CAMBRIDGE SEMANTICS

Anzo® 5.3 Administration Guide

Last Updated: 4/6/2023

Online documentation is available at docs.cambridgesemantics.com

Table of Contents

Accessing the Administration Application

Go to the following URL to open the Administration application:

```
https://<hostname>/sdl/index.html#/admin
```

Where <hostname> is the Anzo server DNS name or IP address. You can change the URL for the Administration application by configuring the **Admin Home Page** value in server settings. For more information, see Configure the Default Root Pages.

To access the Administration application from the Anzo application, click the administration icon (⚙) on the right side of the top menu bar. Clicking the icon opens the Administration menu, and selecting a menu item opens the application.

Anzo Server Administration

The topics in this section provide information about managing the Anzo server configuration.

Starting and Stopping Anzo

If Anzo is run via a systemd service, as described in Configure and Start the Anzo Service in the Deployment Guide, use systemctl to start and stop Anzo. To start Anzo, run the following command:

```
sudo systemctl start <service_name>
```

For example: `sudo systemctl start anzo-server`

To stop Anzo, run the following command:

```
sudo systemctl stop <service_name>
```

For example: `sudo systemctl stop anzo-server`

To start Anzo using the AnzoServer utility, run the following command. Make sure that you are logged in as the Anzo service user before stopping or starting Anzo:

```
<install_path>/Server/AnzoServer start
```

To stop Anzo, run the following command:

```
/<install_path>/Server/AnzoServer stop
```

You can also start and stop Anzo from the symbolic links if they were created for your installation. For example, `/etc/init.d/AnzoServer start` or `/etc/init.d/AnzoServer stop`.

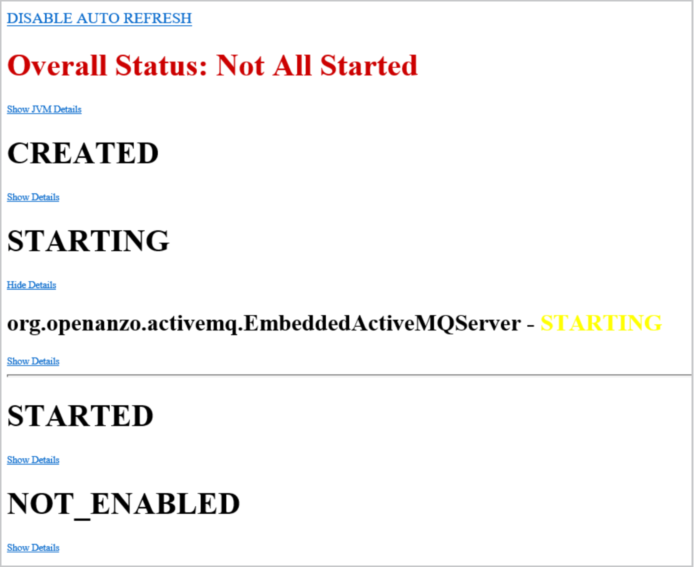
Monitoring Startup Status

It can take a few minutes for Anzo to complete the startup process. You can monitor the status by viewing the Anzo Status page. To see the Status page, go to the following URL in your browser:

```
http://<server_name_or_IP_address>:8945/status
```

Where <server_name_or_IP_address> is the name or IP address of the server that hosts Anzo.

For example, the following image shows the Status page message displayed while Anzo is starting:



DISABLE AUTO REFRESH

Overall Status: Not All Started

Show JVM Details

CREATED

Show Details

STARTING

Hide Details

org.openanzo.activemq.EmbeddedActiveMQServer - STARTING

Show Details

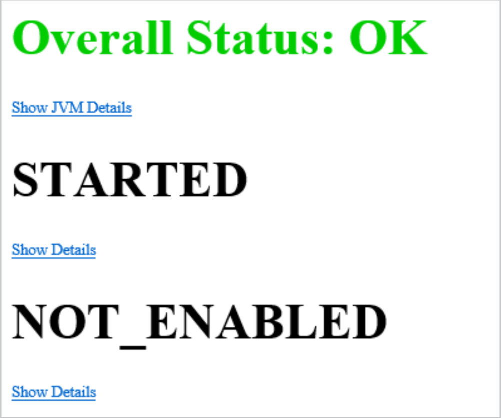
STARTED

Show Details

NOT_ENABLED

Show Details

The image below shows the Status page message when Anzo startup is complete:



Overall Status: OK

Show JVM Details

STARTED

Show Details

NOT_ENABLED

Show Details

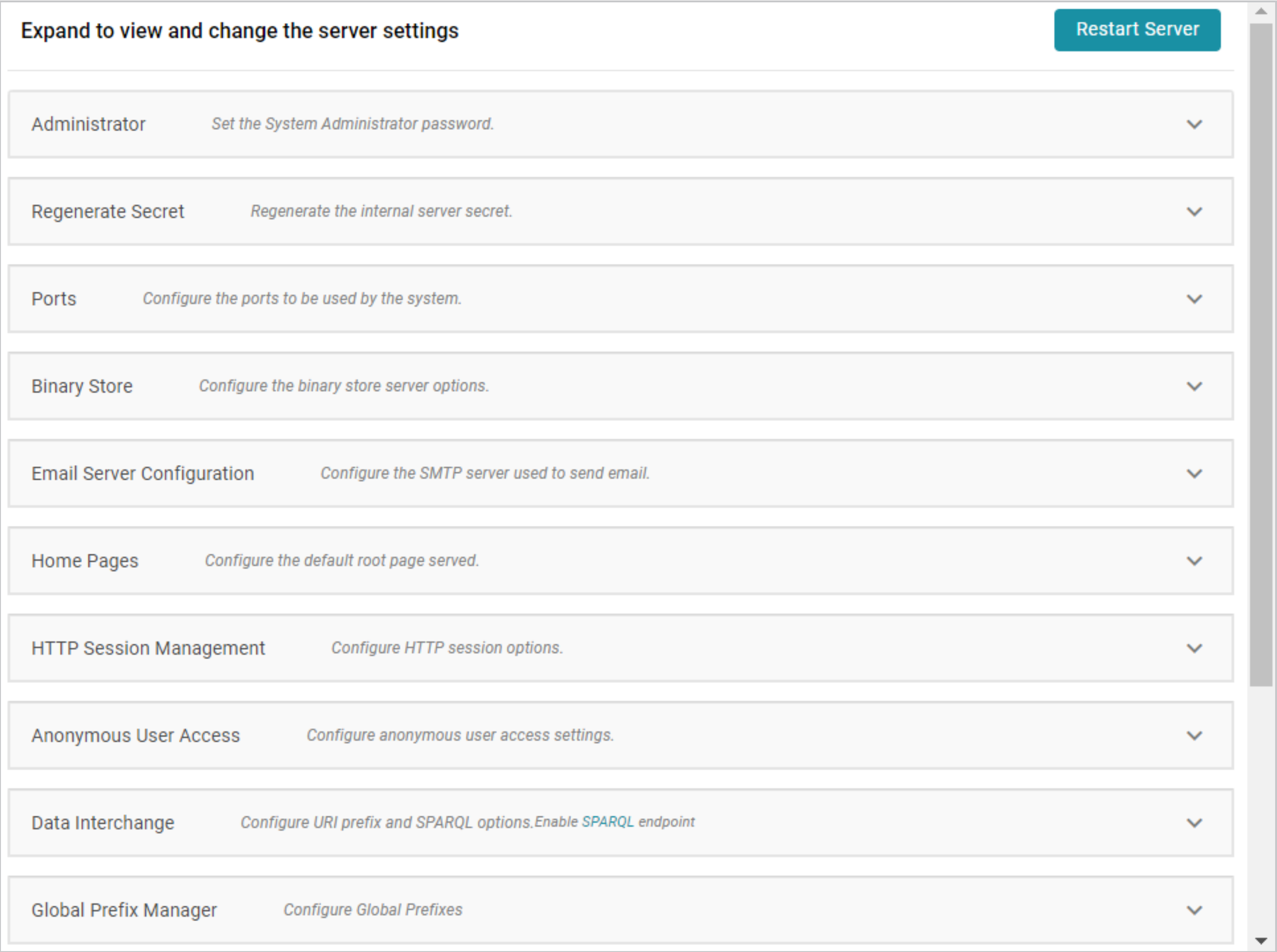
Changing Anzo Server Settings

This topic provides instructions for changing Anzo server settings as well as reference information for each of the options.

- Changing Settings
- Settings Reference

Changing Settings

1. In the Administration application, expand the **Servers** menu and click **Server Settings**. The Server Settings screen is displayed. The options that you can configure are described on the screen:



Expand to view and change the server settings

Administrator Set the System Administrator password.

Regenerate Secret Regenerate the internal server secret.

Ports Configure the ports to be used by the system.

Binary Store Configure the binary store server options.

Email Server Configuration Configure the SMTP server used to send email.

Home Pages Configure the default root page served.

HTTP Session Management Configure HTTP session options.

Anonymous User Access Configure anonymous user access settings.

Data Interchange Configure URI prefix and SPARQL options.Enable SPARQL endpoint

Global Prefix Manager Configure Global Prefixes

2. To change the configuration, expand an option to display the related settings. Then click the **Edit** button and specify the desired value for each setting. For specifics about each option, see Settings

Reference below.

> **Note**
> You can have one option open for editing at a time. If you are in the process of modifying an option and have not saved the changes, all other **Edit** buttons are disabled until you save or cancel the changes.

3. Click **Save** to save the changes, and then restart Anzo to complete the configuration change.

> **Important**
> After changing any of the server configuration settings, you must restart Anzo to apply the change.

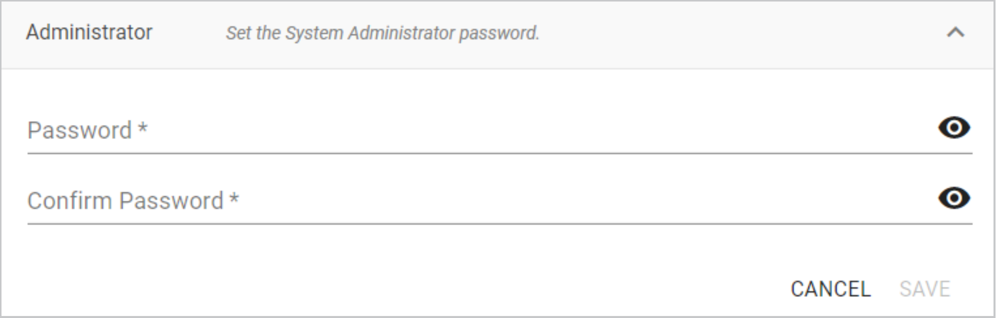
Settings Reference

This section provides reference information for each configuration option.

- Set the System Administrator Password
- Regenerate the Internal Server Secret
- Configure the Ports to be Used by the System
- Configure the Binary Store Server Options
- Configure the SMTP Server Used to Send Email
- Configure the Default Root Pages
- Configure HTTP Session Options
- Configure Anonymous User Access
- Configure URI Prefix and SPARQL Options
- Configure Global Prefixes
- Configure the Versioning Environment
- Configure Network Connections to an Anzo Distributed Unstructured Cluster
- Configure the Default ETL Engine
- Configure the Default Anzo Data Store

Set the System Administrator Password

To change the system administrator (**sysadmin**) password, expand the **Administrator** option and click **Edit**.

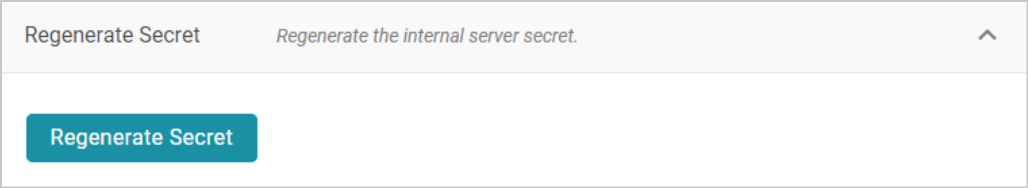


Type the new password in the **Password** and **Confirm Password** fields. Then click **Save**.

Regenerate the Internal Server Secret

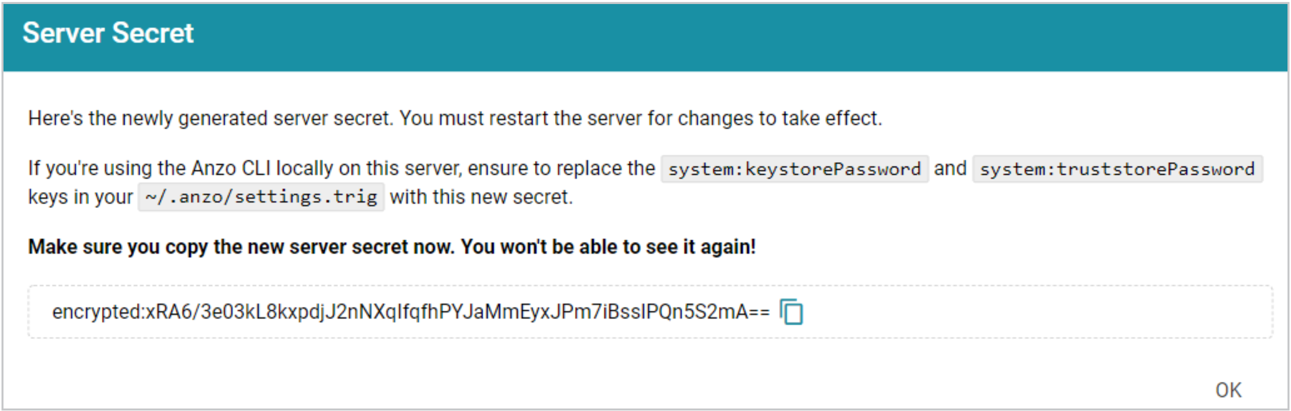
> **Note**
> Regenerating the secret requires a restart of Anzo. In addition, Cambridge Semantics recommends that you back up the current Anzo installation before regenerating the secret.

1. To change the password for the Anzo key and trust stores, expand the **Regenerate Secret** option and click **Edit**.



2. To proceed, click the **Regenerate Secret** button. Review the confirmation message that is displayed and click **Yes** to generate a new secret.

   Anzo generates the new secret and presents a dialog box that displays the encrypted secret to copy. For example:



Server Secret

Here's the newly generated server secret. You must restart the server for changes to take effect.

If you're using the Anzo CLI locally on this server, ensure to replace the `system:keystorePassword` and `system:truststorePassword` keys in your `~/.anzo/settings.trig` with this new secret.

Make sure you copy the new server secret now. You won't be able to see it again!

```
encrypted:xRA6/3e03kL8kxpdjJ2nNXqlfqhPYJaMmEyxJPm7iBssIPQn5S2mA==
```

3. Make sure you copy the secret because it is not possible to view again.

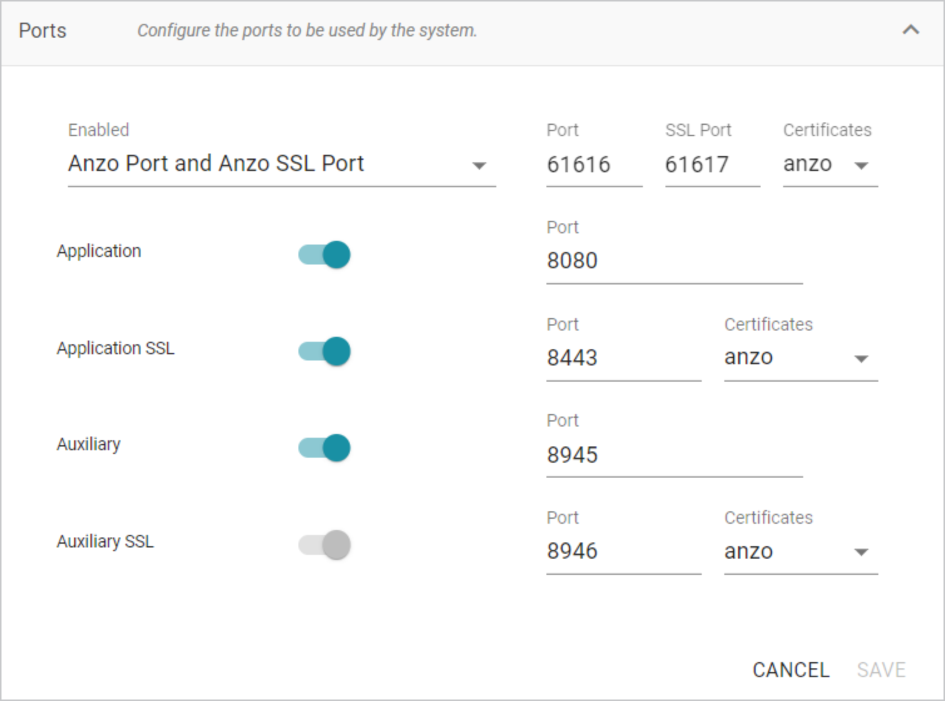
4. If you regenerated the secret on a server where the Anzo Admin CLI is used, the new secret also needs to be changed in the `~/.anzo/settings.trig` file for the Anzo service user. To replace the secret in `settings.trig`, follow these steps:
   a. Open `~/.anzo/settings.trig` for editing.
   b. Locate the `system:keystorePassword` and `system:truststorePassword` properties.
   c. Replace both the object values for both properties with the secret that was copied in step 3. Replace only the content between the quotation marks as shown below:

      ```
      system:keystorePassword "<new_secret>"^^anzo:password ;
      system:truststorePassword "<new_secret>"^^anzo:password ;
      ```

   d. Save and close settings.trig.
5. Restart Anzo to apply the new secret.

Configure the Ports to be Used by the System

To change, enable, or disable the Anzo server ports, expand the **Ports** option and click **Edit**.



Ports *Configure the ports to be used by the system.*

Enabled: Anzo Port and Anzo SSL Port
Port: 61616
SSL Port: 61617
Certificates: anzo

Application ●
Port: 8080

Application SSL ●
Port: 8443
Certificates: anzo

Auxiliary ●
Port: 8945

Auxiliary SSL ○
Port: 8946
Certificates: anzo

Change the values in the Port fields to specify alternate port numbers. To enable or disable a port, move the slider next to the application name to the left or right. The list below describes the settings:

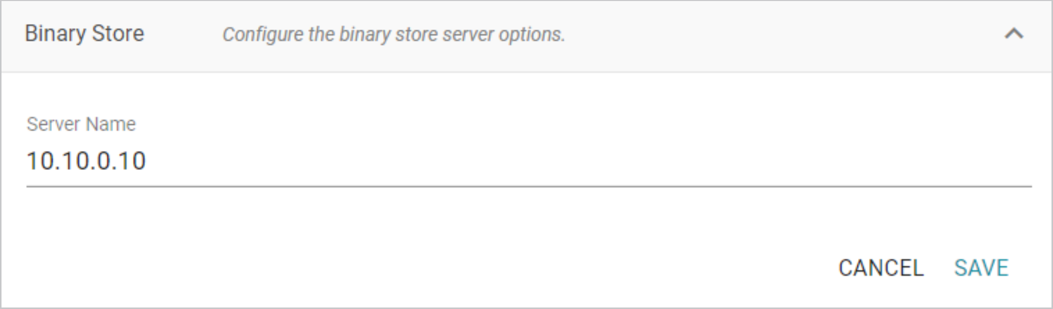
- The fields at the top of the screen specify the Anzo server ports. By default, the Anzo and Anzo SSL ports are enabled. If you want to disable one of the ports, click the **Enabled** drop down list and select the option that you want to leave enabled. To change port numbers, click in the **Port** field and specify the port.

- The **Application** and **Application SSL** ports are the HTTP and HTTPS client application ports.
- The **Auxiliary** and **Auxiliary SSL** ports are the HTTP and HTTPS Administration client ports.

For information about managing the certificates to use for the SSL ports, see Replacing the Anzo Certificate.

Configure the Binary Store Server Options

To change the host server for the binary (blob) store, expand **Binary Store** and click **Edit**.



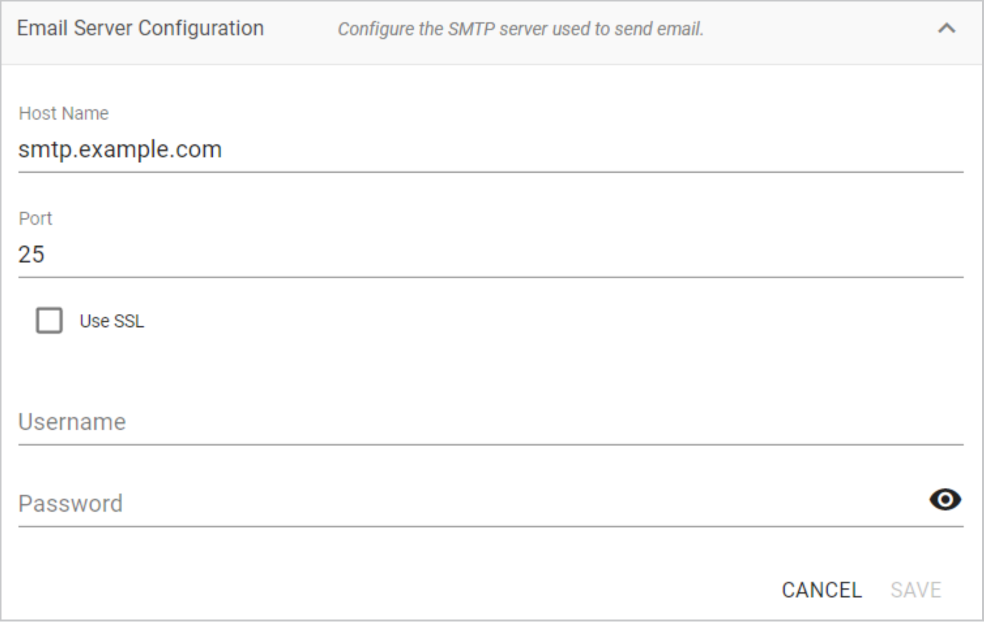
Binary Store *Configure the binary store server options.*

Server Name
10.10.0.10

The Server Name defaults to the host name or IP address for the Anzo server. To specify a different host for the binary store, type the new host name or IP address in the **Server Name** field, and then click **Save**.

