Polarion Configuration for Perseus

Obeo.fr

Version 2023.4.0, June 6, 2023

Table of Contents

Perseus Polarion Configuration Guide for the 2023.3.0 Release.

> **Perseus** is the name of the technical component behind the OCP Publication and Publication for Capella offers. In this documentation, we will refer only to Perseus.

Overview of Polarion - Perseus Connectivity

Perseus exposes model data according to the OSLC AM (Architecture Management) domain.

Polarion can be configured to make it possible to link Polarion items to model objects exposed by Perseus.

This document describes the necessary configuration that must be done in Polarion.

Polarion Configuration

Global Configuration

- Declare the Perseus server as a friend:

  - In Polarion, connect as an administrator;

  - Go to *Global Administration > Linked Data > Linked Data Friends*;

    - This configuration is made at `http://your.polarion.server/polarion/#/administration/linkeddata/friends`;

  - Click on *Add Friend* and enter the following information:

    - URI: `"http://<your perseus server>/oslc/rootservices"`, e.g. `http://perseus.mycompany.com:9090/oslc/rootservices`;

    - Name: `"Perseus Server"` (or any name to recognise the Perseus server instance);

    - OAuth Secret: a password - remember it, there will be no way to recover it;

  - Click on Next. A dialog appears to validate the provisional key that will be used to authenticate communications between the two servers:

- Use the Perseus admin credentials to connect in this dialog (by default, "admin"/"odweb"):

- Enter a name for the Polarion server, as seen from Perseus;

- Leave `"Trust Level"` to `"Trusted"`;

- Click on *Approve* - You can still edit the server name, and access that edition page later from Perseus.

- Close the dialog with the Close button;
  - If you don't have these credentials, ask the Perseus administrator to connect to Perseus server and navigate to http://perseus.mycompany.com:9090/admin/oauth/consumers to validate the provisional key;
  - The Perseus friend appears in the list of OSLC friends;

- Configure the Linked Data Semantics:
  - In the `<namespaces>` element, add a namespace for OSLC AM with `prefix="oslc_am"` and `url="http://open-services.net/ns/am#"`;

- In the `<domains>` element, add domain with `prefix="oslc_am"` and resourceType `label="Architecture Resource"`, `uri="oslc_am:Resource"`;
- In the `<linking>` element, add the relevant links according to the kinds of links you want to make available. An example is provided below;
- Save the configuration.

```
<namespaces>
  (...)
  <namespace prefix="oslc_am" url="http://open-services.net/ns/am#"/>
</namespaces>
<domains>
  (...)
  <!-- OCP -->
  <domain prefix="oslc_am">
    <resourceType label="Architecture Resource" uri="oslc_am:Resource"/>
  </domain>
</domains>
<linking>
  (...)
  <!-- OCP -->
  <link name="oslc_rm:elaboratedBy" reverse="oslc_rm:elaborates">
    <from type="oslc_rm:Requirement"/>
    <to type="oslc_am:Resource"/>
  </link>
  <link name="oslc_rm:satisfiedBy" reverse="oslc_rm:satisfies">
    <from type="oslc_rm:Requirement"/>
    <to type="oslc_am:Resource"/>
  </link>
  <link name="oslc_rm:specifiedBy" reverse="oslc_rm:specifies">
    <from type="oslc_rm:Requirement"/>
    <to type="oslc_am:Resource"/>
  </link>
</linking>
```

Project Configuration

A project-specific configuration must be made for each Polarion project that needs to support linking with Perseus model elements.

