamily", "dr_RemediatedDRs");  
schedServ.setID("target DRF id"); schedServ.setProperty("departure_ids", "*");  
schedServ.setProperty("is_auto_remediation", "1"); return schedServ.apply();
```

Step 4 – Admin can change **Innovator/Server/appsettings.json** config file to see logs with the following information "MinimumLevel": "Fatal", -> "MinimumLevel": "Information".

By default, the automatic remediation process logs the fixes only, that are made in UNIDRs data. If User wants to log all UNIDRs values (even those ones that stayed the same) then **dr_RemediationLoggingLevel** Innovator Variable should be set to 'verbose'. **Step 5** – Information on the results of DRF remediation can be found in two places: Innovator log files.

Innovator\Server\logs\InnovatorServer_YYYYMMDD.log
(Aras.DerivedRelationships.Loggers.RemediationProcessLogger), e.g.

```
2021-10-07T17:10:57.757 [INFO] Aras.Server.Core.CacheManager | Cache synchronization started for Application | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.757 [INFO] Aras.Server.Core.CacheManager | Cache synchronization started for Metadata | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.759 [INFO] Aras.Server.Core.CacheManager | Cache synchronization started for Application | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.759 [INFO] Aras.Server.Core.CacheManager | Cache synchronization started for Metadata | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.875 [INFO] Aras.Server.Core.CacheManager | Cache synchronization started for Application | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.875 [INFO] Aras.Server.Core.CacheManager | Cache synchronization started for Metadata | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.875 [INFO] Aras.Server.Core.CacheManager | Cache synchronization started for Application | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.875 [INFO] Aras.Server.Core.CacheManager | Cache synchronization started for Metadata | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.875 [INFO] Aras.Server.Core.CacheManager | Cache synchronization started for Application | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.875 [INFO] Aras.Server.Core.CacheManager | Cache synchronization started for Metadata | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.876 [INFO] Aras.Server.Core.CacheManager | Cache synchronization started for Application | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.876 [INFO] Aras.Server.Core.CacheManager | Cache synchronization started for Metadata | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.877 [INFO] Aras.Server.Core.CacheManager | Cache synchronization started for Application | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.877 [INFO] Aras.Server.Core.CacheManager | Cache synchronization started for Metadata | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.885 [INFO] Aras.Server.Core.CacheManager | Cache synchronization started for Application | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.885 [INFO] Aras.Server.Core.CacheManager | Cache synchronization started for Metadata | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.939 [INFO] Aras.DerivedRelationships.Loggers.RemediationProcessLogger | Universal Derived Relationships were removed by automatic remediation process | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.969 [INFO] Microsoft.AspNetCore.Mvc.Infrastructure.ObjectResultExecutor | Executing ObjectResult, writing value of type 'System.Xml.XmlDocument'. | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.970 [INFO] Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker | Executed action Aras.Server.Controllers.InnovatorServerController.ProcessInnovatorServerRequest (Aras.Server) in 1089.460ms | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.970 [INFO] Microsoft.AspNetCore.Routing.EndpointMiddleware | Executed endpoint 'Aras.Server.Controllers.InnovatorServerController.ProcessInnovatorServerRequest (Aras.Server)' | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.970 [INFO] Serilog.AspNetCore.RequestLoggingMiddleware | HTTP POST /s14-0-sp2/Server/InnovatorServer.aspx responded 200 in 1089.0704 ms | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:10:57.970 [INFO] Microsoft.AspNetCore.Hosting.Diagnostics | Request finished in 1091.0389ms 200 application/xml | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:11:02.355 [INFO] Microsoft.AspNetCore.Hosting.Diagnostics | Request starting HTTP/1.1 POST http://localhost/s14-0-sp2/Server/InnovatorServer.aspx text/xml; charset=UTF-8 448 | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:11:02.355 [INFO] Microsoft.AspNetCore.Cors.Infrastructure.CorsService | CORS policy execution failed. | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:11:02.355 [INFO] Microsoft.AspNetCore.Routing.EndpointMiddleware | Request origin http://localhost does not have permission to access the resource. | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:11:02.356 [INFO] Microsoft.AspNetCore.Routing.EndpointMiddleware | Executing endpoint 'Aras.Server.Controllers.InnovatorServerController.ProcessInnovatorServerRequest (Aras.Server)' | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:11:02.356 [INFO] Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker | Route matched with {action = "ProcessInnovatorServerRequest", controller = "InnovatorServer"}. Executing controller action with s  
2021-10-07T17:11:02.361 [INFO] Aras.Server.Core.CacheManager | Cache synchronization started for Application | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe  
2021-10-07T17:11:02.364 [INFO] Microsoft.AspNetCore.Mvc.Infrastructure.ObjectResultExecutor | Executing ObjectResult, writing value of type 'System.Xml.XmlDocument'. | SessionId: f2407149-5418-14c8-1e03-ffecbb2a89fe
```

Conversion Tasks used for remediation processing Open Conversion task that was created for this Remediation process and see the statistics for it in User data property, e.g.