Configure the SMTP Server Used to Send Email

To configure an SMTP server for sending email, expand **Email Server Configuration** and click **Edit**.



Email Server Configuration *Configure the SMTP server used to send email.*

Host Name
smtp.example.com

Port
25

☐ Use SSL

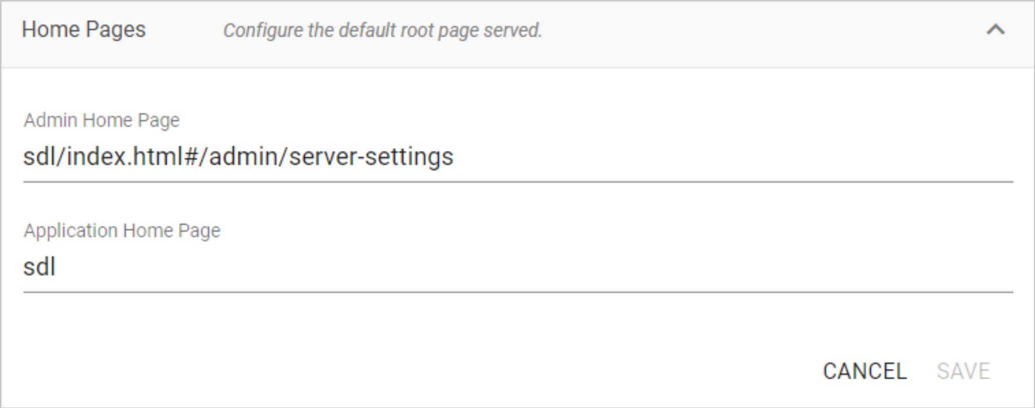
Username

Password

- **Host Name** is the host name or IP address for the SMTP server.
- **Port** is the port for the connection.
- If the email server is configured for SSL authentication, select the **Use SSL** checkbox to enable SSL authentication.
- Specify the **Username** and **Password** to use for authentication.

Configure the Default Root Pages

To change the home page path for the Anzo application and Administration application URLs, expand **Home Pages** and click **Edit**.



Home Pages *Configure the default root page served.*

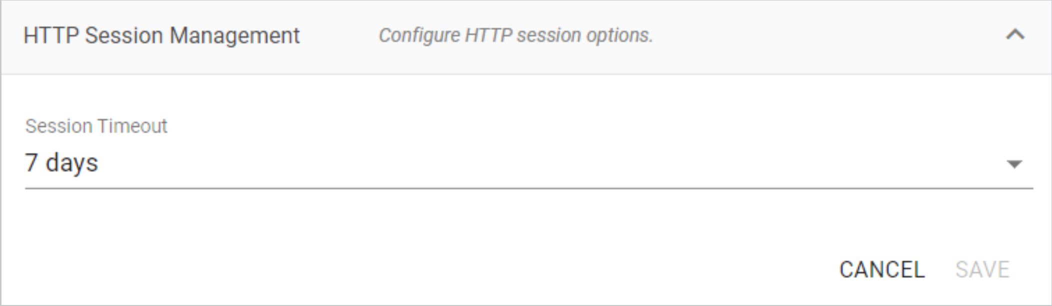
Admin Home Page
sdl/index.html#/admin/server-settings

Application Home Page
sdl

- The **Admin Home Page** is the home page path for the Administration application.
- The **Application Home Page** is the home page path for the Anzo application.

Configure HTTP Session Options

To configure the HTTP session timeout value, expand **HTTP Session Management** and click **Edit**.



HTTP Session Management *Configure HTTP session options.*

Session Timeout
7 days

Click the **Session Timeout** drop-down list and select the timeout value.

Configure Anonymous User Access

Before enabling anonymous access, consider the following security implications:

- Anonymous User Permissions
- Anonymous User Limitations
- Important Considerations

Anonymous User Permissions

When anonymous access is enabled:

- The server allows any user to connect to the Hi-Res Analytics application without a username and password. A user can connect to without having an account in Anzo.
- Anonymous users are considered members of the Everyone role. Anonymous users can read data in Anzo that is tagged as readable by Everyone.

Anonymous User Limitations

Anonymous users cannot:

- Add, delete, or modify data. Anonymous users cannot write or delete data even if the Everyone role has write or delete access.
- Change permissions on the artifacts in Anzo. Anonymous users cannot change the Sharing or Security tab settings for any data on the server even if the Everyone role has write or delete access to an artifact's metadata.

Important Considerations

This section lists important ideas to consider before enabling anonymous access.

Consider Existing Access Control

Users might have been permissions without anticipating that users could have anonymous access. Before enabling anonymous access, consider that data that is viewable by the **Everyone** role becomes visible to anonymous users. You might need to change the permissions for existing data, such as by granting read access to the **Authenticated Users** role instead of the Everyone role. For more information about permissions, see Predefined Anzo Roles and Permissions.

Consider Server Network Protections

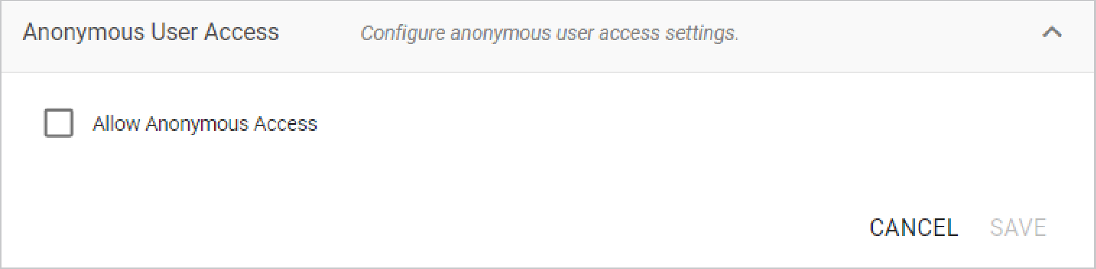
Consider that anyone who can reach the server via the network will be able to use it as an anonymous user. Evaluate firewalls and other network protection mechanisms to limit access to the Anzo server as desired. For example, you might want to allow anonymous access to anyone inside your organization's internal network but disable access to the server from the public internet.

Anonymous Access Can Be Useful

Allowing anonymous access makes it easy to share data and views of data with others. For example, it means that you can share your Hi-Res Analytics dashboards with people who do not have a user account. It also lets you embed read-only interactive Hi-Res Analytic views inside other websites.

Configuring Anonymous Access

To enable or disable anonymous user access, expand **Anonymous User Access** and click **Edit**.



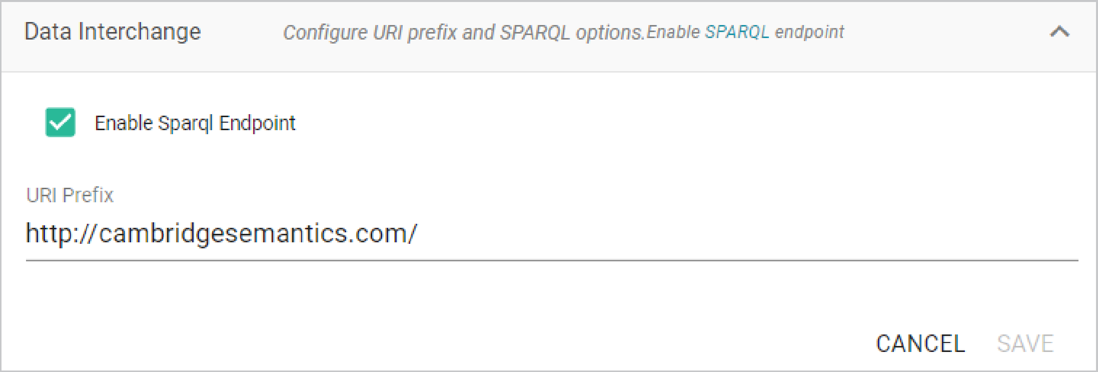
Anonymous User Access *Configure anonymous user access settings.*

□ Allow Anonymous Access

To enable anonymous access, select the **Allow Anonymous Access** checkbox. To disable anonymous access if it is enabled, clear the checkbox. Then click **Save**.

Configure URI Prefix and SPARQL Options

To enable or disable the Anzo SPARQL endpoint or customize the URI prefix that Anzo generates for data identifiers, expand **Data Interchange** and click **Edit**.



Data Interchange *Configure URI prefix and SPARQL options.Enable SPARQL endpoint*

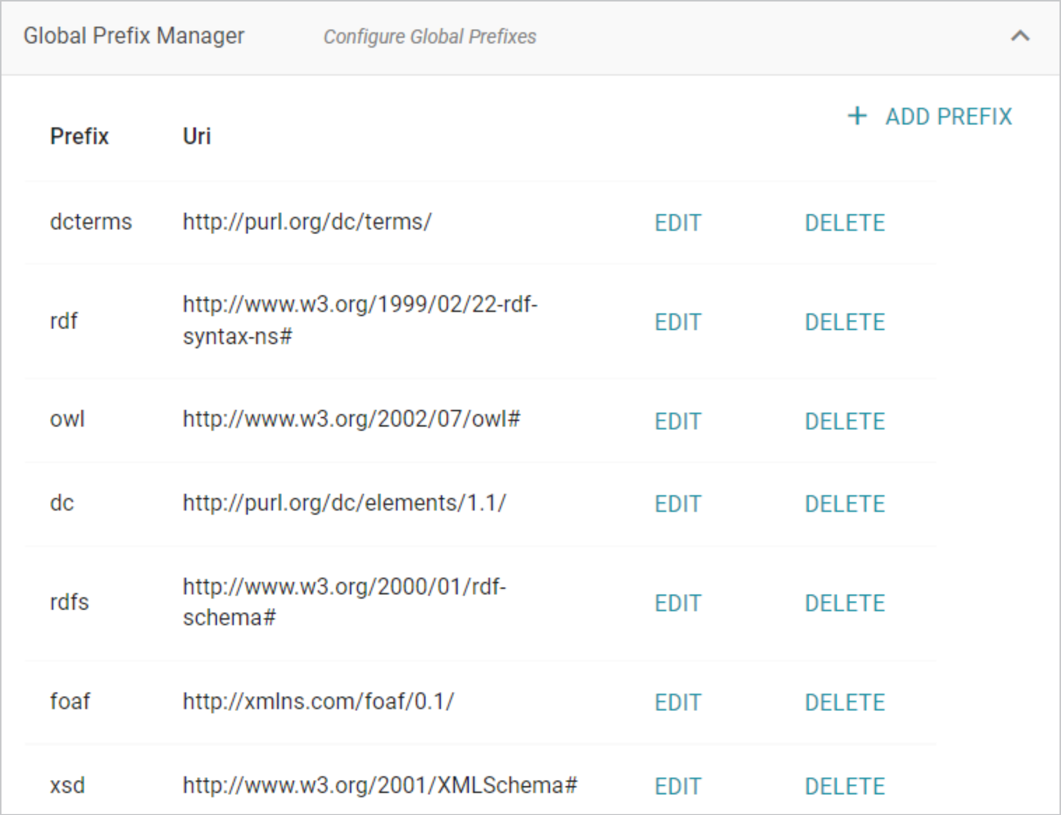
☑ Enable Sparql Endpoint

URI Prefix
http://cambridgesemantics.com/

- If you want to enable or disable the Anzo SPARQL endpoint, select or clear the **Enable SPARQL Endpoint** checkbox.
- To change the prefix that Anzo uses when generating URIs, type the new value in the **URI Prefix** field. The URI Prefix is mostly used for consistency in internal data, but it is also used by default for data model URI prefixes when the model does not define the URI template to use. When changing the URI Prefix, make sure that the value is a valid prefix. See Relative IRIs in the SPARQL Query Language specification for more information.

Configure Global Prefixes

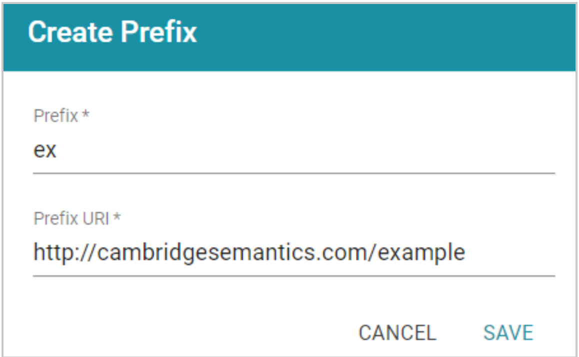
The Global Prefix Manager stores standard prefixes and any custom prefixes that you want Anzo to recognize globally. Defining global prefixes creates shortcuts for inserting the prefixes in Query Builder and data layer queries. To manage global prefixes, expand **Global Prefix Manager**:



Global Prefix Manager *Configure Global Prefixes*

Prefix	Uri		
dcterms	http://purl.org/dc/terms/	EDIT	DELETE
rdf	http://www.w3.org/1999/02/22-rdf-syntax-ns#	EDIT	DELETE
owl	http://www.w3.org/2002/07/owl#	EDIT	DELETE
dc	http://purl.org/dc/elements/1.1/	EDIT	DELETE
rdfs	http://www.w3.org/2000/01/rdf-schema#	EDIT	DELETE
foaf	http://xmlns.com/foaf/0.1/	EDIT	DELETE
xsd	http://www.w3.org/2001/XMLSchema#	EDIT	DELETE

To add a prefix, click **Add Prefix**. Anzo opens the Create Prefix dialog box. In the **Prefix** field, specify the abbreviation that you want to use to represent the URI. In the **Prefix URI** field, specify the full, valid URI. For example:

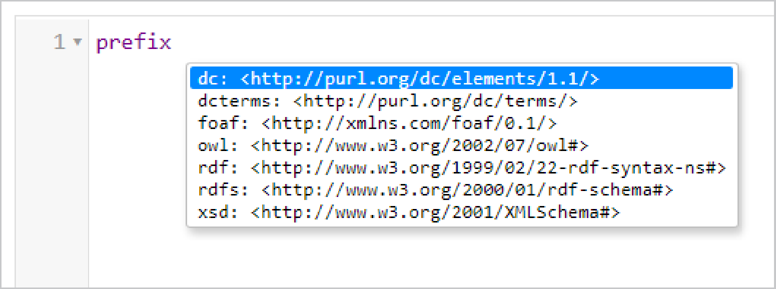


Create Prefix

Prefix *
ex

Prefix URI *
http://cambridgesemantics.com/example

Click **Save** to save the definition. To use global prefix shortcuts in the Anzo application, type "prefix" followed by a space in the Query Builder or a Query Step to open a tooltip that lists the global prefixes. For example:

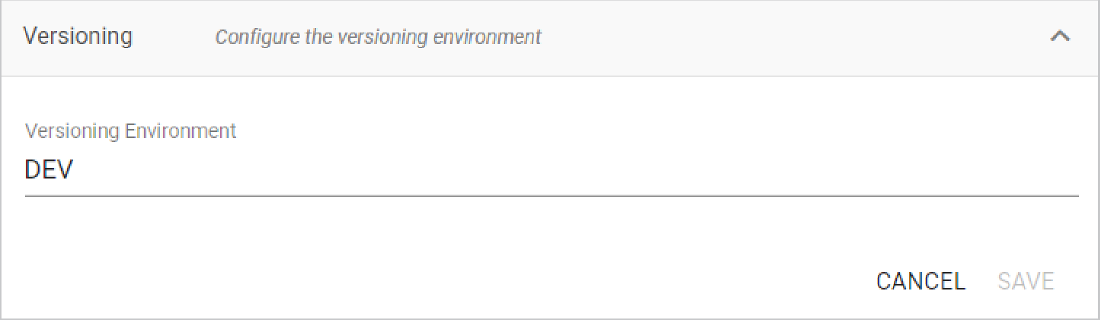


```
prefix
dc: <http://purl.org/dc/elements/1.1/>
dcterms: <http://purl.org/dc/terms/>
foaf: <http://xmlns.com/foaf/0.1/>
owl: <http://www.w3.org/2002/07/owl#>
rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
rdfs: <http://www.w3.org/2000/01/rdf-schema#>
xsd: <http://www.w3.org/2001/XMLSchema#>
```

Clicking a prefix inserts a PREFIX statement into the query. In addition, typing the abbreviation for a global prefix followed by a colon (:) automatically inserts the PREFIX statement into the query without opening the tooltip. For example, typing **ex:** inserts a statement for the prefix that was defined in the example above.

Configure the Versioning Environment

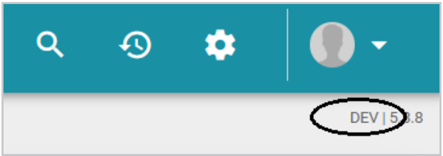
To change the variable value for the Version Environment tag that is displayed at the top of the Anzo application and that Anzo adds to archived versions of entities, expand **Versioning** and click **Edit**.



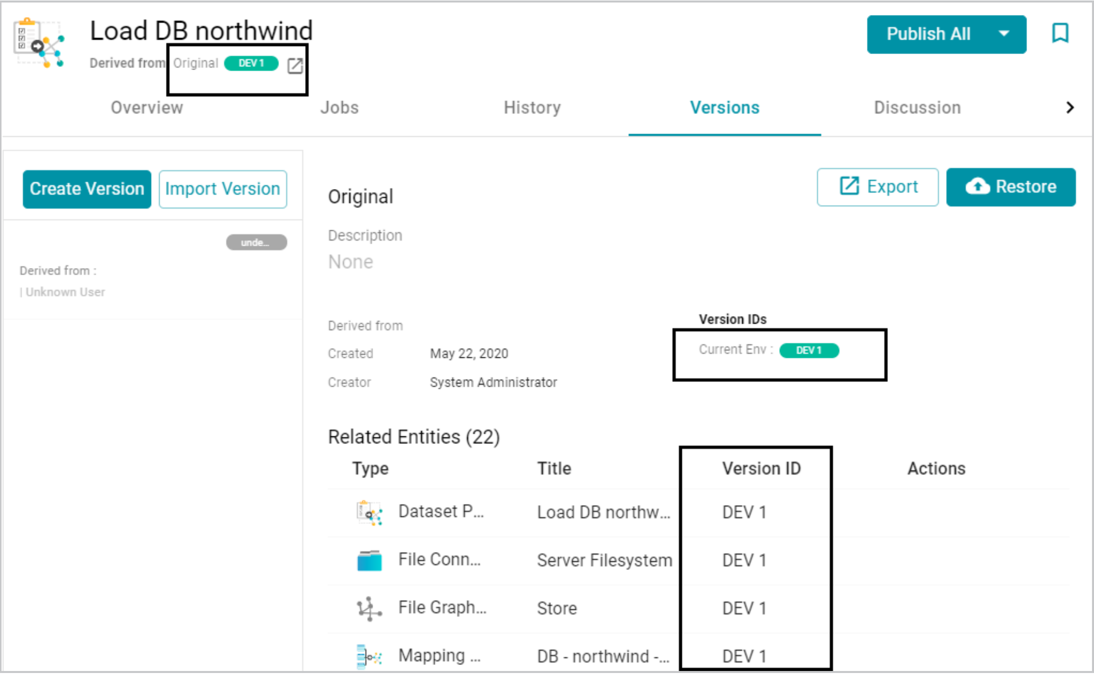
Versioning *Configure the versioning environment*

Versioning Environment: DEV

Edit the value in the **Versioning Environment** field and click **Save**. The images below show examples of the version tags that are controlled by the Versioning Environment setting. This image shows the version at the top of the Anzo application:



For artifact versions, the black rectangles in the image below highlight the areas where the environment version variable value is displayed:



Load DB northwind

Derived from Original DEV 1

Derived from :
| Unknown User

Original

Description
None

Derived from

Created May 22, 2020

Creator System Administrator

Version IDs
Current Env : DEV 1

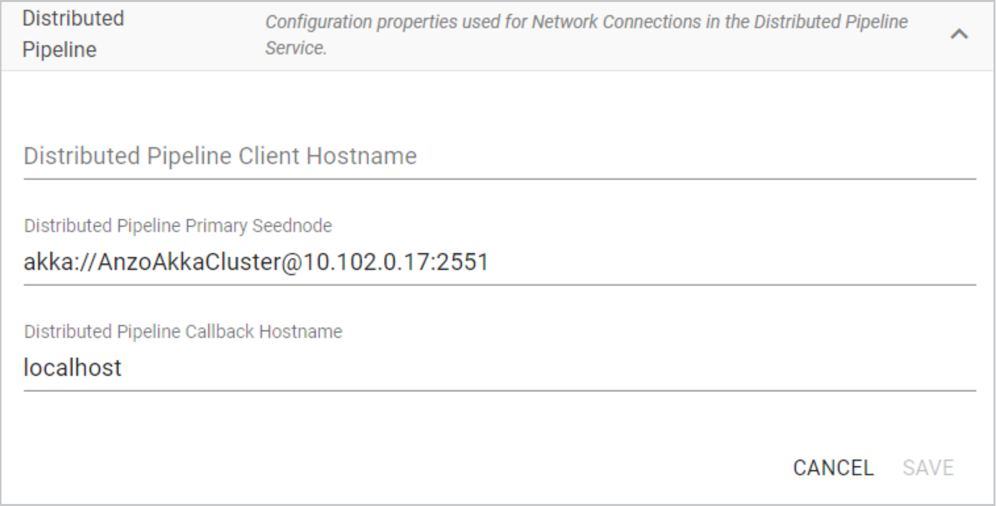
Related Entities (22)

Type	Title	Version ID	Actions
Dataset P...	Load DB northw...	DEV 1	
File Conn...	Server Filesystem	DEV 1	
File Graph...	Store	DEV 1	
Mapping ...	DB - northwind -...	DEV 1	

Configure Network Connections to an Anzo Distributed Unstructured Cluster

To change the network settings for an Anzo Distributed Unstructured cluster, expand **Distributed Pipeline** and click **Edit**.