- Go to the configuration management page of the Polarion Project;
  - This configuration is made at `http://your.polarion.server/polarion/#/project/<your_polarion_project>/administration/linkeddata/serviceproviders`;
- Declare the Perseus server as a friend:
  - Click on *Add Association*;
    - In the dialog that opens, select the relevant Perseus friend server (that has been
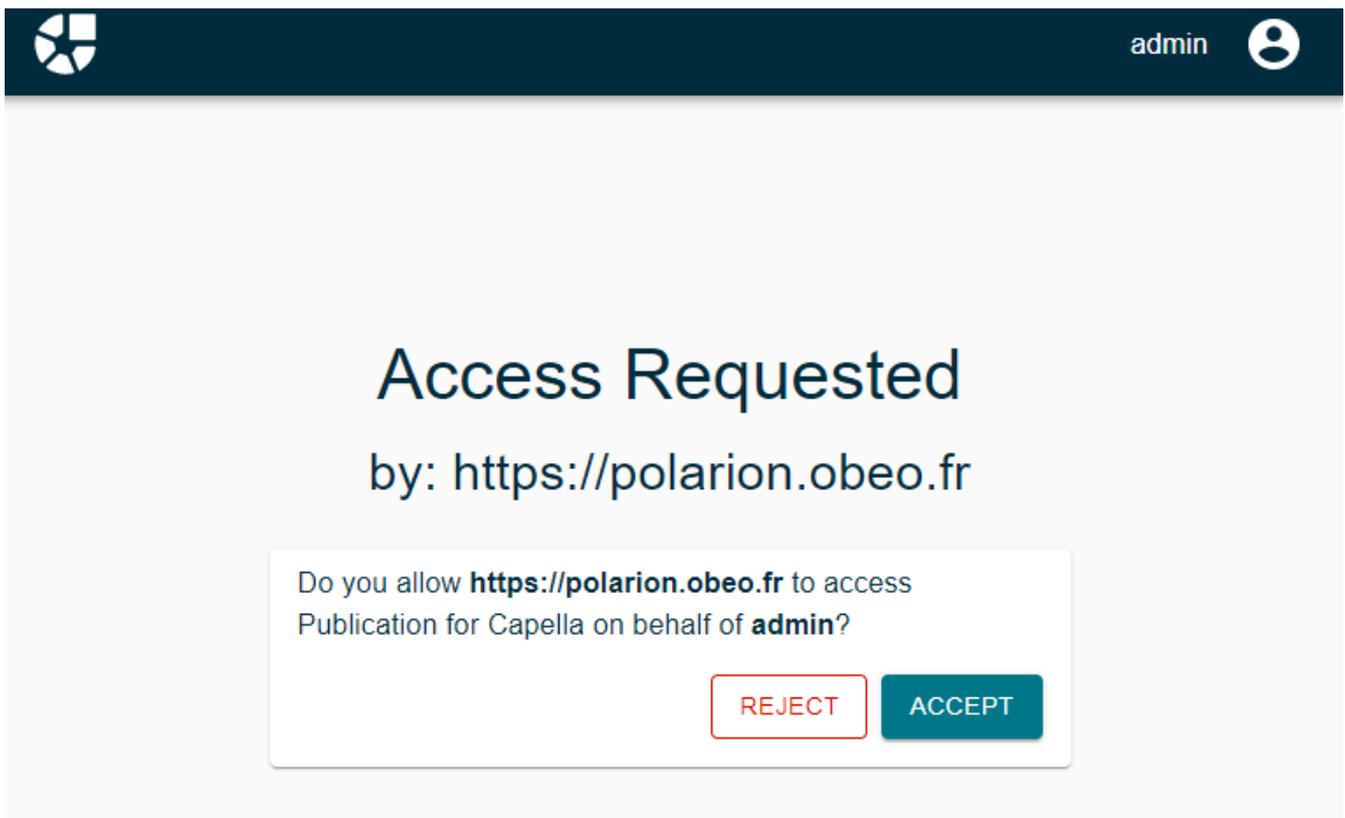
configured in section Global Configuration):
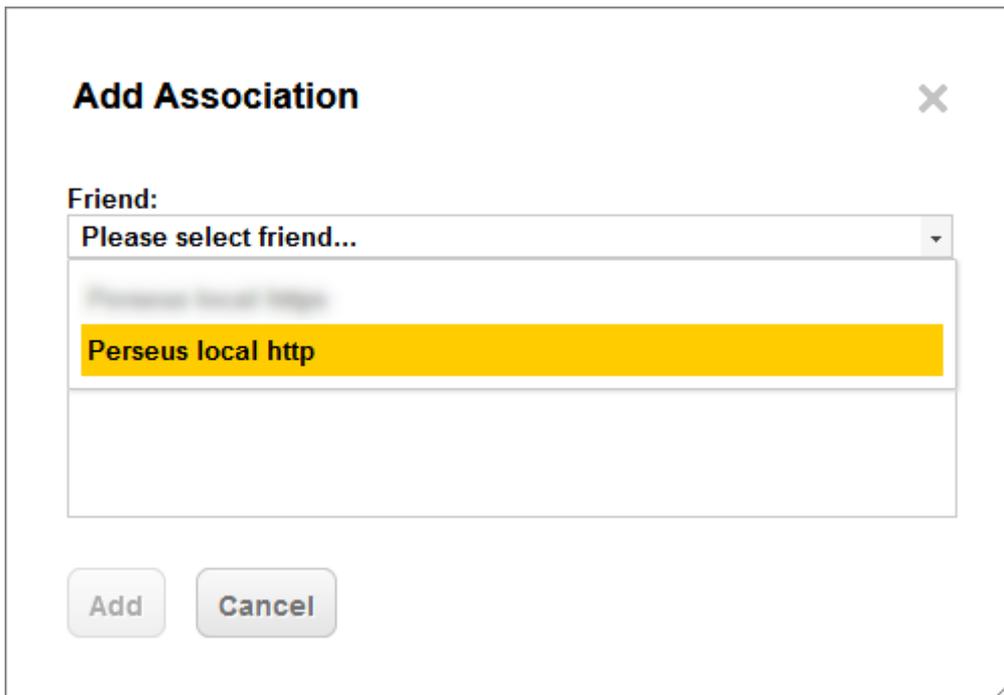
- A pop-up window opens (check your browser pop-up settings if it doesn't):