 D1D03E7A415B46B98D0A0B17D3AC2F9E ☆ 



Conversion Task

Status
Succeeded

Rule
[dr_DerivedRelationshipGeneratorRule](#)

File Type	File

Started On 5/9/2022 5:12:36 PM	Finished On 5/9/2022 5:12:37 PM
--	---

User Data

```
<AML>
  <Item type="dr_RelationshipFamily" id="D630DE7BBB804D1BA296344A127E2D56">

    <departure_ids>50B53A6CC84D3E4ABDCAFEA0157C49AC,50E9C7F4C615BAA001AE7A3A2B1ECDE3,E173C0E2A5FEE6DBBBF1077B61D46AEC,864E72AECDDDD5FAD3FF032582C7E3CD,E4AD23A7FA6CD0DE4ADB78F7B6332A07,B65BFFDF62772EEF4EBA65FD5A0DFA67,D2B14E33EE5D481E9BD6F7D4FA48095A,D5DFF57BE221F0F72CEAF9B0A263FB3C,E6ED0DD88F324AF48B85C6A215546779,E84E8EB9DDAD7B5E8ED42507AB723C3C</departure_ids>

    <remediation_created_on>2022-05-09T14:12:28</remediation_created_on>
    <added_count>3</added_count>
    <removed_count>2</removed_count>

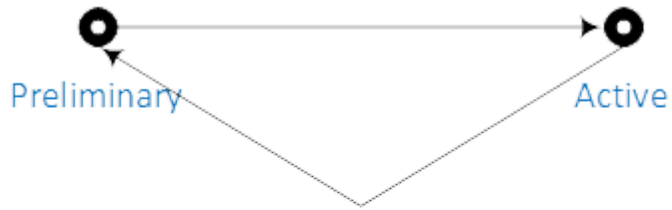
  </Item>
</AML>
```

- **<is_auto_remediation>** - flag equals to '1' that indicates the automatic run of remediation process, otherwise it does not exist
- **<added_count>** - quantity of records that was added during Conversion Task(s) execution
- **<removed_count>** - quantity of records that was deleted during Conversion Task(s) execution

DAC Definitions

When you create a DAC definition, you must specify the DAC Definition name and the Derived Relationship Family. Once you save this information, you cannot modify the definition to include a different DR Family. The DR Family defines the root ItemType of the DAC Definition. You need to create a Subdomain policy for each of the DR Family elements and the associated subdomains. The Subdomain name is the same as the corresponding Derived Family element. The ItemType for the subdomain items is the Derived Family element destination ItemType. You may then define a DAC Subdomain policy for each of the subdomains included in the DAC Domain definition. Once you create a DAC definition, you need to activate it. The following diagram shows the DAC Definition Lifecycle.



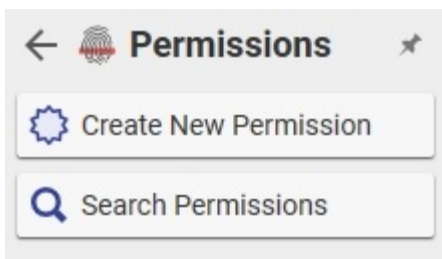


When a DAC Definition is in the Preliminary state, it means that it is being created/edited. The appropriate DAC domains have not been created yet. In this state the DAC domain has no effect on access control. Once you create and validate a DAC Definition, the Administrator activates it. The corresponding DAC Domains are created dynamically using new root items. The Access Control logic takes this information into account when determining an access request. The procedure described in the following section is an example of creating a DAC Definition.

Creating Subdomain Permissions

Use the following procedure to create the appropriate access permissions to be used for DAC Subdomain Policies.