> **Note**
> If the Kubernetes infrastructure is set up to deploy Anzo Unstructured clusters on-demand, you do not need to configure these settings. For information about Kubernetes-based deployments, see Using K8s for Dynamic Deployments of Anzo Components in the Deployment Guide.



Distributed Pipeline

Configuration properties used for Network Connections in the Distributed Pipeline Service.

Distributed Pipeline Client Hostname

Distributed Pipeline Primary Seednode
akka://AnzoAkkaCluster@10.102.0.17:2551

Distributed Pipeline Callback Hostname
localhost

Modify the settings as needed:

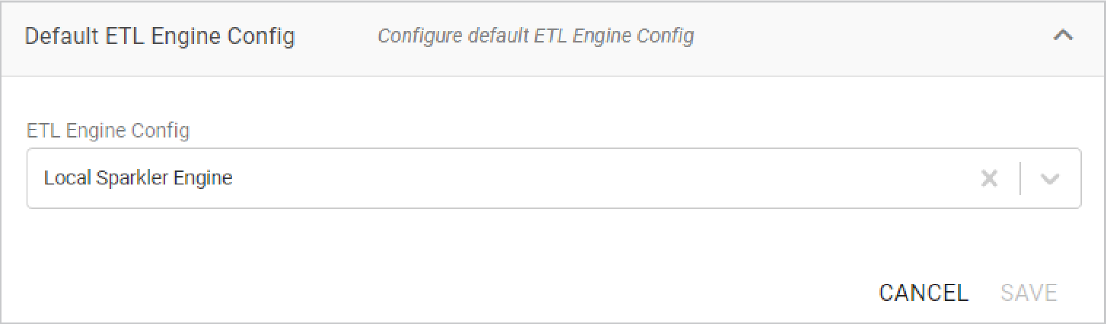
- **Distributed Pipeline Client Hostname**: The hostname or IP address for the Anzo Unstructured leader instance.

  > **Important**
  > The value must be a routable IP address or hostname. If the leader instance is installed on the Anzo host server, specify the IP address or hostname of the server; do not use 127.0.0.1 or localhost.

- **Distributed Pipeline Primary Seednode**: The IP address and port for the leader instance. By default the leader port is **2551**.
- **Distributed Pipeline Callback Hostname**: The hostname or IP address for the Anzo Unstructured leader instance. Typically this is the same value as the **Distributed Pipeline Client Hostname**.

Configure the Default ETL Engine

To set the default ETL engine so that it is automatically selected when users set up ingestion pipelines, expand **Default ETL Engine Config** and click **Edit**.



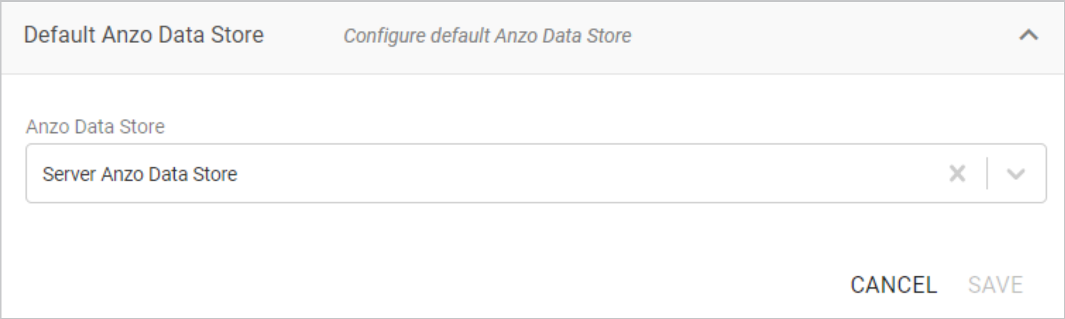
Default ETL Engine Config — *Configure default ETL Engine Config*

ETL Engine Config: Local Sparkler Engine

Click the **ETL Engine Config** drop-down list and select the ETL engine to make the default engine. Then click **Save**.

Configure the Default Anzo Data Store

To set the default Anzo Data Store so that so that it is automatically selected when users set up ingestion pipelines, expand **Default Anzo Data Store** and click **Edit**.



Default Anzo Data Store — *Configure default Anzo Data Store*

Anzo Data Store: Server Anzo Data Store

Click the **Anzo Data Store** drop-down list and select the data store to make the default store. Then click **Save**.

Managing Certificates

The topics in this section provide information about managing server certificates.

Replacing the Anzo Certificate

By default, Anzo installations include a self-signed certificate. Follow the instructions below if you want to replace the default certificate with a trusted one. The steps guide you through using OpenSSL to generate an SSL certificate and signing request and then uploading the signed certificate to Anzo.

- Generating an SSL Certificate and Signing Request
- Uploading a Trusted Certificate to Anzo

Generating an SSL Certificate and Signing Request

1. If necessary, install OpenSSL.
2. Create a request configuration file. For example, create a file called **certificate.cnf**. Then add the following contents to the file. These contents include parameters for creating a multi-domain certificate:

```
# certificate.cnf

[req]
default_bits = 2048
prompt = no
default_md = rsa
req_extensions = req_ext
distinguished_name = dn

[ dn ]
C = <country>
ST = <state>
L = <locality>
O = <organization-or-company-name>
OU = <organizational-unit>
emailAddress = <email-address>
CN = <common-name-or-server-fqdm>

[ req_ext ]
subjectAltName = @alt_names

[ alt_names ]
DNS.1 = <domain1-name-or-ip>
DNS.2 = <domain2-name-or-ip>
DNS.3 = <domain3-name-or-ip>
```

3. Replace the placeholders in the file with the appropriate values. For example:

```
# certificate.cnf

[req]
default_bits = 2048
prompt = no
default_md = rsa
req_extensions = req_ext
distinguished_name = dn

[ dn ]
C = US
ST = MA
L = Boston
O = Cambridge Semantics
OU = IT
emailAddress = webmaster@cambridgesemantics.com
CN = sample.cambridgesemantics.com

[ req_ext ]
subjectAltName = @alt_names

[ alt_names ]
DNS.1 = sample1.domain.com
DNS.2 = 10.0.33.103
DNS.3 = sample3.domain.com
```

4. Run the following command to generate the signing request and private key using the configuration file:

```
openssl req -new -sha256 -nodes -out <csr_file_name>.csr -newkey rsa:2048
-keyout <key_name>.pem -config <config_file_name>.cnf
```

For example:

```
openssl req -new -sha256 -nodes -out anzo-csr.csr -newkey rsa:2048
-keyout anzo-key.pem -config certificate.cnf
```

5. Send the resulting CSR to a certificate authority for signing.

Uploading a Trusted Certificate to Anzo

1. When you receive the signed certificate from the certificate authority, rename the certificate to **anzo-crt.crt**.
2. Then follow the steps below to create a PKCS12 key:
   a. Run the following command to concatenate the signed certificate and private key file that you generated into an `anzo.pem` file:

   ```
   cat <key_name>.pem anzo-crt.crt > anzo.pem
   ```

   For example:

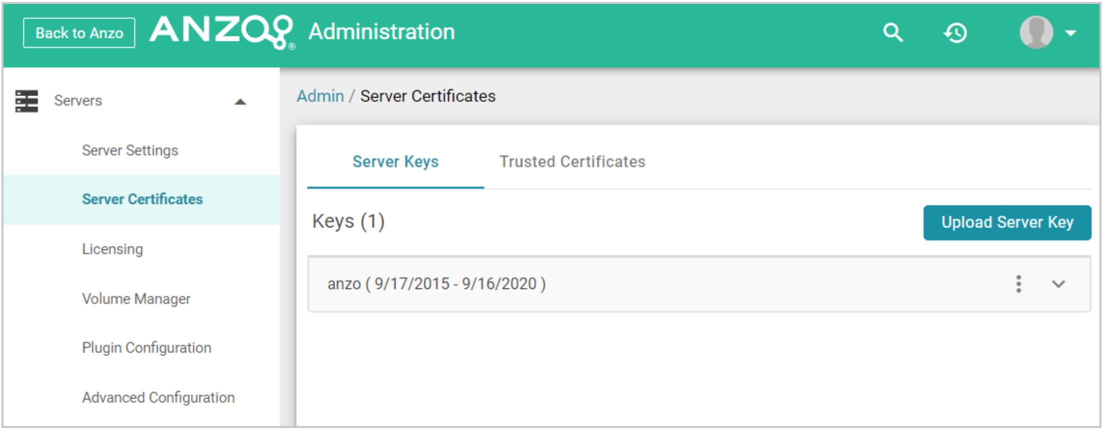
   ```
   cat anzo-key.pem anzo-crt.crt > anzo.pem
   ```

   b. Run the following command to convert the resulting `anzo.pem` file to PKCS12, choose a name for the certificate, and set an export password:

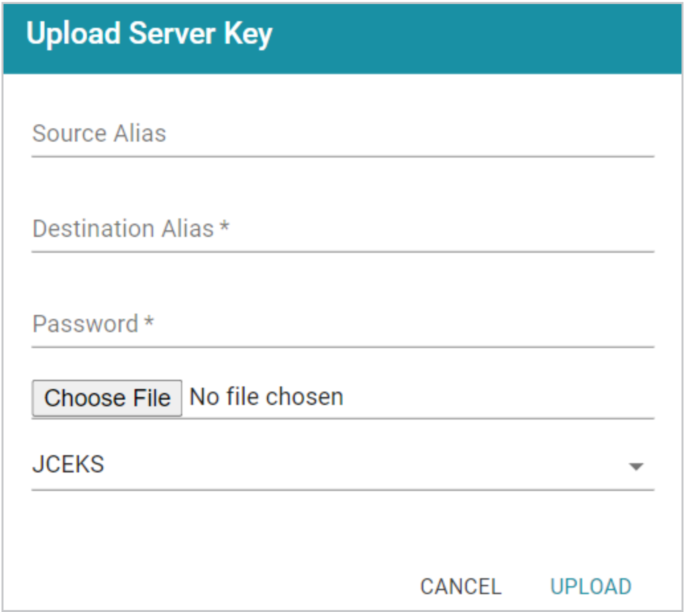
   ```
   openssl pkcs12 -export -in anzo.pem -out anzo.pkcs12 -name
   "<destination_alias>"
   ```

   ```
   Enter Export Password:
   Verifying - Enter Export Password:
   ```

3. Copy the `anzo.pkcs12` certificate to your computer if necessary.
4. In the Administration application, expand the **Servers** menu and click **Server Certificates**. Anzo displays the Server Certificates screen. For example:



5. Click **Upload Server Key**. Anzo displays the Upload Server Key dialog box.

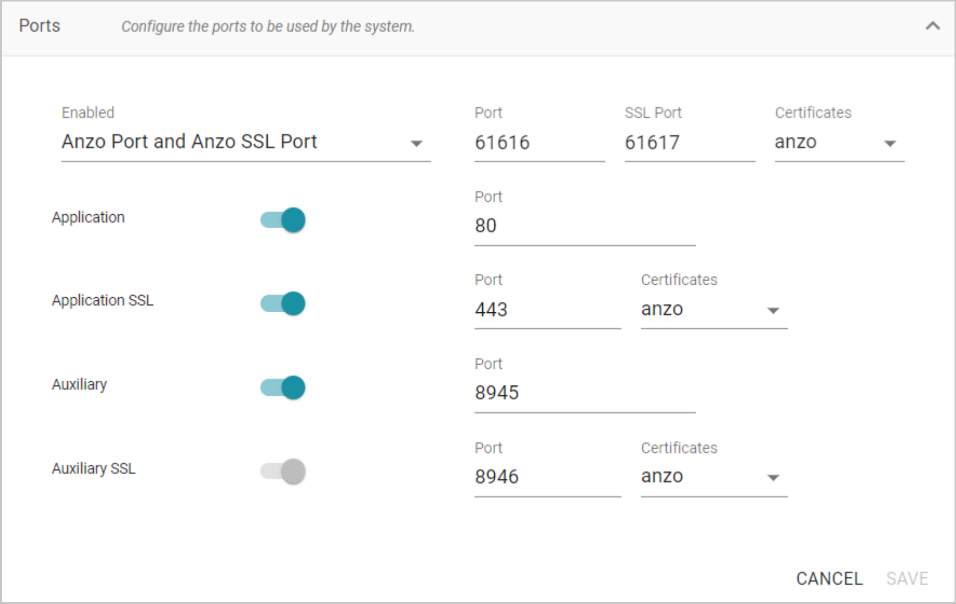


6. Supply the required values:
   - In the **Destination Alias** field, specify the destination alias that you chose when you created the PKCS12 certificate.
   - In the **Password** field, specify the Export Password that you set when you created the PKCS12 certificate.
   - Click the **Choose File** button and select the **anzo.pkcs12** file.
   - Click the **Keystore type** field and select **PKCS12** from the drop-down list.
7. Click **Upload** to upload the certificate.

8. Finally, follow these steps to apply the new certificate to the Anzo server SSL ports:

   a. In the Servers menu, click **Server Settings**.

   b. On the Server Settings screen, expand **Ports** and click **Edit**. For example:



Ports *Configure the ports to be used by the system.*

Enabled: Anzo Port and Anzo SSL Port | Port: 61616 | SSL Port: 61617 | Certificates: anzo

Application | Port: 80

Application SSL | Port: 443 | Certificates: anzo

Auxiliary | Port: 8945

Auxiliary SSL | Port: 8946 | Certificates: anzo

   c. Click the **Certificates** drop-down list for each of the enabled SSL ports and select the new certificate. Then click **Save**.

9. Restart Anzo to apply the configuration change.

Related Topics

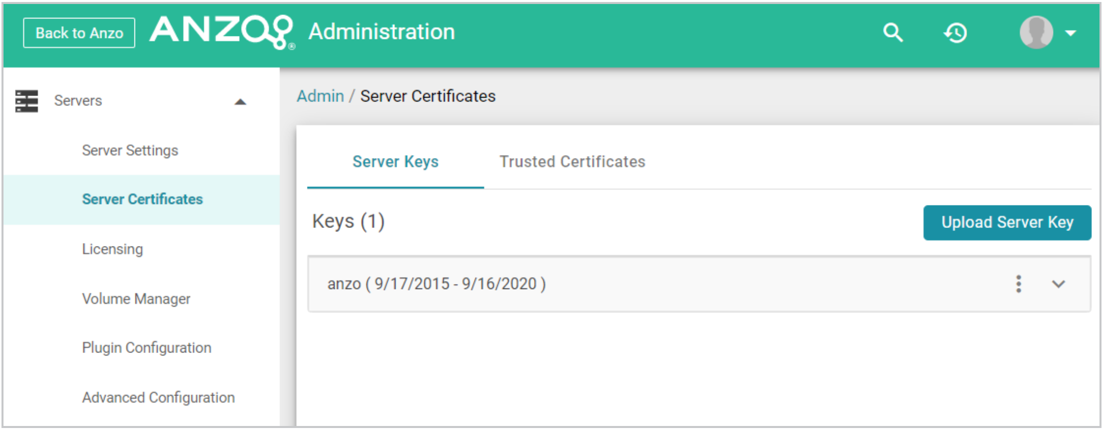
Changing Anzo Server Settings

Adding a Certificate to the Anzo Trust Store

Adding a Certificate to the Anzo Trust Store

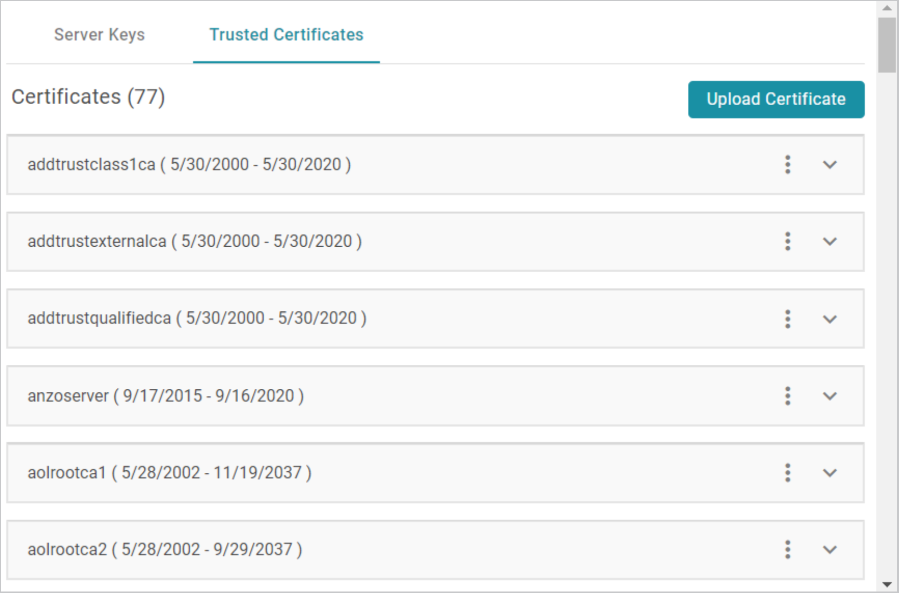
To add a certificate to the Anzo trust store, follow the steps below.

1. In the Administration application, expand the **Servers** menu and click **Server Certificates**. Anzo displays the Server Certificates screen. For example:



anzo (9/17/2015 - 9/16/2020)

2. On the Server Certificates screen, click the **Trusted Certificates** tab. Anzo displays the list of existing certificates. For example:



addtrustclass1ca (5/30/2000 - 5/30/2020)

addtrustexternalca (5/30/2000 - 5/30/2020)

addtrustqualifiedca (5/30/2000 - 5/30/2020)

anzoserver (9/17/2015 - 9/16/2020)

aolrootca1 (5/28/2002 - 11/19/2037)

aolrootca2 (5/28/2002 - 9/29/2037)

3. To upload a new certificate, click the **Upload Certificate** button. Browse to the certificate file, and double-click the file to upload it to Anzo.

Related Topics

Replacing the Anzo Certificate

Updating the Server License

This topic provides important information about licenses and user accounts as well as instructions for updating a license key.

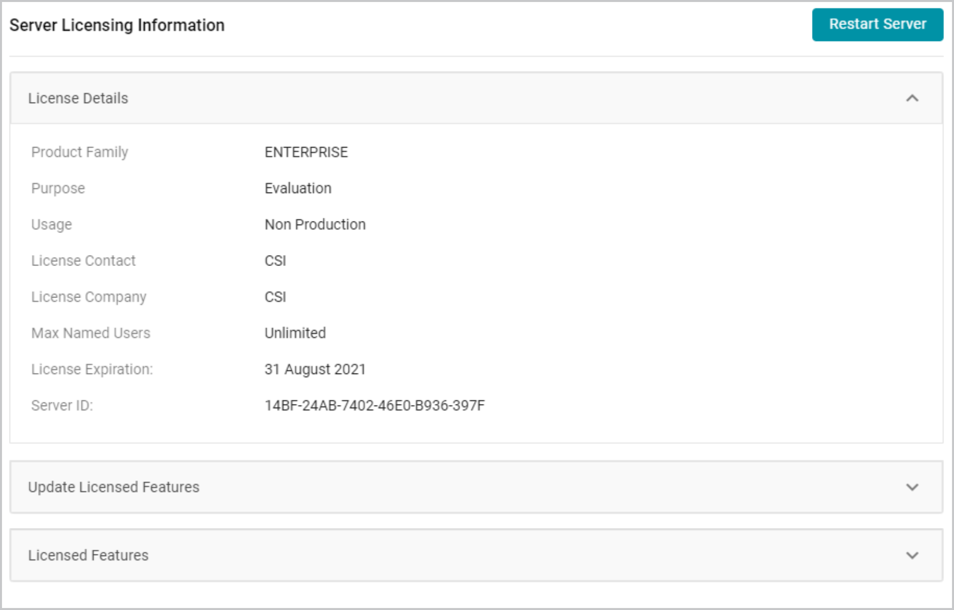
- Updating the License Key
- Licensing and User Account Best Practices

Updating the License Key

Follow the instructions below to update the Anzo server license key.

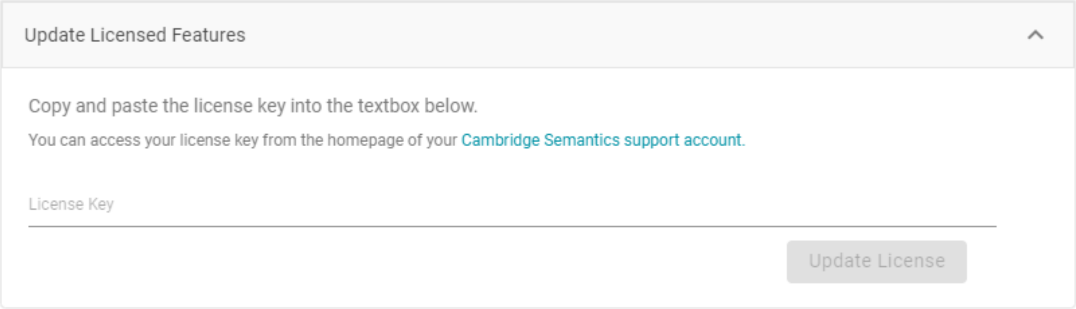
> **Important**
>
> **If your license is expired**, do not follow the steps below. The Server Licensing screen (shown in step 1 below) will be blank except for an "Access Denied/Forbidden License is invalid" error message. To update an expired license, you must stop and restart Anzo from the command line. For example, run `sudo systemctl stop anzo-server` and `sudo systemctl start anzo-server` if you use the Anzo systemd service or `/install_path/Anzo/Server/AnzoServer stop` and `/install_path/Anzo/Server/AnzoServer start` if you do not have the anzo-server service set up.
>
> Once Anzo is restarted, you will be presented with the same license key entry screen that was displayed when Anzo was installed and started for the first time.