> Sometimes, Polarion's authentication is rejected because the authentication token is too old. In such a case, you will get an error message indicating that the connection failed. Close the pop-up window and reopen it and Polarion should authenticate properly.

- Enter valid Perseus credentials to connect to perseus;

- Select the relevant service provider, and click on *Add*:

- The association appears in the list of the Polarion project's linked data associations;

Linked Data Associations

ⓘ Associate Project resources with Linked Data Friend resources.

Associations

Friend	URI	Artifact Container	Actions
~~Perseus local http~~	~~http://perseus.lde.fr:8080/services/serviceProviders/0~~	~~Service Provider 'Perseus Server Model Provider'~~	🗑 Delete
Perseus local http	http://perseus.lde.fr:8080/services/serviceProviders/0	Service Provider 'Perseus Server Model Provider'	🗑 Delete
			➕ Add Association

- Go to the Linked Data Mapping page of the Polarion Project;
  - This configuration is made at `http://your.polarion.server/polarion/#/project/<your_polarion_project>/administration/linkeddata/mapping`;
  - In the `<link-role-mappings>` section, add mappings for the roles you want to use to link polarion objects to Perseus model elements;
  - Save the configuration;
    - Example:

```
<link-role-mappings>
  (...)
  <link-role-mapping linkRole="elaborated_by"
oslcLinkProperty="oslc_rm:elaboratedBy"/>
  <link-role-mapping linkRole="satisfied_by" oslcLinkProperty="oslc_rm:satisfiedBy"/>
  <link-role-mapping linkRole="specified_by" oslcLinkProperty="oslc_rm:specifiedBy"/>
</link-role-mappings>
```

- Go to the configuration management page of the Work item enumerations;
  - This configuration is made at `http://your.polarion.server/polarion/#/project/drivepilot/administration/workitems/enumerations`;
- Edit the `workitem-link-role-enum.xml` by clicking on the *Edit* link in the column *Actions*;
  - Make sure there is a line for each role defined in the previous step, and that these roles are associated to the right type of Polarion element;
    - Example:

elaborated_by	is elaborated by	elaborates	☐	☐	1	Used to link to Model elements hosted on Perseus
satisfied_by	is satisfied by	satisfies	☐	☐		Used to link to Model elements hosted on Perseus
specified_by	is specified bu	specifies	☐	☐		Used to link to Model elements hosted on Perseus

- Click on *Save* to save the configuration.

Configuration is over. It should now be possible to link Polarion work items to Perseus model elements.