1. In the Navigation Pane, click **Administration>Permissions**. The following menu appears:



2. Click **Create New Permission**. The following screen appears.

Permission 1

Permission 1

Save

Done

Delete

^
Permission

Name

^
Access

Identities
▼
☆

Hidden
▼

▼
▼

	Name	Get	Upd...	Dele...	Can Discover	Show Permis...	Can change a...
--	------	-----	--------	---------	--------------	----------------	-----------------

2. Enter the appropriate permission name in the **Name** field.
3. Click the **Add Identities** icon on the **Access** tab to specify permissions for existing identities.
Click the **New Identities** icon to create an identity to associate with the Permission.



Part Related to ...

Part Related to Project

Edit

Permission

Name
Part Related to Project

Access

Identities

Hidden

	Name	Get	Upd...	Dele...	Can Discover	Show Permis...	Can change a...
	Team Manager	☐	☐	☐	☐	☐	☐
	Team Member	☐	☐	☐	☐	☐	☐

Important

If a Team is associated with the Domain Root ItemType (Project in our example), roles in the Subdomain permission need to match those in the team in order to take effect. Refer to the Creating Teams section in the Appendix for more information.

- Click **Done** to save and unclaim the Permission item.

Creating a DAC Definition

- In the Navigation pane, click **Administration>Access Control>DAC Definitions** from the TOC. The following menu appears:



← **DAC Definitions** ★

⚙ Create New DAC Definition

🔍 Search DAC Definitions

2. Click **Create New DAC Definition**. The DAC Definition window appears:

DAC Definition 1 ✕

File Edit Views Actions Reports Tools Help

⚙ ⬇ ✖ ↺ 🖨 📊 📄 🔒 🔓 ↶ ↷ ⚙ ? ✓

 **Name**

State

Derived Relationship Family

...

Root ItemType

Description

3. Enter a name for the DAC Definition in the **Name** field.
4. Click the ellipses in the **Derived Relationship Family** field to select the appropriate DR Family.

DAC Definition 2 x

File Edit Views Actions Reports Tools Help

⚙️ ⬇️ ❌ ↺️ 🖨️ 📊 📄 🔒 🔓 ↶️ ↷️ ⚙️ ? ✓

Name
Part Related to a Project

State

Derived Relationship Family
Part Related to Project ...

Root ItemType

Description

- Click on the arrow icon. The DAC Definition screen appears with links to the Derived Relationship Family and the Root ItemType defined in the DR Family.

Part Related to ... x

File Edit Views Actions Reports Tools Help

⚙️ ⬇️ ✕ ↺️ 🖨️ 📊 📄 🔒 🔓 ↶️ ↷️ ⚙️ ? ✓