1. In the Administration application, expand the **Servers** menu and click **Licensing**. Anzo displays the Server Licensing Information screen. For example:



Product Family	ENTERPRISE
Purpose	Evaluation
Usage	Non Production
License Contact	CSI
License Company	CSI
Max Named Users	Unlimited
License Expiration:	31 August 2021
Server ID:	14BF-24AB-7402-46E0-B936-397F

2. Click **Update Licensed Features** to expand that section of the screen.

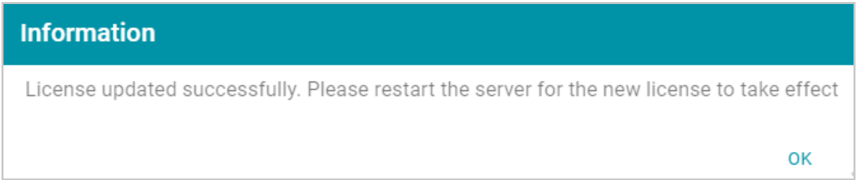


Update Licensed Features

Copy and paste the license key into the textbox below.

You can access your license key from the homepage of your Cambridge Semantics support account.

3. Paste the new license key into the **License Key** field, and then click the **Update License** button. The license is updated but does not take effect until Anzo is restarted. The following dialog box is displayed:



Information

License updated successfully. Please restart the server for the new license to take effect

4. Click **OK** to close the dialog box. Then restart Anzo to apply the license updates. You can click the **Restart Server** button at the top of the screen. For information about other ways to stop and start Anzo, see Starting and Stopping Anzo.

> **Note**
>
> It may take Anzo noticeably longer to start for the first time after the license is updated. Subsequent starts will return to the usual startup time.

Licensing and User Account Best Practices

When Anzo is initially installed, a server ID is generated based on a number of system properties, including the user account that runs the installation script. The Anzo server license is tied to that server ID. If Anzo is re-installed (for instance, during an upgrade) by a different user account, a new server ID is generated and the existing license becomes invalid for the current installation. Whenever you upgrade or re-install Anzo, it is important to use the same user account that was used for the initial installation.

Restoring the Server ID if Anzo is Updated by the Wrong User

If Anzo is updated by a different user, the best way to resolve the issue is to revert the server ID to its original value by rolling back the update:

- If it was a new installation that used the wrong user account, uninstall Anzo. Then change to the correct user and run the installation script again.
- If your backup is a snapshot of the previous application disk, restore the disk. Then change to the correct user and update the installation.
- If it was an upgrade that used the wrong user account, restore Anzo from the backup that was saved before the upgrade:

  If your backup is a copy of the Anzo system journal, follow these steps:

  a. Uninstall Anzo.
  b. Change to the correct user account.
  c. Reinstall the previous version of Anzo using the original installation script.
  d. After the installation, replace the **anzo.jnl** file in the `install_path/Server/data/journal` directory with the backup version of the file.

     At this point, Anzo is restored to the previous version and has the server ID that is associated with the license.
  e. Now Anzo can be re-upgraded to the later release.

  If your backup is a copy of the entire Anzo installation directory, follow these steps:

  a. Uninstall Anzo.
  b. Change to the correct user account.
  c. Move the copy of the previous Anzo installation directory to the original location on the file system.

At this point, Anzo is restored to the previous version and has the server ID that is associated with the license.

d. Now Anzo can be re-upgraded to the later release.

> **Important**
> Cambridge Semantics strongly recommends that you do NOT change the user running Anzo. If it is absolutely necessary, the license can be changed so that it is associated with the new server ID, and Anzo can be restarted once the license is updated. However, using a new server ID resets (or regenerates from non-customer-specific templates) all previously configured OSGI properties to their default values. Changing the Anzo user should only be attempted if there is a complete record of all of the customized OSGI properties and their values as well as a thorough change log so that the configuration can be restored if necessary.

Managing Volumes

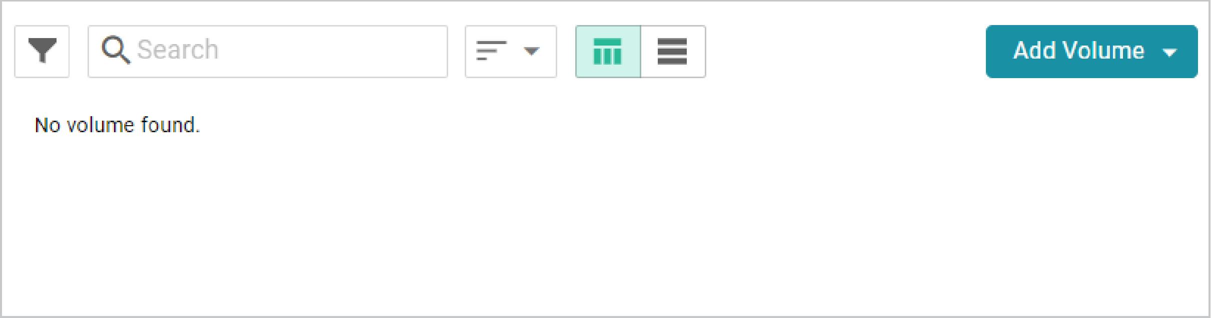
The topics in this section provide information about creating new volumes (also known as journals or database instances) and mounting existing volumes.

Creating a New Volume

This topic provides instructions for creating new volumes or journals.

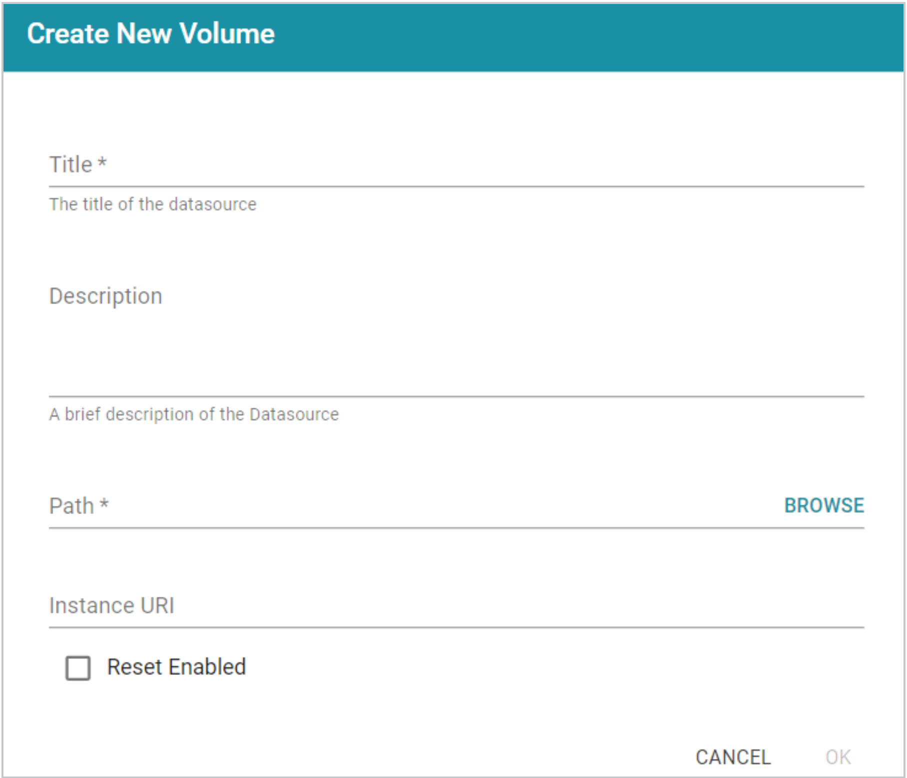
> **Note**
> The number of volumes that you can create depends on your software license. For more information, contact Cambridge Semantics Support.

1. In the Administration application, expand the **Servers** menu and click **Volume Manager**. Anzo displays the Volume Manager screen, which lists any existing user-defined volumes (system volumes can be displayed by selecting the system data filter). For example:

   No volume found.

2. Click the **Add Volume** button and select **Add Volume**. Anzo displays the Create New Volume dialog box.



Create New Volume

Title *

The title of the datasource

Description

A brief description of the Datasource

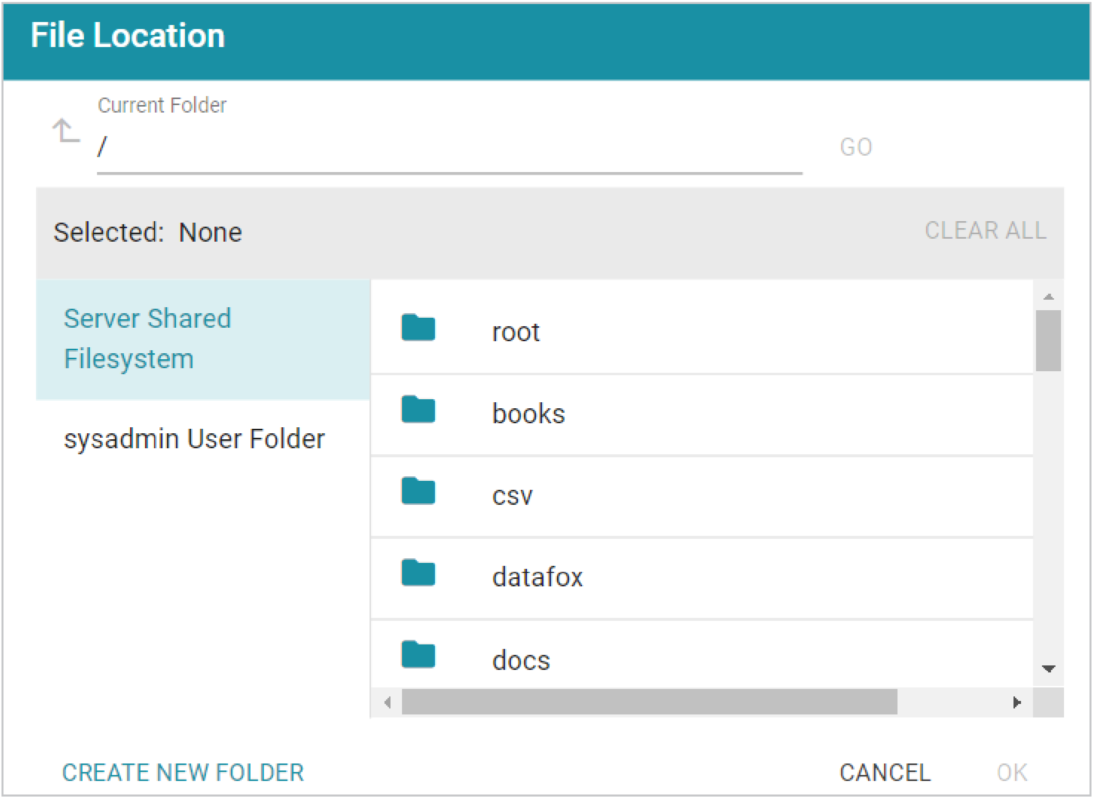
Path *

Instance URI

□ Reset Enabled

3. In the **Title** field, type a name for the new volume, and type an optional description in the **Description** field.

4. Click the **Path** field to open the File Location dialog box. For example:



File Location

Current Folder

/

Selected: None

5. On the left side of the screen, select the file store where you want to create this volume. On the right side of the screen, select the directory where you want Anzo to save the volume. If needed, you can click **Create New Folder** to create a new directory. Then click **OK** to close the File Location dialog box.
6. On the Create New Volume screen, complete the remaining fields:
   - **Instance URI**: Anzo automatically assigns an instance URI to this volume. If you want to specify a custom URI, type the URI in this field.
   - **Reset Enabled**: Specifies whether to enable resets. When reset is enabled, the option to reset the entire contents of the volume becomes available. To enable resets for this volume, select the **Reset Enabled** checkbox.
7. Click **Save** to create the new volume in the location that you specified.

Related Topics

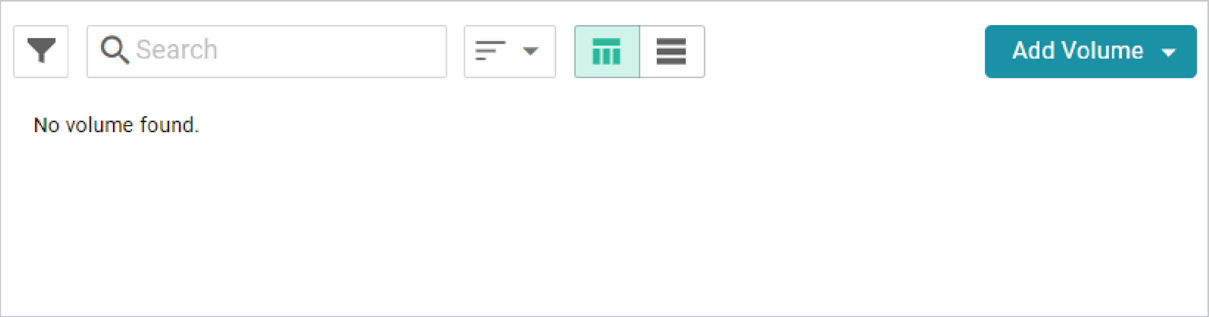
Mounting an Existing Volume

Mounting an Existing Volume

This topic provides instructions for mounting an existing volume or journal.

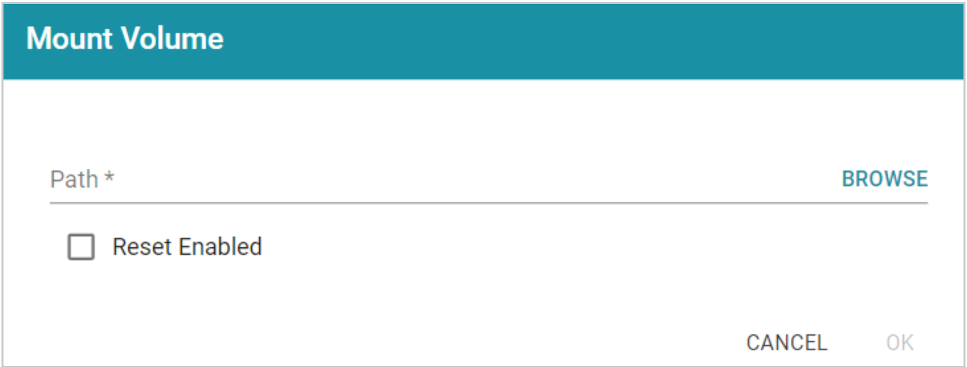
> **Note**
> The number of volumes that you can mount depends on your software license. For more information, contact Cambridge Semantics Support.

1. In the Administration application, expand the **Servers** menu and click **Volume Manager**. Anzo displays the Volume Manager screen, which lists any existing user-defined volumes (system volumes can be displayed by selecting the system data filter). For example:

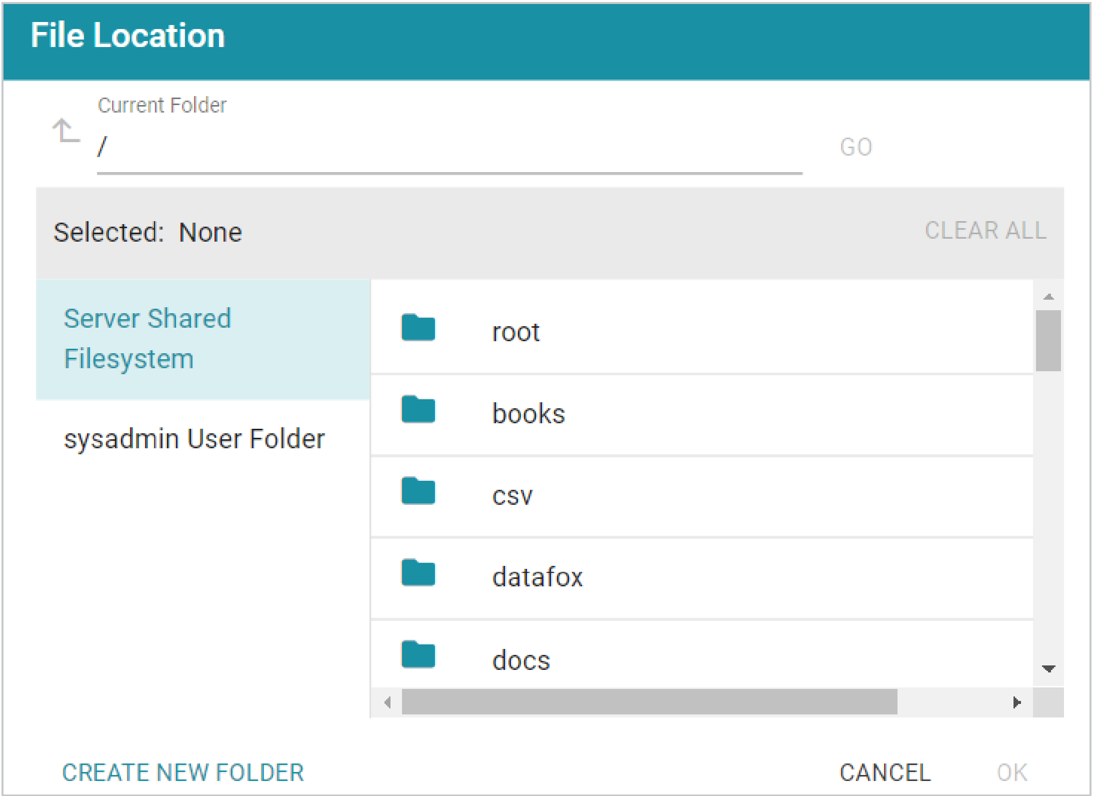
   

   No volume found.

2. Click the **Add Volume** button and select **Mount Volume**. Anzo displays the Mount Volume screen.

3. Click the **Path** field to open the File Location dialog box. For example:



Current Folder: /

Selected: None

4. On the left side of the screen, select the file store that hosts the volume (.jnl file) that you want to mount. On the right side of the screen, navigate to the .jnl file and select it. Then click **OK**. Anzo mounts the new volume.

Related Topics

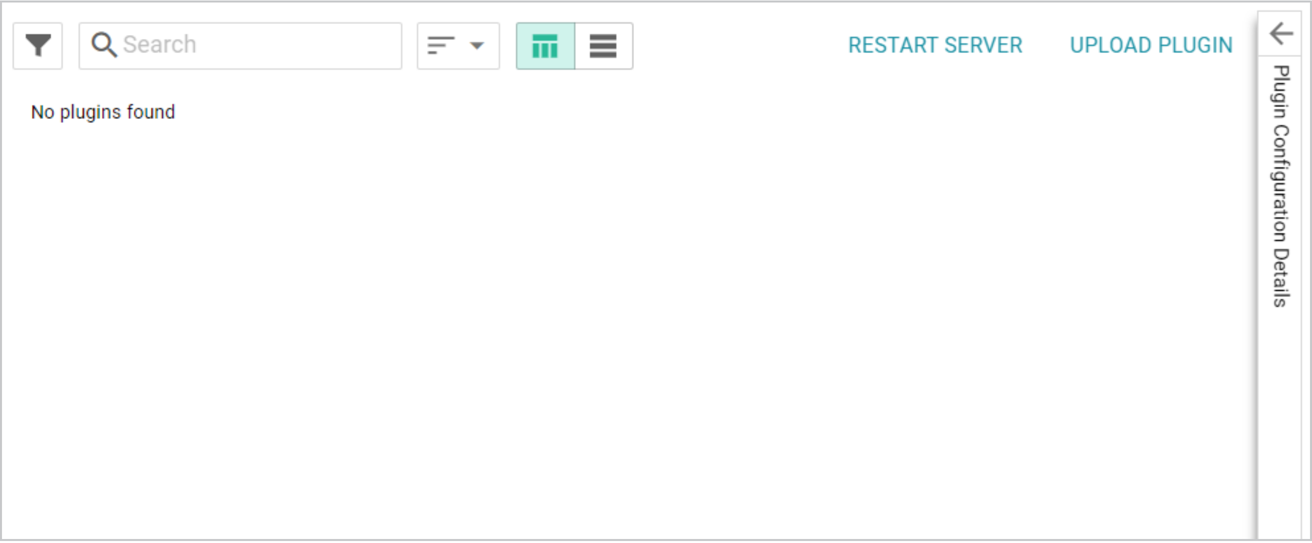
Creating a New Volume

Uploading a Plugin

When connecting to a relational database to import data, you may need to upload a JDBC driver to Anzo. You may also need to import custom bundles or other bundles received from Cambridge Semantics. This topic provides instructions for uploading executable .jar files from your computer to Anzo.

> **Note**
> Not all .jar files are compatible with Anzo. Custom drivers need to be converted to an OSGI bundle before they are uploaded. For more information and instructions on creating an OSGI bundle from a .jar file, contact your Cambridge Semantics Customer Success manager.

1. In the Administration application, expand the **Servers** menu and click **Plugin Configuration**. Anzo displays the Plugin Configuration screen. For example:



No plugins found

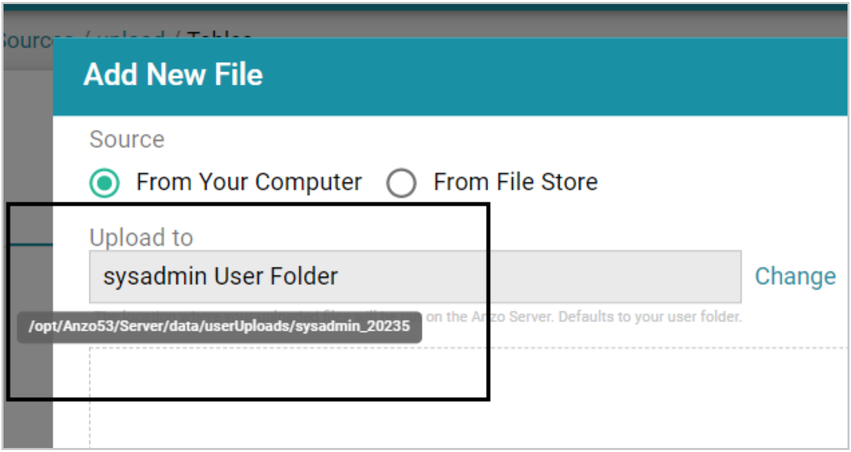
2. In the top right corner, click **Upload Plugin**. The application opens the file browser on your computer.
3. In the file browser, navigate to the .jar file to upload, and then double-click the file to upload it. Anzo uploads the file and displays a "Completed" message. You do not need to restart Anzo to apply the new executable.

Advanced Configuration of Semantic Services

The topics in the section provide instructions for making the types of semantic service or application configuration changes that are commonly desired.

Setting the Default Base File Store Path for File Uploads

By default, if a user uploads a file (such as a CSV, XML, or JSON file) to a data source from their computer, Anzo is configured to copy the file to the server's data directory, `<install_path>/Anzo/Server/data/userUploads`. This is the path that is selected by default in the **Upload To** field on the Add New File screen. For example, the image below shows the default upload path for the sysadmin user:



Upload to: sysadmin User Folder

/opt/Anzo53/Server/data/userUploads/sysadmin_20235

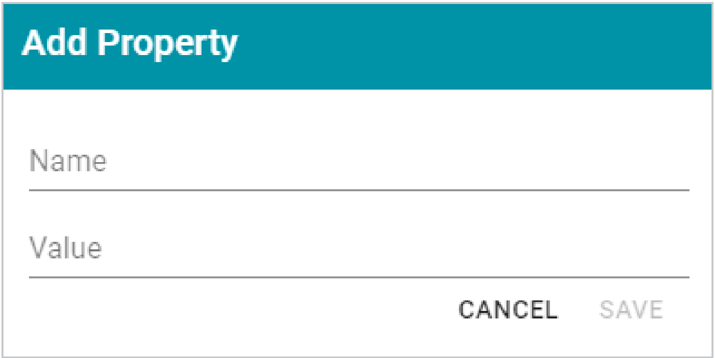
on the Anzo Server. Defaults to your user folder.

When the file is in the server installation path and not the shared File Store it is not accessible by applications like AnzoGraph or Spark. In addition, other users cannot publish pipelines for that Data Source because they typically do not have access to the file. Source files that are routinely updated and re-ingested should be hosted on the shared File Store.

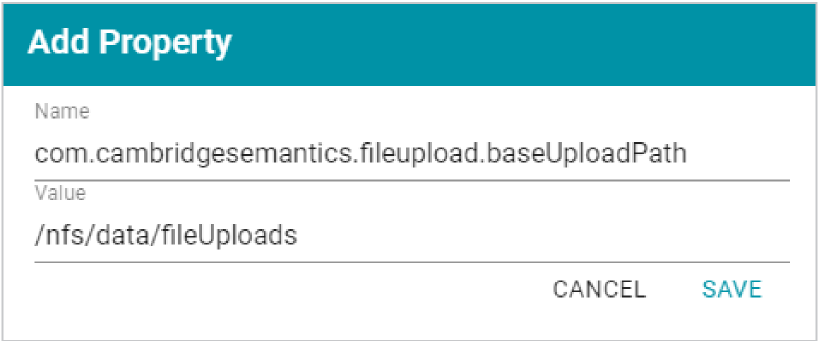
Follow the instructions below to configure the base upload path so that it points to a location on the File Store by default.

1. If necessary, create a directory on the shared File Store that you can designate as the base location for saving uploaded files.
2. In the Administration application, expand the **Servers** menu and click **Advanced Configuration**. Click **I understand and accept the risk**.
3. Search for the **Anzo File Upload** bundle and view its details.
4. Click the **Services** tab and expand the **com.cambridgesemantics.anzo.fileupload.FileUploadServlet** service.

5. Click **Add Property** next to the service name. Anzo opens the Add Property dialog box.



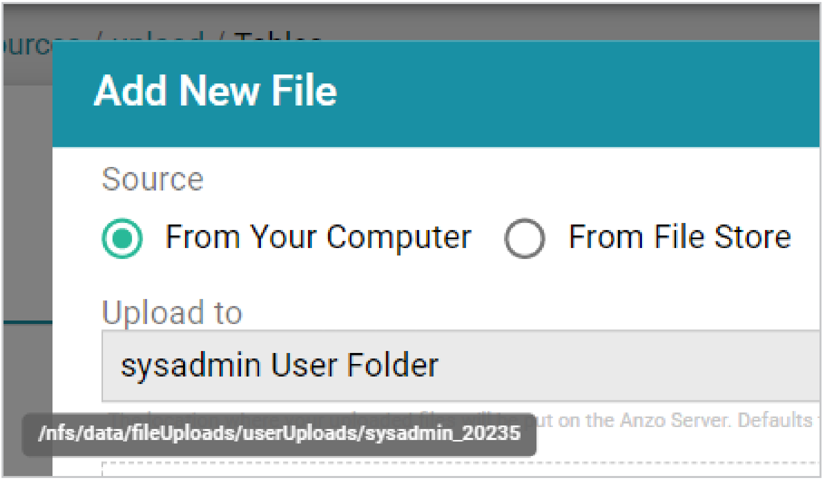
6. In the **Name** field, specify **com.cambridgesemantics.fileupload.baseUploadPath**, and then set the **Value** to the location on the file store where uploaded files should be saved. The base directory that you specify must exist on the file store. For example:



Name
com.cambridgesemantics.fileupload.baseUploadPath
Value
/nfs/data/fileUploads

7. Click **Save** to add the new property. And restart Anzo to apply the configuration changes.

When the base upload path is configured, the location that you specified becomes the default path in the Upload To field on the Add New File dialog box. For example, the image below shows the Add New File screen for the sysadmin user when baseUploadPath is set to **/nfs/data/fileUploads**.



Upload to
sysadmin User Folder

/nfs/data/fileUploads/userUploads/sysadmin_20235

ut on the Anzo Server. Defaults

Enabling and Configuring the System Monitor Service

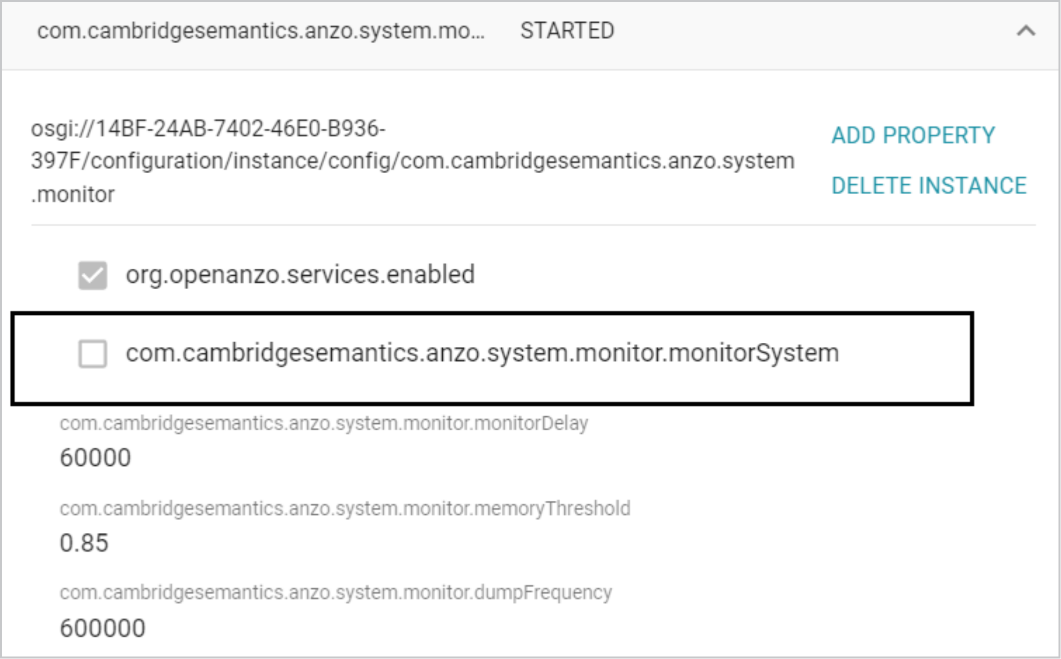
The System Monitor service, which monitors the state of the Java virtual machine (JVM), is disabled by default. You can enable the service to poll the state of the JVM at a certain interval and capture stack and heap dumps when memory utilization increases beyond a specified threshold. This topic provides instructions for enabling the service and configuring its options.

- Enabling the System Monitor Service
- Configuring the System Monitor Service

Enabling the System Monitor Service

Follow the steps below to enable the System Monitor.

1. In the Administration application, expand the **Servers** menu and click **Advanced Configuration**. Click **I understand and accept the risk**.
2. Search for the **Anzo System Monitor** bundle and view its details.
3. Click the **Services** tab and expand **System Monitor Activator**.
4. Locate the **com.cambridgesemantics.anzo.system.monitor.monitorSystem** property (shown in the image below).



com.cambridgesemantics.anzo.system.mo... STARTED

osgi://14BF-24AB-7402-46E0-B936-397F/configuration/instance/config/com.cambridgesemantics.anzo.system.monitor

☑ org.openanzo.services.enabled

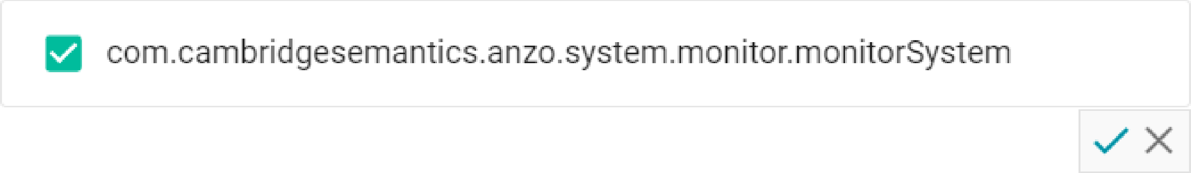
☐ com.cambridgesemantics.anzo.system.monitor.monitorSystem

com.cambridgesemantics.anzo.system.monitor.monitorDelay
60000

com.cambridgesemantics.anzo.system.monitor.memoryThreshold
0.85

com.cambridgesemantics.anzo.system.monitor.dumpFrequency
600000

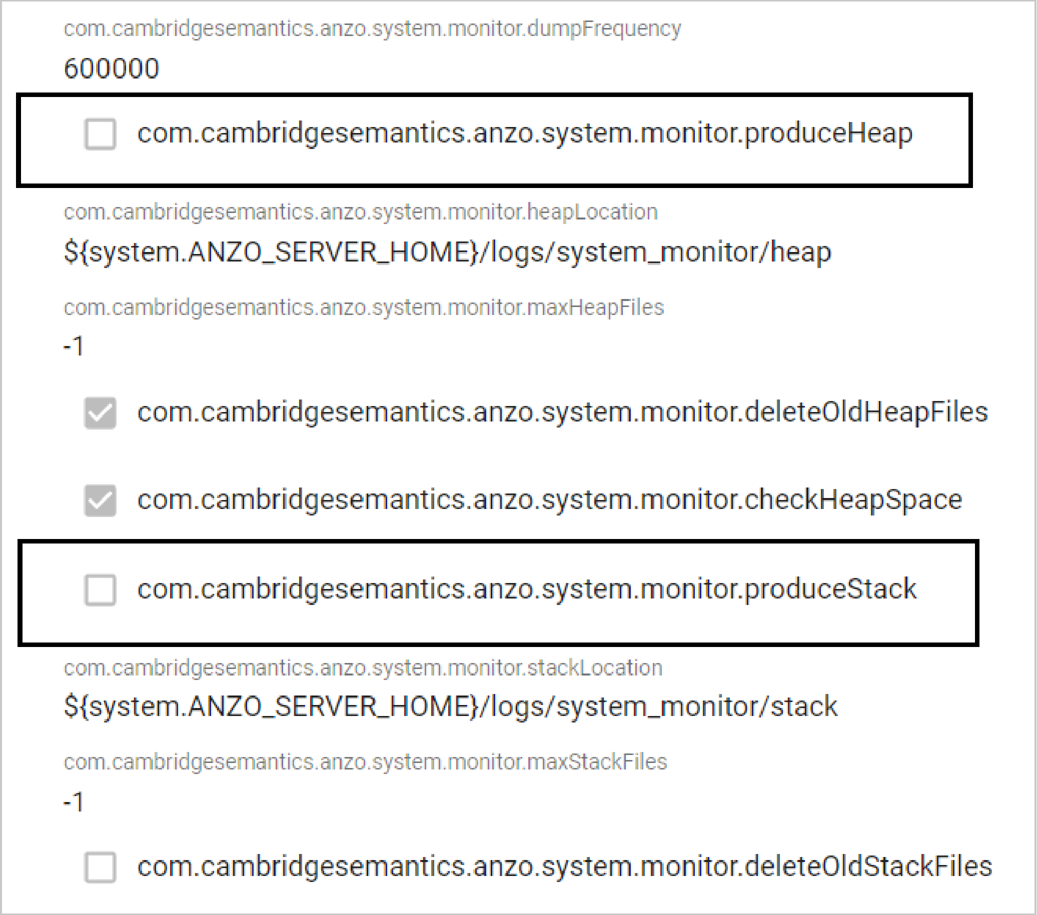
5. Click the property to make it editable, and then select the checkbox to enable it.



☑ com.cambridgesemantics.anzo.system.monitor.monitorSystem

6. Click the checkmark icon (✓) for that property to save the change.

7. Next, configure the service to dump the stack and/or heap logs to disk by enabling the properties under the **monitorSystem** property:



com.cambridgesemantics.anzo.system.monitor.dumpFrequency
600000

□ com.cambridgesemantics.anzo.system.monitor.produceHeap

com.cambridgesemantics.anzo.system.monitor.heapLocation
${system.ANZO_SERVER_HOME}/logs/system_monitor/heap

com.cambridgesemantics.anzo.system.monitor.maxHeapFiles
-1

☑ com.cambridgesemantics.anzo.system.monitor.deleteOldHeapFiles

☑ com.cambridgesemantics.anzo.system.monitor.checkHeapSpace

□ com.cambridgesemantics.anzo.system.monitor.produceStack

com.cambridgesemantics.anzo.system.monitor.stackLocation
${system.ANZO_SERVER_HOME}/logs/system_monitor/stack

com.cambridgesemantics.anzo.system.monitor.maxStackFiles
-1

□ com.cambridgesemantics.anzo.system.monitor.deleteOldStackFiles

To create heap dumps, enable **com.cambridgesemantics.anzo.system.monitor.produceHeap**. To create stack dumps, enable **com.cambridgesemantics.anzo.system.monitor.produceStack**.

8. You can restart Anzo to enable the service without performing additional configuration. Or see Configuring the System Monitor Service below for information about the configuration options.

Configuring the System Monitor Service

By default, the System Monitor Service is configured to monitor memory usage and take the following actions:

- Every **60 seconds** (60000 milliseconds), evaluate whether a stack or thread dump should be written.
- Write stack and/or heap dumps if the memory threshold reaches **85%** (0.85).
- Continue to write stack and/or heap dumps at an interval of every **10 minutes** (600000 milliseconds) as long as memory usage remains at or above the threshold.
- Save heap and stack dumps in the `<install_path>/Server/logs/system_monitor/heap` and `stack` directories.

To modify the characteristics described above, you can change the values for the following properties:

- To change the frequency with which memory usage is evaluated to see if it has reached the threshold, update the **com.cambridgesemantics.anzo.system.monitor.monitorDelay** property. Specify the number of milliseconds to wait between checks.
- To change the memory threshold, update the **com.cambridgesemantics.anzo.system.monitor.memoryThreshold** property. Specify the percent of total memory as a decimal value.
- To change how often stack and/or heap dumps are written when memory usage is above the threshold, update the **com.cambridgesemantics.anzo.system.monitor.dumpFrequency** property. Specify the number of milliseconds to wait between dumps.
- To change the location where heap and/or stack dumps are saved, update the **com.cambridgesemantics.anzo.system.monitor.heapLocation** and/or **com.cambridgesemantics.anzo.system.monitor.stackLocation** property to specify an alternate path and directory.

After changing any of the properties, make sure that you restart Anzo to apply the configuration change.

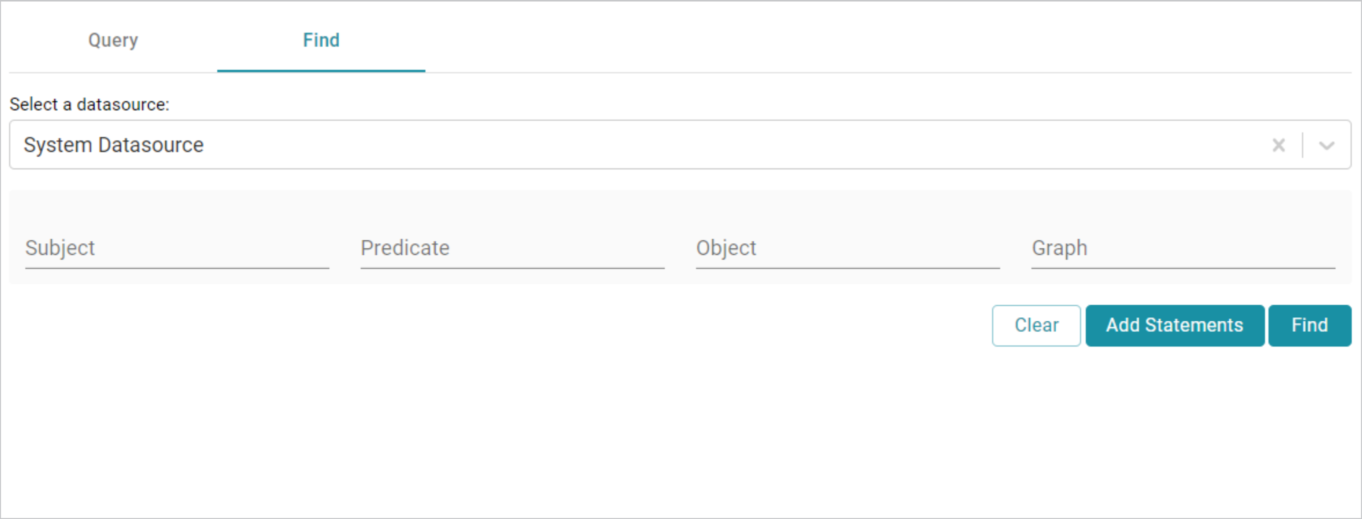
Related Topics

Viewing the Current Stack in a Browser

Routing Hi-Res Analytics to a Custom URL

If you have a custom skin or personality for the Hi-Res Analytics application, and you want those customizations to be loaded automatically when users access the application, you can configure the Anzo application to re-route users to the preferred URL. Follow the instructions below to change the entry points to the Hi-Res application in the Anzo application. The instructions use the Find feature in the Query Builder to find and modify the object of the Hi-Res Analytics routing property.

1. In the Anzo application, expand the **Access** menu and click **Query Builder**.
2. In the Query Builder, click the **Find** tab. The Find screen is displayed with the **System Datasource** selected as the Source.