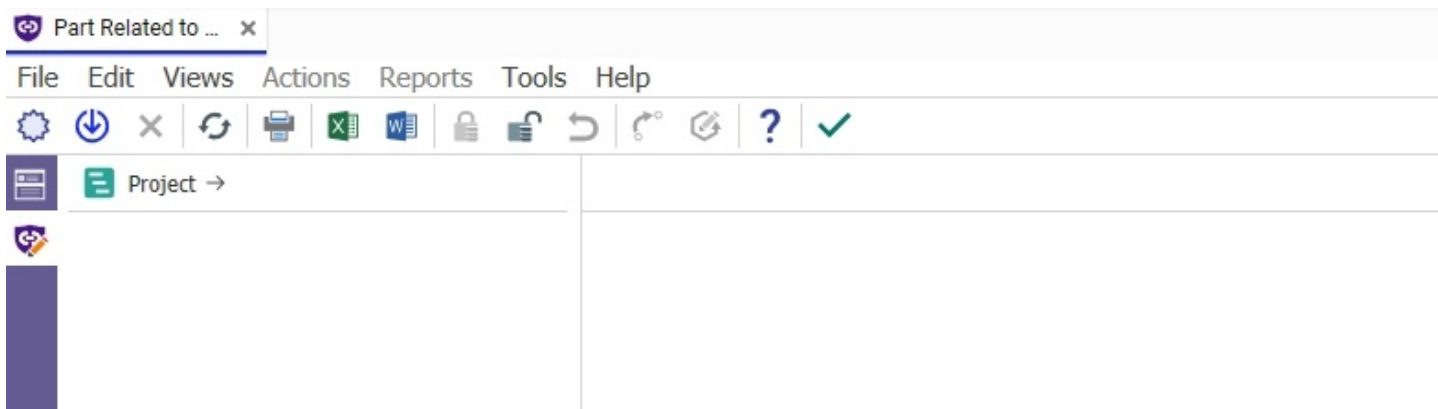
Name
Part Related to Project

Derived Relationship Family
[Part Related to Project](#)

Root ItemType
[Project](#)

Description

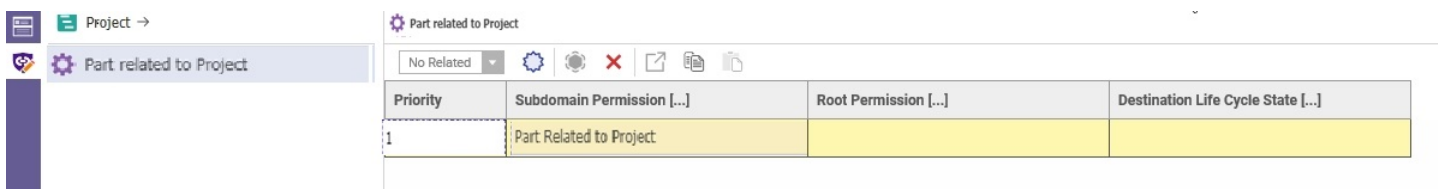
6. Click the Editor icon that appears in the left margin. A screen like the following appears. It highlights the Derived relationship associated with the DAC Definition.



7. Click the **New Relationship** icon to specify permissions for the DAC Definition.



8. Enter a number in the **Priority** field.
9. Click on the ellipses in the **Subdomain Permission** field and select a Permission. In Section 2.5, the example created was **Part related to Project**.



10. Click on the ellipses in the **Root Permission** field and select a Permission for the root. In this example, the Destination Life Cycle State is left empty to grant subdomain permissions to parts that have any Life Cycle state. If the Subdomain Policy should apply to a part in a specific Life Cycle State, click on the ellipses in the Destination Life Cycle State cell and select a Life Cycle State.

Important



A Subdomain Permission with an empty Destination Lifecycle State must have lower priority than a Subdomain Permission with a specified Destination Lifecycle with the same Root Permission.

11. Repeat this process for all Subdomains that should be mapped.
12. When complete, click **save** and then **unclaim** the DAC Definition.

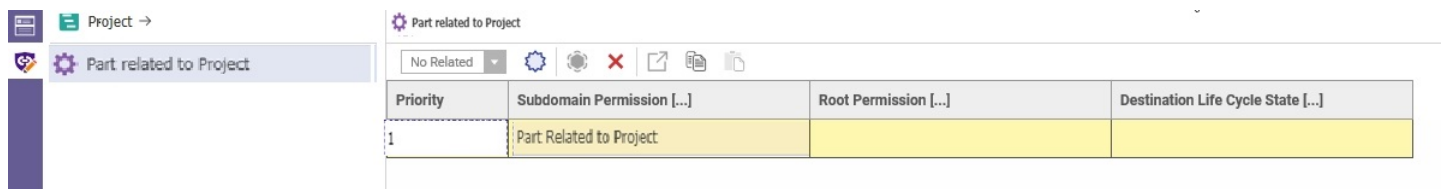
Activating/Deactivating a DAC Definition

Once you have fully defined a DAC Definition and saved it, perform the following actions before activating it:

- Make sure that the underlying DR Family is active. If it isn't you will not be able to activate the DAC Definition.
- Validate the associated Subdomain policies.

The following procedure is an example of activating a DAC Definition:

1. Open the DAC definition and click the Editor button. A screen like the following appears:



Priority	Subdomain Permission [...]	Root Permission [...]	Destination Life Cycle State [...]
1	Part Related to Project		

2. Unclaim the DAC Definition and click the **Promote Item** icon. The Promote dialog box appears.



Promote Part Related to a Project



To State

Active

Ready

3. Select **Active** and click the green checkmark. The DAC Definition is now in Active state and all permissions are propagated.