Select a datasource: System Datasource

Subject

Predicate

Object

Graph

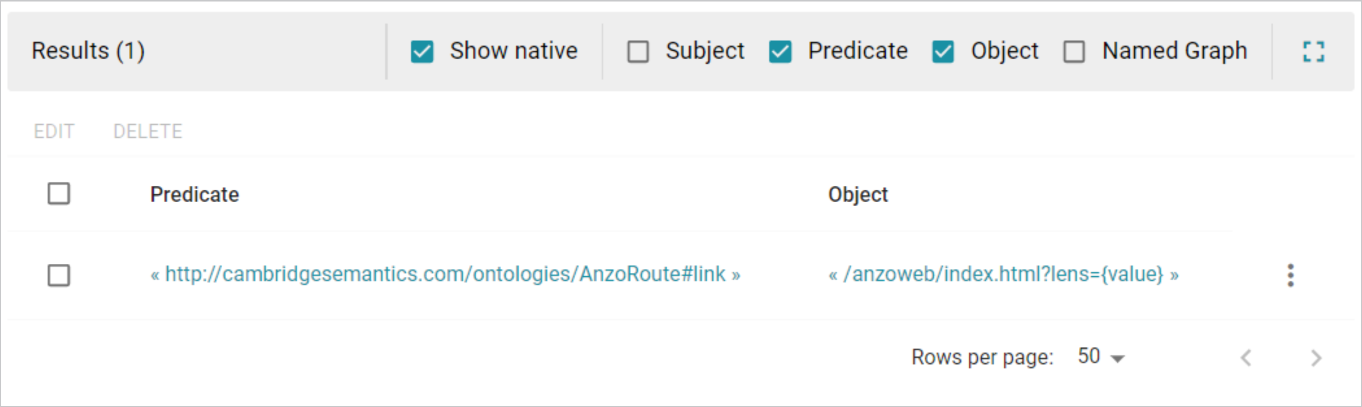
3. In the **Subject** field, specify the following URI:

```
http://cambridgesemantics.com/Routes/sdi/hi-res-analytics-urn
```

4. In the **Predicate** field, specify this URI:

```
http://cambridgesemantics.com/ontologies/AnzoRoute#link
```

5. Click **Find** to display the quads with the specified subject and predicate. You can clear the **Subject** and **Named Graph** Quick Filter checkboxes to make the results easier to read. For example:



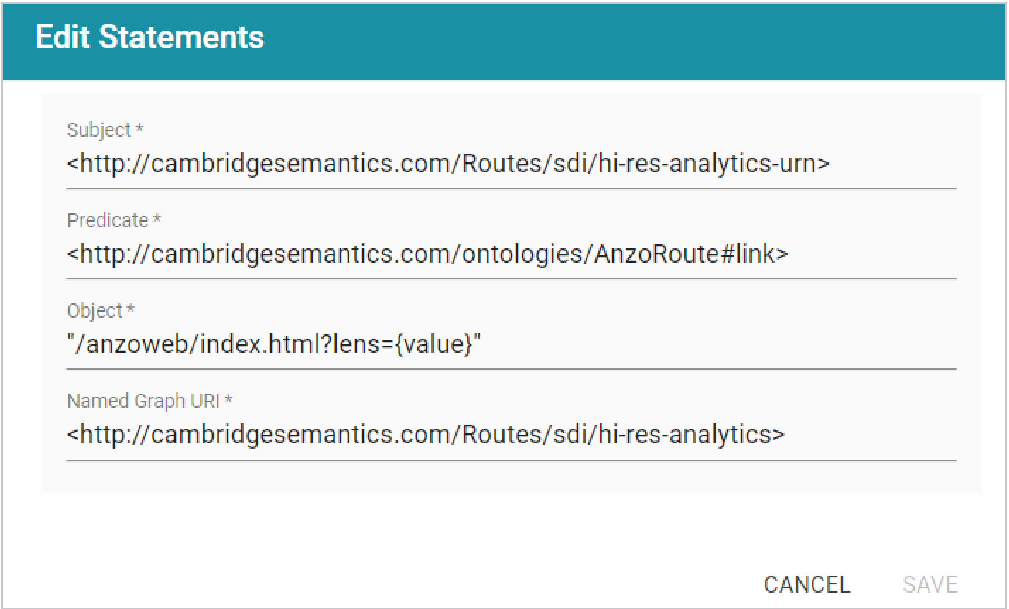
Results (1) ☑ Show native □ Subject ☑ Predicate ☑ Object □ Named Graph

EDIT DELETE

□	Predicate	Object
□	« http://cambridgesemantics.com/ontologies/AnzoRoute#link »	« /anzoweb/index.html?lens={value} »

Rows per page: 50

6. Click the menu icon (⋮) for the quad and select **Edit**. Anzo opens the Edit Statements dialog box.



Edit Statements

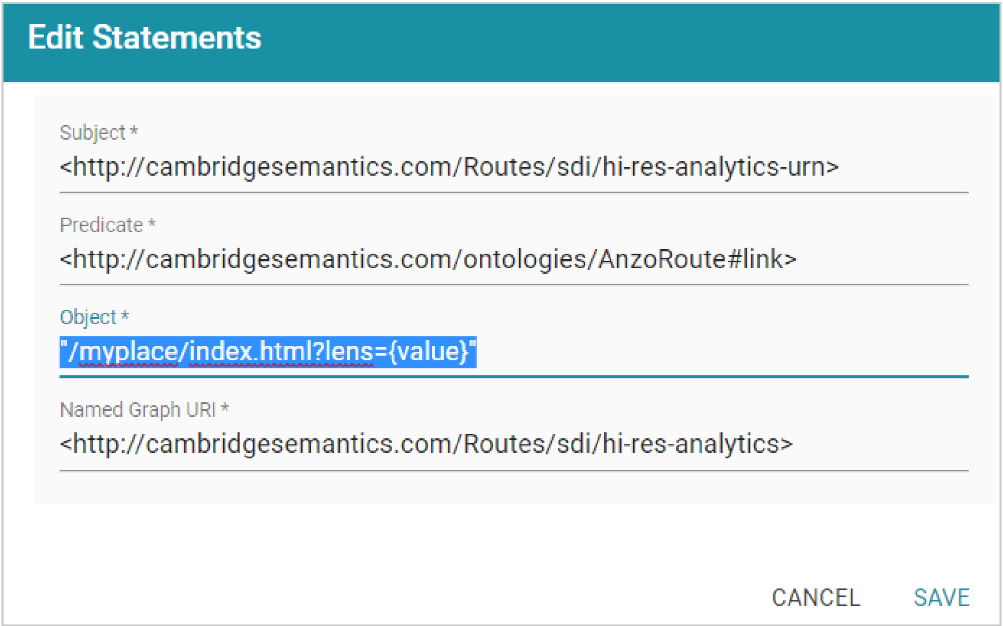
Subject *
<http://cambridgesemantics.com/Routes/sdi/hi-res-analytics-urn>

Predicate *
<http://cambridgesemantics.com/ontologies/AnzoRoute#link>

Object *
"/anzoweb/index.html?lens={value}"

Named Graph URI *
<http://cambridgesemantics.com/Routes/sdi/hi-res-analytics>

7. In the Edit Statement dialog box, replace the **Object** value (**"/anzoweb/index.html?lens={value}"**) with the URL that you want to route users to. For example: **"/myplace/index.html?lens={value}"**.



Edit Statements

Subject *
<http://cambridgesemantics.com/Routes/sdi/hi-res-analytics-urn>

Predicate *
<http://cambridgesemantics.com/ontologies/AnzoRoute#link>

Object *
"/myplace/index.html?lens={value}"

Named Graph URI *
<http://cambridgesemantics.com/Routes/sdi/hi-res-analytics>

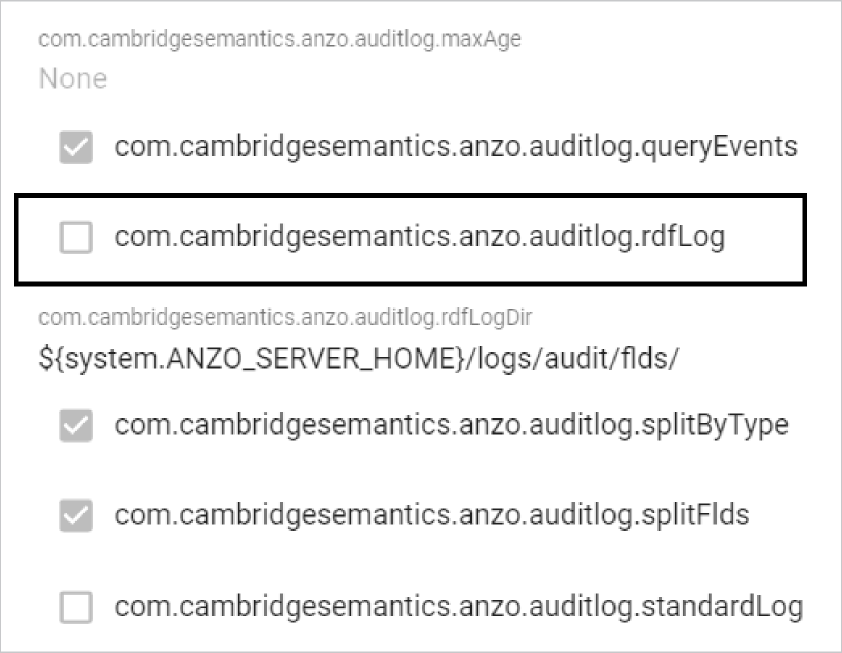
8. Click **Save** to apply the change and return to the Find screen.

The Anzo application is now configured to route users to the custom URL if they open the Hi-Res Analytics application from the Home page, open a dashboard from the Hi-Res Analytics screen, or click **Create Dashboard** from a Graphmart screen.

Separating Audit Logs by Type of Event

By default, when Audit Log Packages, such as UserAudit, are enabled and set to Log Level **Info**, all types of audit events are logged to a single file: **anzo_audit_info.log**. You have the option, however, to configure Anzo to create and store smaller audit logs by generating separate files in subdirectories that are sorted by event type, such as userEvents, queryEvents, accessEvents, etc. Follow the instructions below to enable this option:

1. In the Administration application, expand the **Servers** menu and click **Advanced Configuration**. Click **I understand and accept the risk**.
2. Search for the **Anzo Audit Logging Framework** bundle and view its details.
3. Click the **Services** tab and expand **com.cambridgesemantics.anzo.AuditLog**.
4. Find the **com.cambridgesemantics.anzo.auditlog.rdfLog** property (shown below).



com.cambridgesemantics.anzo.auditlog.maxAge
None

☑ com.cambridgesemantics.anzo.auditlog.queryEvents

□ com.cambridgesemantics.anzo.auditlog.rdfLog

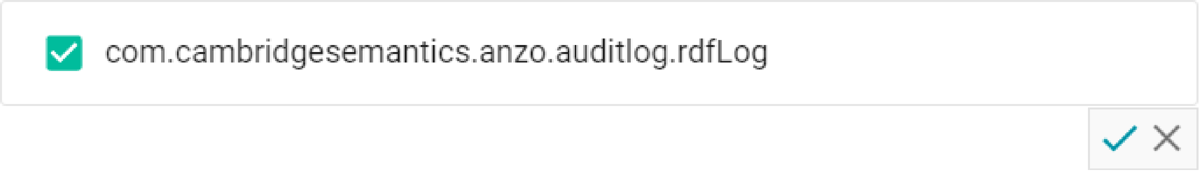
com.cambridgesemantics.anzo.auditlog.rdfLogDir
${system.ANZO_SERVER_HOME}/logs/audit/flds/

☑ com.cambridgesemantics.anzo.auditlog.splitByType

☑ com.cambridgesemantics.anzo.auditlog.splitFlds

□ com.cambridgesemantics.anzo.auditlog.standardLog

5. Click the property to make it editable, and then select the checkbox to enable it.



☑ com.cambridgesemantics.anzo.auditlog.rdfLog

6. Click the checkmark icon (✓) to save the change.
7. Restart Anzo to apply the configuration changes.

Once new audit events are triggered, an `audit/audit-flds` subdirectory is created in the `<install_path>/Server/logs` directory. And audit logs will be created in the userEvents, queryEvents, accessEvents, etc. subdirectories.

Related Topics

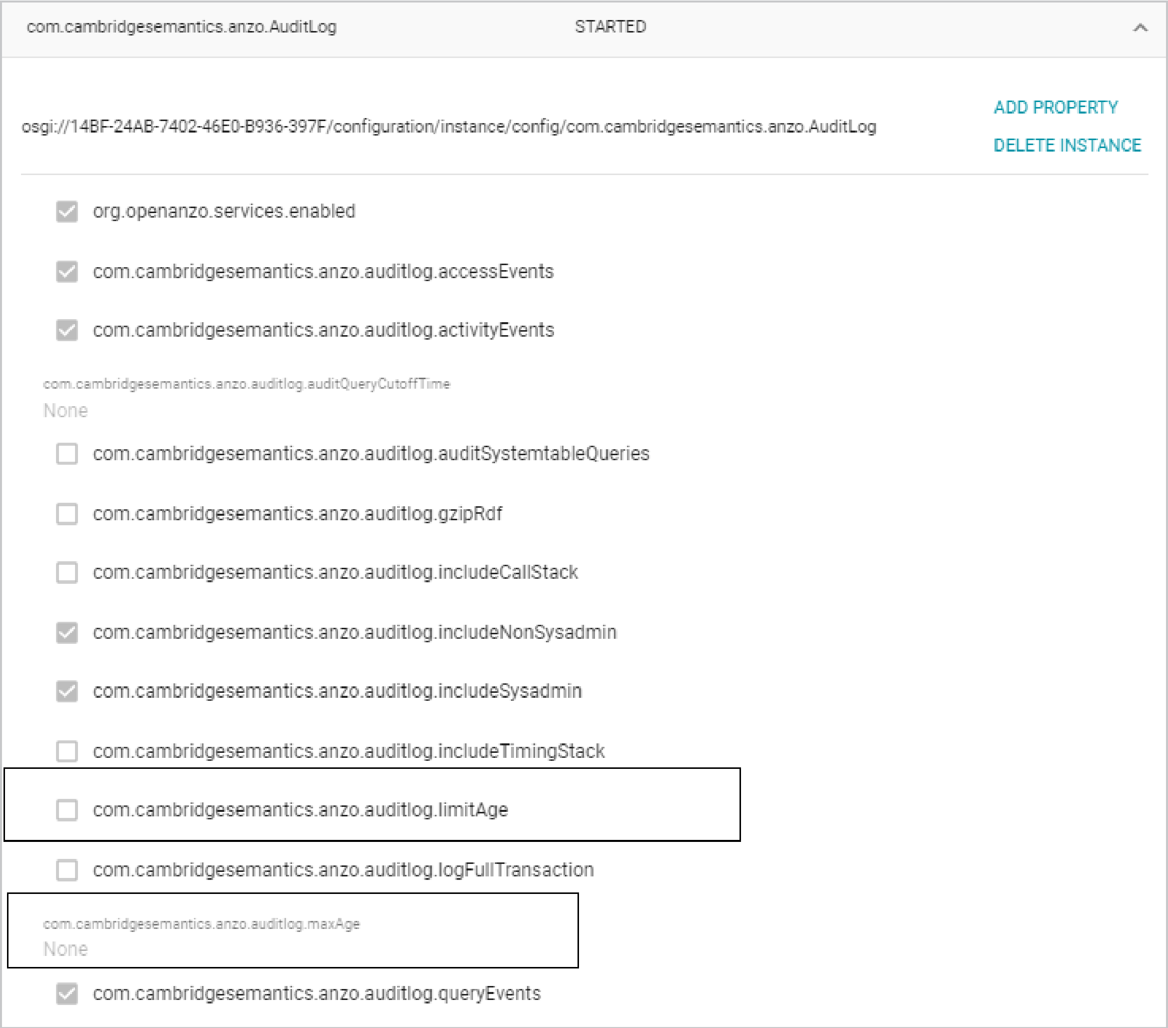
System Query Audit

Enabling and Viewing Audit Logs

Limiting the Age (and Size) of Audit Logs

If you want to retain all of the audit log data but work with smaller data sets when loading and analyzing the log, you can configure Anzo to add an age limit (in days) to audit log data sets. Once an audit log data set reaches that age, Anzo stops writing to it and a new audit log data set is started. Follow the instructions below to configure the audit log service to add an age limit.

1. In the Administration application, expand the **Servers** menu and click **Advanced Configuration**. Click **I understand and accept the risk**.
2. Search for the **Anzo Audit Logging Framework** bundle and view its details.
3. Click the **Services** tab and expand **com.cambridgesemantics.anzo.AuditLog**.
4. Find the **limitAge** and **maxAge** properties (shown below).



com.cambridgesemantics.anzo.AuditLog STARTED

osgi://14BF-24AB-7402-46E0-B936-397F/configuration/instance/config/com.cambridgesemantics.anzo.AuditLog

☑ org.openanzo.services.enabled

☑ com.cambridgesemantics.anzo.auditlog.accessEvents

☑ com.cambridgesemantics.anzo.auditlog.activityEvents

com.cambridgesemantics.anzo.auditlog.auditQueryCutoffTime
None

□ com.cambridgesemantics.anzo.auditlog.auditSystemtableQueries

□ com.cambridgesemantics.anzo.auditlog.gzipRdf

□ com.cambridgesemantics.anzo.auditlog.includeCallStack

☑ com.cambridgesemantics.anzo.auditlog.includeNonSysadmin

☑ com.cambridgesemantics.anzo.auditlog.includeSysadmin

□ com.cambridgesemantics.anzo.auditlog.includeTimingStack

□ com.cambridgesemantics.anzo.auditlog.limitAge

□ com.cambridgesemantics.anzo.auditlog.logFullTransaction

com.cambridgesemantics.anzo.auditlog.maxAge
None

☑ com.cambridgesemantics.anzo.auditlog.queryEvents

5. Select the **com.cambridgesemantics.anzo.auditlog.limitAge** checkbox to enable the age limit feature.
6. Edit the **com.cambridgesemantics.anzo.auditlog.maxAge** property to specify the maximum number of days to log in each data set. When the current audit log reaches that age, Anzo starts writing to a new data set.
7. Restart Anzo to apply the configuration changes.

Related Topics

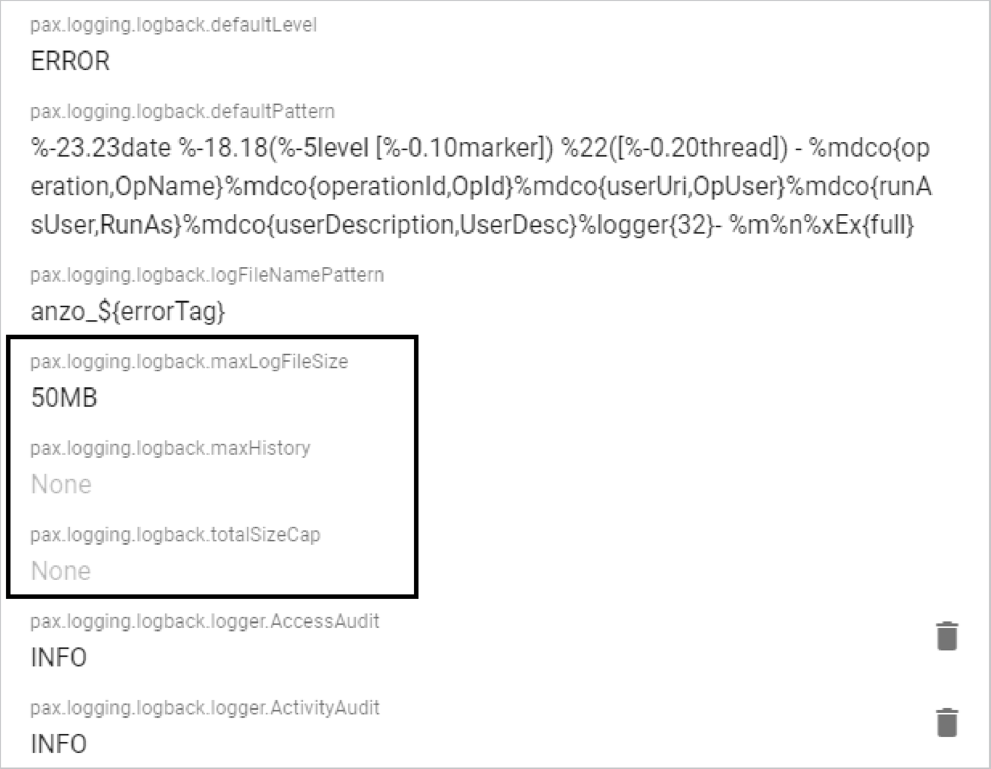
System Query Audit

Enabling and Viewing Audit Logs

Limiting the Size and Number of anzo_full Logs

Follow the instructions below if you want to configure the Pax Logging SLF4j Listener Service to limit the size and number of anzo_full logs that are retained on disk. You can also set a limit on the total size of all anzo_* logs.

1. In the Administration application, expand the **Servers** menu and click **Advanced Configuration**. Click **I understand and accept the risk**.
2. Search for the **Pax Logging SLF4j Listener** bundle and view its details.
3. Click the **Services** tab and expand the **SLF4j Log Listener** service.
4. Find the **maxLogFileSize**, **maxHistory**, and **totalSizeCap** properties (shown below).



```
pax.logging.logback.defaultLevel
ERROR

pax.logging.logback.defaultPattern
%-23.23date %-18.18(%-5level [%-0.10marker]) %22([%-0.20thread]) - %mdco{op
eration,OpName}%mdco{operationId,OpId}%mdco{userUri,OpUser}%mdco{runA
sUser,RunAs}%mdco{userDescription,UserDesc}%logger{32}- %m%n%xEx{full}

pax.logging.logback.logFileNamePattern
anzo_${errorTag}

pax.logging.logback.maxLogFileSize
50MB

pax.logging.logback.maxHistory
None

pax.logging.logback.totalSizeCap
None

pax.logging.logback.logger.AccessAudit
INFO

pax.logging.logback.logger.ActivityAudit
INFO
```

5. Edit any of the following properties to set them to the desired values:
   - **pax.logging.logback.maxLogFileSize**: This property sets the maximum file size for anzo_ full.log. When the maximum size is reached, Anzo stops writing to that file and creates a new one.
   - **pax.logging.logback.maxHistory**: This property specifies the maximum number of historical anzo_full.log files to keep. When this limit is reached, Anzo deletes the oldest file.

- **pax.logging.logback.totalSizeCap**: This property sets the total size limit for all anzo_* log files combined.

6. After editing a property, click the checkmark icon (✓) for that property to save the change.
7. Restart Anzo to apply the configuration changes.

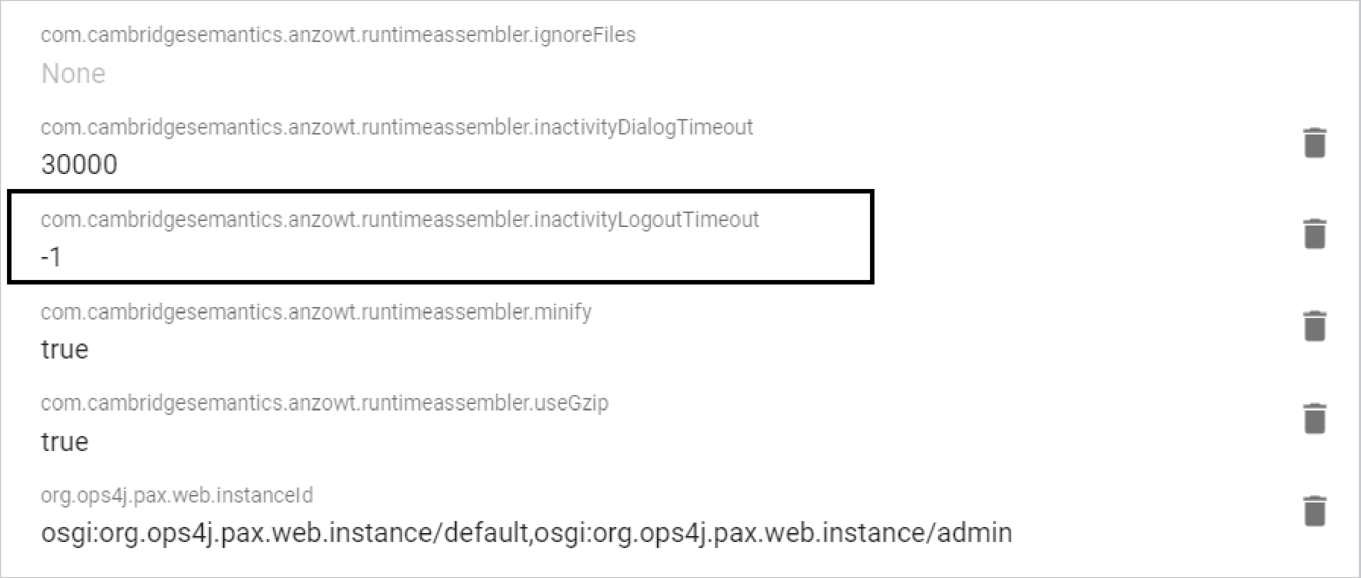
Related Topics

Managing Anzo Logging

Configuring a User Inactivity Timeout

By default, the user inactivity timeout setting in the Anzo Java Script Runtime Assembler service is set to **unlimited**, meaning Anzo will not automatically log out users who have a session open but remain inactive. If you want to configure Anzo to log users out if they are inactive for a period of time, follow the instructions below.

1. In the Administration application, expand the **Servers** menu and click **Advanced Configuration**. Click **I understand and accept the risk**.
2. Search for the **Anzo Java Script Runtime Assembler** bundle and view its details.
3. Click the **Services** tab and expand the **Anzo Java Script Runtime Assembler** service.
4. Edit the **com.cambridgesemantics.anzowt.runtimeassembler.inactivityLogoutTimeout** property (shown in the image below) to specify the number of **milliseconds** that a user can remain inactive before being logged out.



com.cambridgesemantics.anzowt.runtimeassembler.ignoreFiles
None

com.cambridgesemantics.anzowt.runtimeassembler.inactivityDialogTimeout
30000

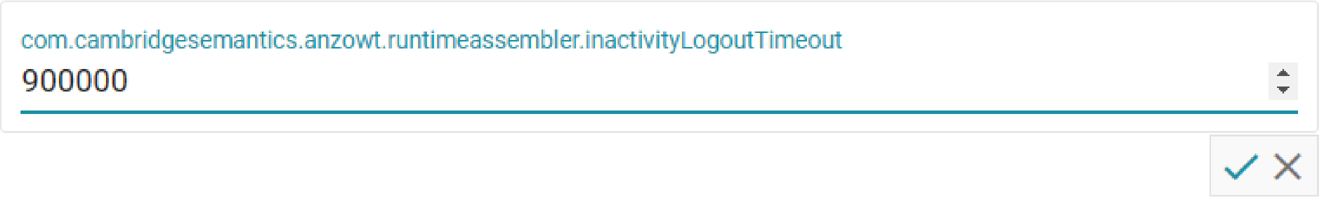
com.cambridgesemantics.anzowt.runtimeassembler.inactivityLogoutTimeout
-1

com.cambridgesemantics.anzowt.runtimeassembler.minify
true

com.cambridgesemantics.anzowt.runtimeassembler.useGzip
true

org.ops4j.pax.web.instanceId
osgi:org.ops4j.pax.web.instance/default,osgi:org.ops4j.pax.web.instance/admin

For example, setting the value to **900000** milliseconds means that a user who is inactive for more than 15 minutes is automatically logged out.



com.cambridgesemantics.anzowt.runtimeassembler.inactivityLogoutTimeout
900000

5. After specifying the value, click the checkmark icon (✓) for that property to save the change.
6. Restart Anzo to apply the configuration change.

> **Note**
> By default, Anzo is not configured to log an event when the user inactivity value is changed. If you would like this event to be noted in the Audit log when the setting is changed, see Enabling and Viewing Audit Logs for instructions.

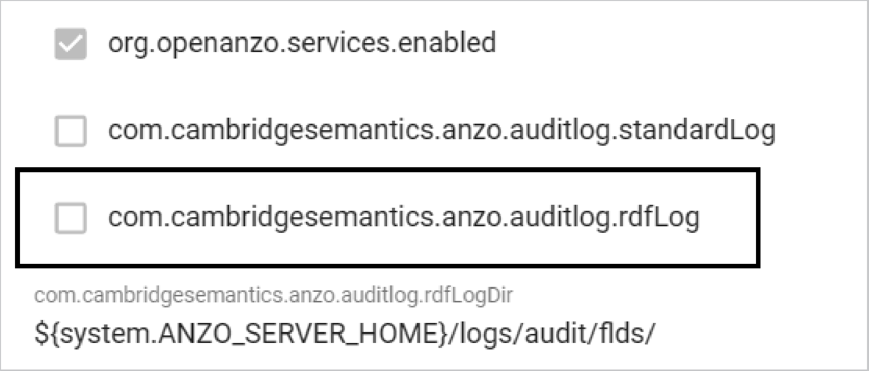
Related Topics

Enabling and Viewing Audit Logs

Reporting on Binary Store Access Events

By default, binary store access events are not captured in the Audit log. You can configure the audit logging framework to capture information about binary store requests, however. Data such as the time of the request, the user who made the request, and the document that was accessed will be captured. Follow the instructions below to configure the log to report on binary store events.

1. In the Administration application, expand the **Servers** menu and click **Advanced Configuration**. Click **I understand and accept the risk**.
2. Search for the **Anzo Audit Logging Framework** bundle and view its details.
3. Click the **Services** tab and expand **com.cambridgesemantics.anzo.AuditLog**.
4. Locate the **com.cambridgesemantics.anzo.auditlog.rdfLog** property (shown in the image below).



☑ org.openanzo.services.enabled

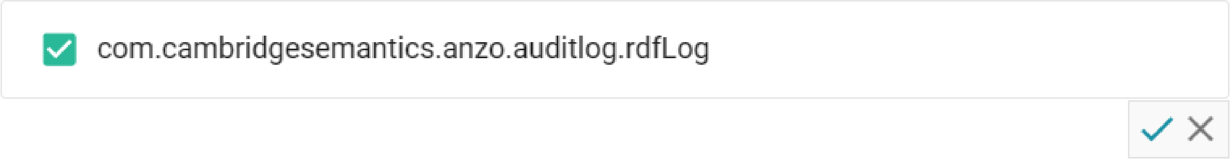
□ com.cambridgesemantics.anzo.auditlog.standardLog

□ com.cambridgesemantics.anzo.auditlog.rdfLog

com.cambridgesemantics.anzo.auditlog.rdfLogDir

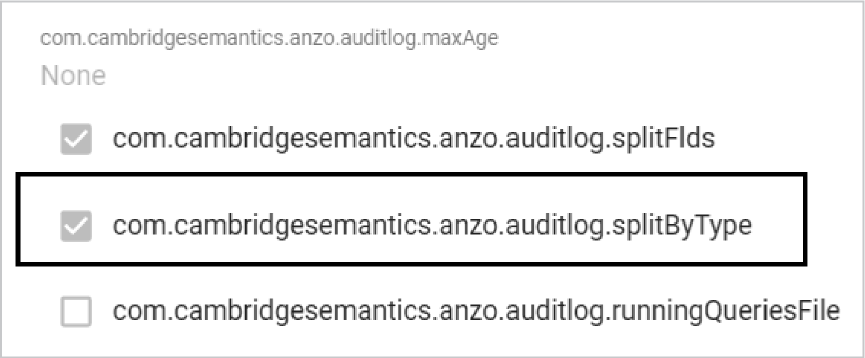
${system.ANZO_SERVER_HOME}/logs/audit/flds/

5. Click the property to make it editable, and then select the checkbox to enable it.



☑ com.cambridgesemantics.anzo.auditlog.rdfLog

6. Click the checkmark icon (✓) for that property to save the change.
7. Scroll down and make sure that the **com.cambridgesemantics.anzo.auditlog.splitByType** property is selected/enabled (it is enabled by default).



com.cambridgesemantics.anzo.auditlog.maxAge
None

☑ com.cambridgesemantics.anzo.auditlog.splitFlds

☑ com.cambridgesemantics.anzo.auditlog.splitByType

□ com.cambridgesemantics.anzo.auditlog.runningQueriesFile

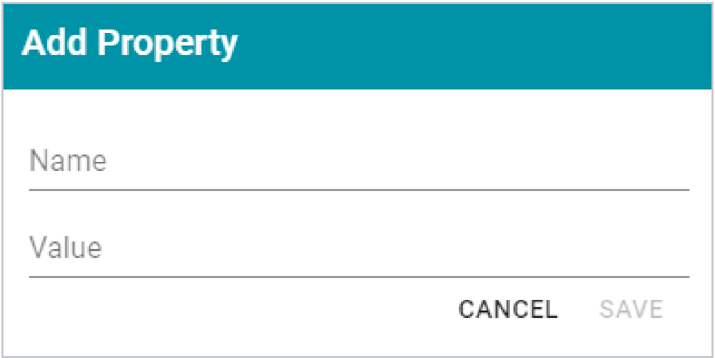
8. Restart Anzo to apply the configuration change.

New binary store access audit events will be added to the logs in the subdirectories under `<install_path>/Server/logs/audit/audit-flds`.

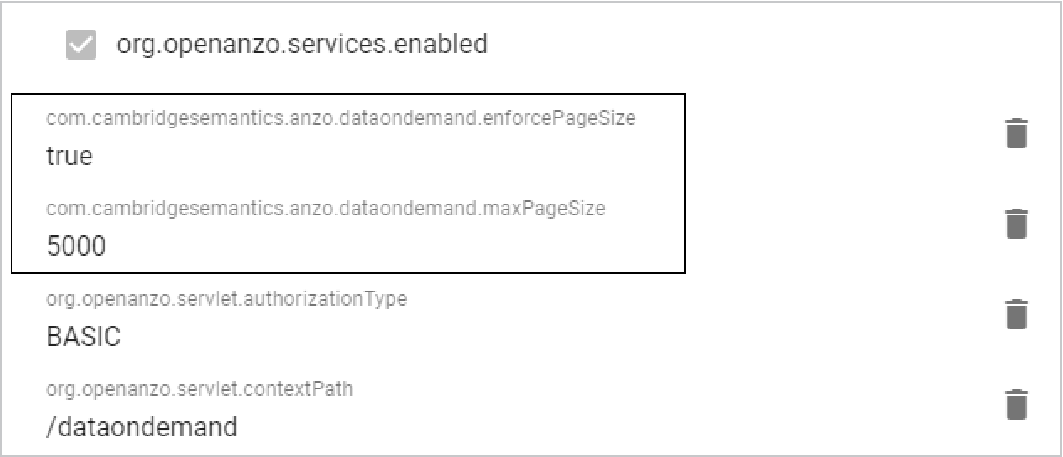
Configuring the Max Page Size for OData Feeds

When a user sends a request to an Anzo Data on Demand endpoint, they do not necessarily know the total number of results that will be returned. In some cases, the result set can be hundreds of millions of values, and the request times out before the results can be returned. You can configure the Data on Demand service to specify a maximum limit on the number of results that can be returned for a single OData feed request. If a user sends a request and the result set is larger than the maximum value, Anzo will limit the results to the configured maximum value. Follow the instructions below to configure the Data on Demand service to enforce a maximum page size.

1. In the Administration application, expand the **Servers** menu and click **Advanced Configuration**. Click **I understand and accept the risk**.
2. Search for the **Anzo DataOnDemand** bundle and view its details.
3. Click the **Services** tab and expand **DataOnDemandServiceActivator**.
4. Click **Add Property** next to the service name. Anzo opens the Add Property dialog box.



5. In the **Name** field, specify **com.cambridgesemantics.anzo.dataondemand.enforcePageSize**, and set the **Value** to **true**. Then click **Save**.
6. Click **Add Property** again. In the **Name** field, specify **com.cambridgesemantics.anzo.dataondemand.maxPageSize**, and set the **Value** to the maximum number of results that to return per request. Then click **Save**. The two settings are displayed on the Services screen. For example:



☑ org.openanzo.services.enabled

com.cambridgesemantics.anzo.dataondemand.enforcePageSize
true

com.cambridgesemantics.anzo.dataondemand.maxPageSize
5000

org.openanzo.servlet.authorizationType
BASIC

org.openanzo.servlet.contextPath
/dataondemand

7. Restart Anzo to apply the configuration changes.

Scanning the Whole CSV File on Import

To help improve accuracy of data type assignment when importing CSV files, you have the option to configure the system so that any time a CSV file is imported, Anzo scans the entire file before inferring the data types for each column. Follow the instructions below if you want to configure the system to scan entire CSV files.

> **Important**
> This change affects all CSV file imports. Users cannot opt-out of a complete scan at import time. This configuration is not related to the **Use Extended Sample** setting in file import options. Choosing to scan entire files will significantly increase the time it takes to import files. However, scanning the complete file is the best way to ensure that data type assignments are accurate.

1. In the Administration application, expand the **Servers** menu and click **Advanced Configuration**. Click **I understand and accept the risk**.
2. Search for the **Anzo Utilityservices VFS** bundle and view its details.
3. Click the **Services** tab and expand **UtilityServices VFS Activator**.
4. Find the **com.cambridgesemantics.anzo.utilityservices.vfs.isSampleEntireFile** property, and select the checkbox to enable the option.

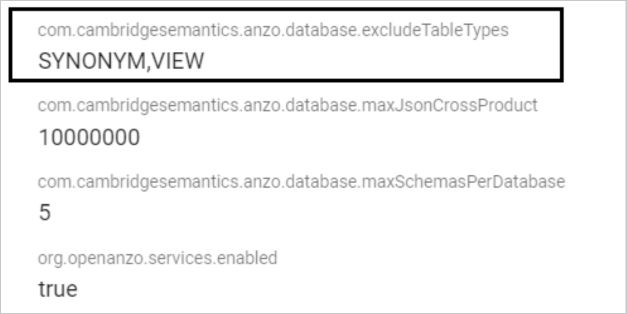
   > **Note**
   > When **SampleEntireFile** is enabled, the values in the **maxSampleSize** and **sampleSize** properties are ignored and Anzo always scans entire CSV files on import.

5. Restart Anzo to apply the configuration changes.

Including Views as Schemas for Database Data Sources

By default, when you create a Database Data Source and import a predefined Schema, Views are excluded from the list of Schemas that are available to import. However, you can configure the Anzo Database DataSource Provider Service to include Views as Schemas. Follow the steps below to remove Views from the list of table types that are excluded from import.

1. In the Administration application, expand the **Servers** menu and click **Advanced Configuration**. Click **I understand and accept the risk**.
2. Search for the **Anzo Database DataSource Provider** bundle and view its details.
3. Click the **Services** tab and expand **com.cambridgesemantics.anzo.database.IDbConnectionService**.
4. Locate the **com.cambridgesemantics.anzo.database.excludeTableTypes** property (shown in the image below).



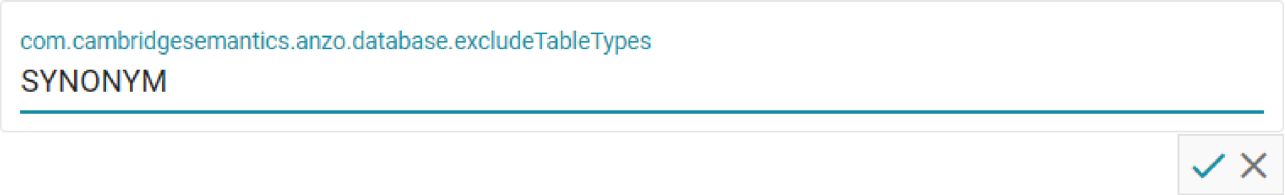
com.cambridgesemantics.anzo.database.excludeTableTypes
SYNONYM,VIEW

com.cambridgesemantics.anzo.database.maxJsonCrossProduct
10000000

com.cambridgesemantics.anzo.database.maxSchemasPerDatabase
5

org.openanzo.services.enabled
true

5. Click the property to make it editable, and then delete the word **VIEW**.



com.cambridgesemantics.anzo.database.excludeTableTypes
SYNONYM

6. Click the checkmark icon (✓) for that property to save the change.
7. Restart Anzo to apply the configuration change.

The service is now configured to display views in the Import Schemas dialog box.

Limiting the Number of Anzo Unstructured Status Journals

To limit the disk space used by Anzo Unstructured pipelines, you have the option to configure the Anzo Unstructured Distributed service to limit the number of status journals that are preserved on disk. When the specified limit is reached and a pipeline generates a new journal, the oldest journal is deleted.

> **Note**
> Journals are removed based on their timestamps alone. The pipeline they are associated with is not a factor in determining the journals to delete.

Follow the instructions below to configure the Unstructured Distributed service to limit the number of status journals on disk.

1. In the Administration application, expand the **Servers** menu and click **Advanced Configuration**. Click **I understand and accept the risk**.
2. Search for the **Anzo Unstructured Distributed** bundle and view its details.
3. Click the **Services** tab and expand **Anzo Unstructured Distributed**.
4. Edit the **com.cambridgesemantics.anzo.unstructured.distributed.defaultNumStatusJournalGlobalLimit** property to specify the maximum number of status journals to keep on disk. The default value is **-1**, which is unlimited.
5. After changing the value, click the checkmark icon (✓) for that property to save the change.
6. Restart Anzo to apply the configuration change.

Connection Administration

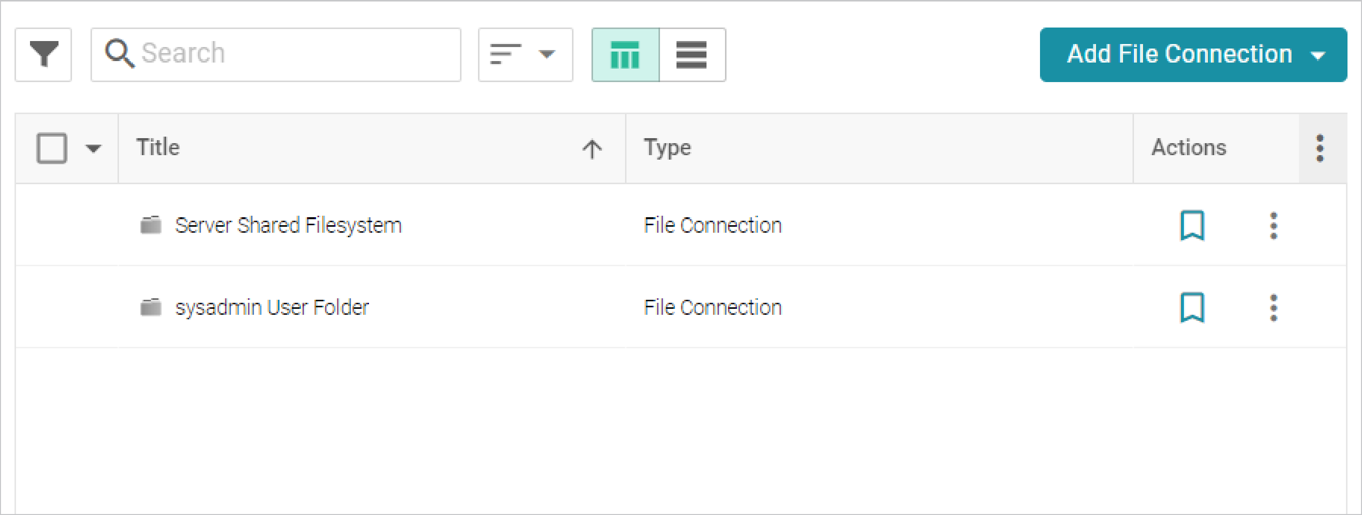
The topics in this section provide information about managing connections to the Anzo server.

Connecting to a File Store

This topic provides instructions for connecting to an additional shared file system that Anzo applications can read from and write to during the onboarding processes. At least one file store needs to be shared between Anzo, AnzoGraph, and any Anzo Unstructured, Elasticsearch, or Spark servers. In almost all cases, organizations create an NFS to mount to all of the servers in the Anzo environment. Mounted file systems typically offer the best performance for reading and writing files. For more information, see Creating the Shared File System in the Deployment Guide.

Anzo supports reading from and writing to local or mounted file systems (such as NFS), Hadoop Distributed File Systems (HDFS), File Transfer Protocol (FTP or FTPS) systems, Google Cloud Platform (GCP) storage, and Amazon Simple Cloud Storage Service (S3).

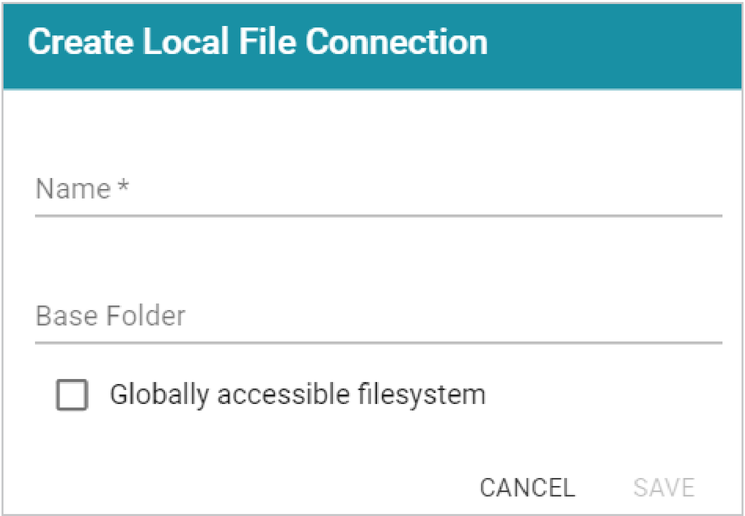
1. In the Administration application, expand the **Connections** menu and click **File Store**. Anzo displays the File Store screen, which lists existing file store connections. For example:



Title	Type	Actions
Server Shared Filesystem	File Connection	
sysadmin User Folder	File Connection	

2. Click the **Add File Connection** button and select the type of file connection that you want to create. For the local disk or mounted NFS, choose **Local File Connection**. Anzo displays the create connection screen for the type of connection you chose.
3. On the connection screen, provide the file system details. The settings that display depend on the type of file connection that you chose. The list below describes the settings for each file connection type.

Local File Connection



Create Local File Connection

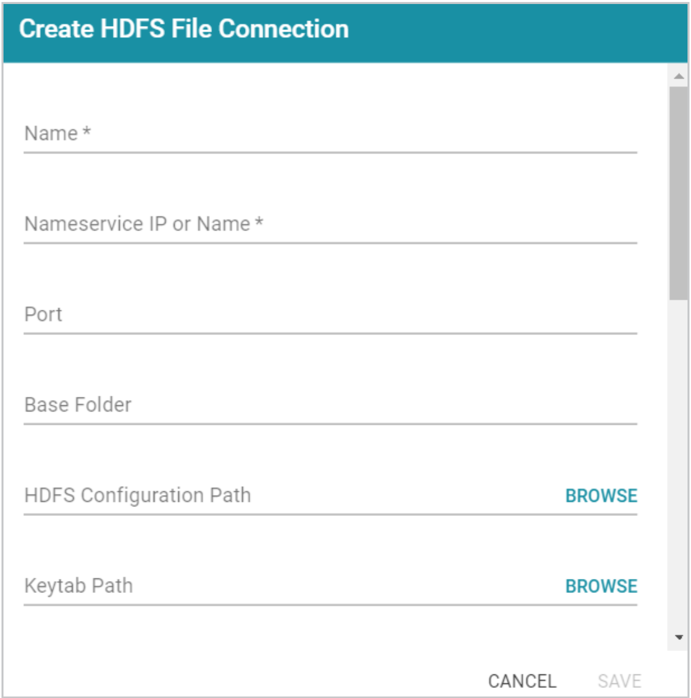
Name *

Base Folder

□ Globally accessible filesystem

- **Name**: The name to use to describe this file connection within Anzo.
- **Base Folder**: The base or root folder on the file system where you want Anzo to either read or write files. Each time Anzo generates new files it creates a new subdirectory under this base location.
- **Globally accessible filesystem**: Select this option if this file store is accessible by all of the servers in an AnzoGraph cluster. If only the AnzoGraph leader server can access this system, leave this option blank.

HDFS File Connection


Create HDFS File Connection

Name *

Nameservice IP or Name *

Port

Base Folder

HDFS Configuration Path

Keytab Path

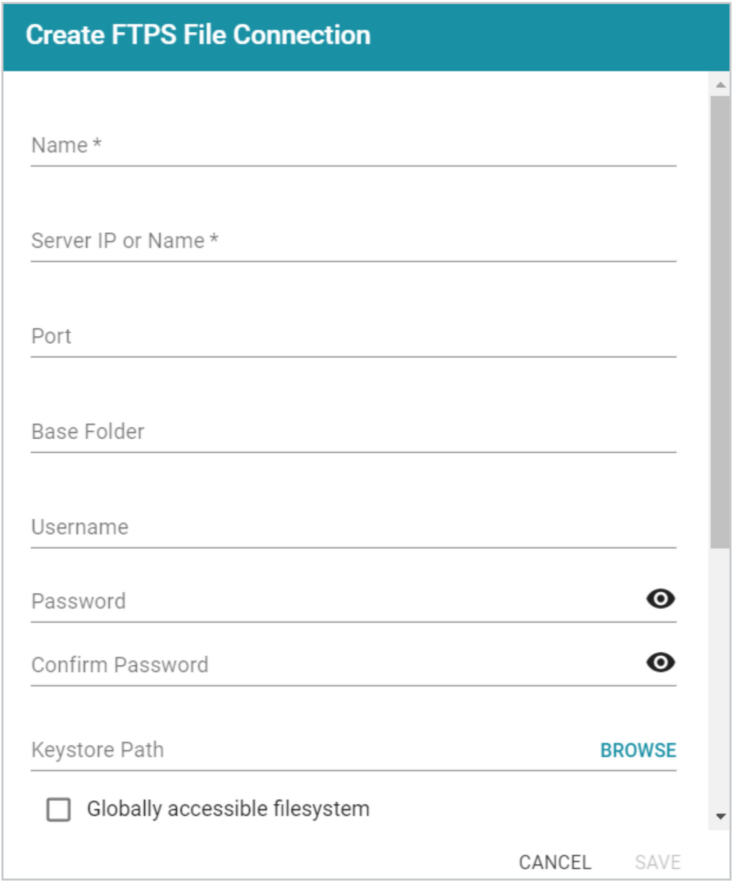
- **Name**: The name to use to describe this file connection within Anzo.
- **Nameservice IP or Name**: The IP address or host name for the storage system.
- **Port**: The RPC port to access the server on. The default RPC port is 8020.
- **Base Folder**: The base or root folder on the file system where you want Anzo to either read or write files. Each time Anzo generates new files it creates a new subdirectory under this base location.
- **HDFS Configuration Path**: Enter the full path to the configuration files.
- **Keytab Path**: The full path to the keytab file.
- **Username**: The user name for the account used to access the server.
- **Password** and **Confirm Password**: The password for the account used to access the server.
- **Nameservice Rest IP or Name**: The HTTP REST IP address or host name. Typically this value is the same as the Nameservice IP or Name.
- **Nameservice Rest Port**: The HTTP port. AnzoGraph uses this port to access HDFS and load the FLDS. The default HTTP port for the namenode is 9870.
- **Nameservice Rest Protocol**: The protocol to use for requests. Specify one of the following values:

- **hdfs**: Specify **hdfs** for non-secure HTTP protocol.
- **shdfs**: Specify **shdfs** for secure HTTPS protocol.
- **khdfs**: Specify **khdfs** for non-secure HTTP protocol with Kerberos authentication.
- **kshdfs**: Specify **kshdfs** for secure HTTPS protocol with Kerberos authentication.

> **Important**
> If you use Kerberos Authentication with HDFS, you must also configure your AnzoGraph cluster to authenticate with Kerberos. For instructions, see Configuring AnzoGraph for Kerberos Authentication.

- **Globally accessible filesystem**: Select this option if this file store is accessible by all of the servers in an AnzoGraph cluster. If only the AnzoGraph leader server can access this system, leave this option blank.

FTP or FTPS File Connection



Create FTPS File Connection

Name *

Server IP or Name *

Port

Base Folder

Username

Password

Confirm Password

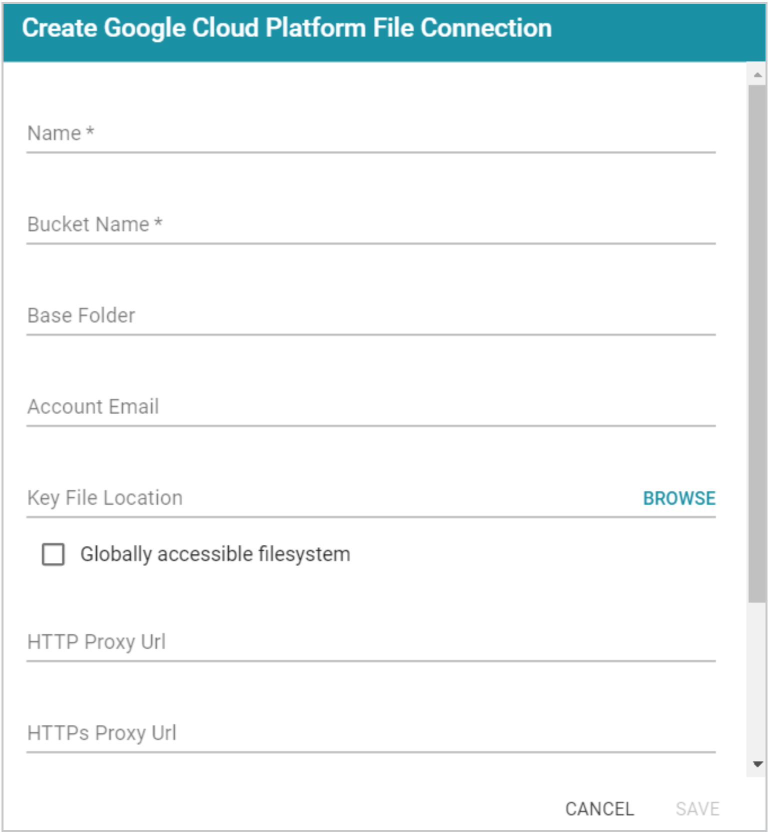
Keystore Path BROWSE

□ Globally accessible filesystem

CANCEL SAVE

- **Name**: The name to use to describe this file connection within Anzo.
- **Server IP or Name**: The IP address or host name for the storage system.
- **Port**: The port to access the server on.
- **Base Folder**: The base or root folder on the file system where you want Anzo to either read or write files. Each time Anzo generates new files it creates a new subdirectory under this base location.
- **Username**: The user name for the account used to access the server.
- **Password** and **Confirm Password**: The password for the account used to access the server.
- **Keystore Path**: For FTPS connections, the full path to the keystore file.
- **Globally accessible filesystem**: Select this option if this file store is accessible by all of the servers in an AnzoGraph cluster. If only the AnzoGraph leader server can access this system, leave this option blank.

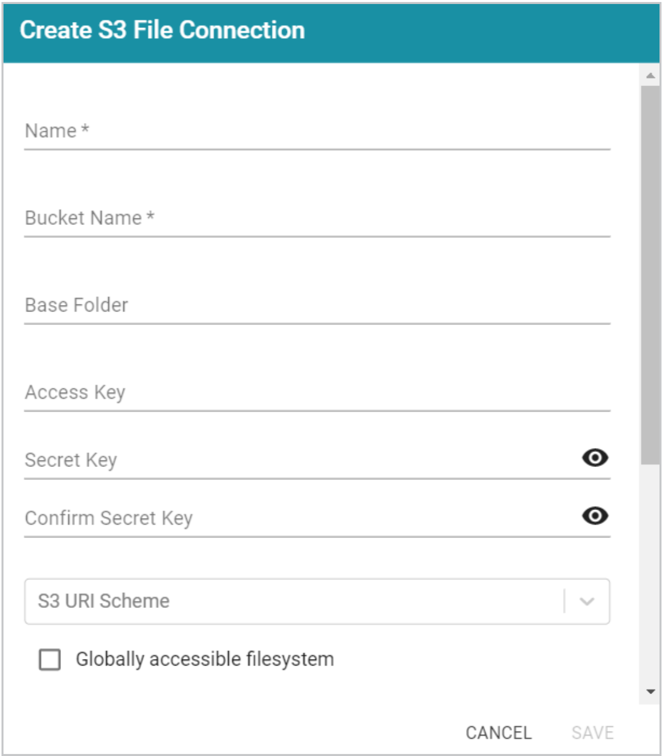
Google Cloud Platform File Connection



- **Name**: The name to use to describe this file connection within Anzo.
- **Bucket Name**: The name of the bucket to store files in.
- **Base Folder**: The base or root folder on the file system where you want Anzo to either read or write files. Each time Anzo generates new files it creates a new subdirectory under this base location.
- **Account Email**: The email address for the account used to access the storage.
- **Key File Location**: The full path to the keystore password file.
- **Globally accessible filesystem**: Select this option if this file store is accessible by all of the servers in an AnzoGraph cluster. If only the AnzoGraph leader server can access this system, leave this option blank.

S3 File Connection

> **Important**
> When using Amazon S3 for file storage, do not use client-side encryption, where data is encrypted before it is sent to Amazon S3. Anzo cannot read files on S3 if the object store uses client-side encryption.



Create S3 File Connection

Name *

Bucket Name *

Base Folder

Access Key

Secret Key

Confirm Secret Key

S3 URI Scheme

□ Globally accessible filesystem

CANCEL SAVE

- **Name**: The name to use to describe this file connection within Anzo.
- **Bucket Name**: The name of the bucket to store files in.
- **Base Folder**: The base or root folder on the file system where you want Anzo to either read or write files. Each time Anzo generates new files it creates a new subdirectory under this base location.
- **Access Key**: The Access Key ID to use for accessing the S3 location.
- **Secret Key** and **Confirm Secret Key**: The Secret Key ID for the Access Key.
- **S3 URI Scheme**: Specifies whether the URI scheme is S3, S3 Native, or S3A.
- **Globally accessible filesystem**: **Required**. Enable this option for S3 file stores.

4. Click **Save** to save the configuration. The file store connection that you specified becomes available as a choice when you create graph data stores or select source files to onboard.

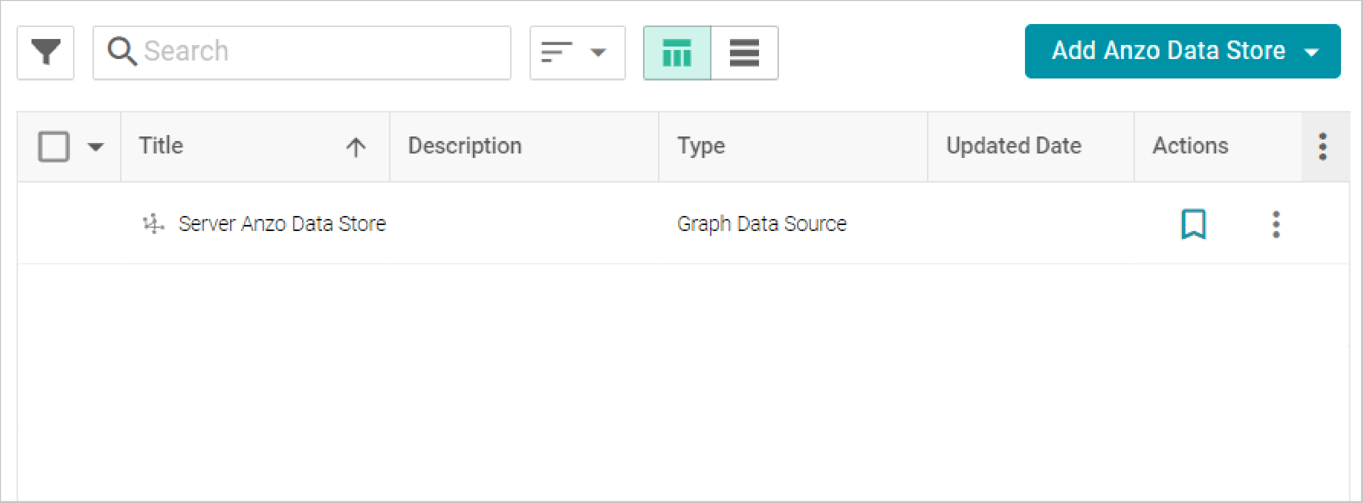
Related Topics

Creating an Anzo Data Store

Creating an Anzo Data Store

This topic provides instructions for creating an Anzo Data Store. Creating a data store means that you designate a directory on a shared file store where file-based linked data sets and other files can be created and shared. If you run Spark or unstructured Pipelines, a data store is required. In addition, a data store is required if you use the Direct Data Load automated workflow for structured data and configure the workflow to automatically export the data to a dataset. You can create one data store and configure all pipelines and workflows to write to that store or you can create multiple data stores to use for different datasets.

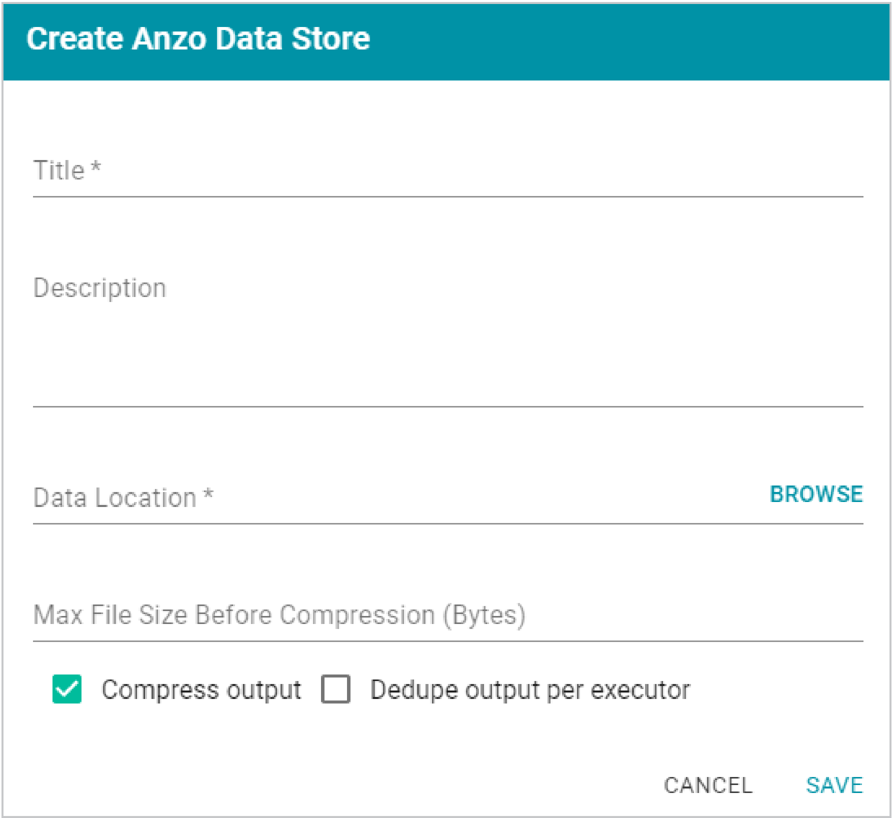
1. In the Administration application, expand the **Connections** menu and click **Anzo Data Store**. Anzo displays the Anzo Data Store screen, which lists any existing data stores. For example:



Title	Description	Type	Updated Date	Actions
Server Anzo Data Store		Graph Data Source		

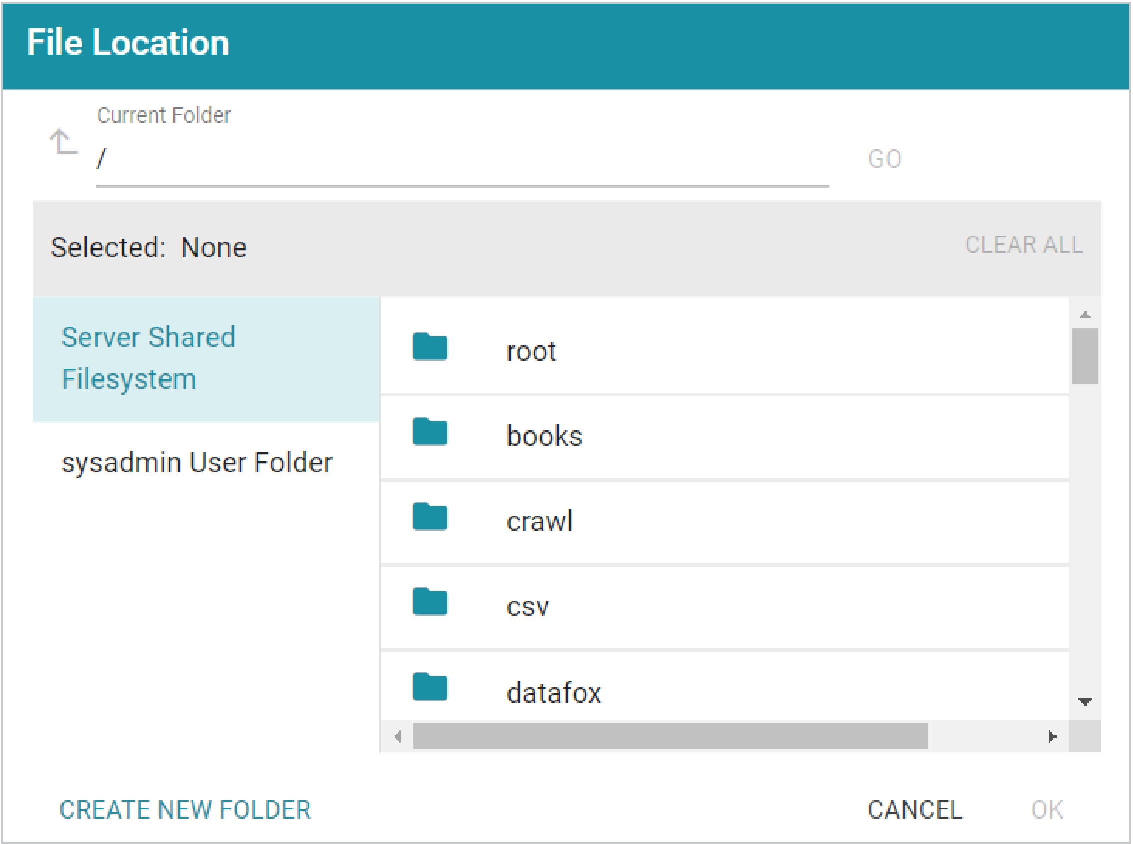
> **Important**
> The **Server Anzo Data Store** is a default data store that points to the local Anzo file system. This store exists so that first-time users can quickly test the onboarding process. It is not meant to be used in production. Do not change the Data Location to a shared file store; reconfiguring this Data Store can cause unexpected consequences when upgrading or migrating the system. It is safe to delete this store so that it is not presented as an option when users configure ingestion pipelines.

2. On the Anzo Data Store screen, click the **Add Anzo Data Store** button and select **Add Anzo Data Store**. Anzo opens the Create Anzo Data Store screen.



3. Type a **Title** and optional **Description** for the data store.

4. Click in the **Data Location** field. Anzo opens the File Location dialog box.



5. On the left side of the screen, select the File Store on which to create this data store. On the right side of the screen, navigate to the directory that you want to designate as the data location. Select a directory, and then click **OK**. Or click **Create New Folder** to create a new directory. Each time a pipeline is run for this data store, a new subdirectory is created under the specified data location.

> **Note**
> The Data Location needs to be a directory on the file store that is shared between Anzo, AnzoGraph, and any Anzo Unstructured, Elasticsearch, or Spark servers. If you want Anzo to generate files for this data store in one location and then load the files into AnzoGraph from another location, specify the file generation location in this field, and then specify the AnzoGraph load location in the **Alternate Data Location** field that is displayed on the Details screen after you save the data store.

6. If necessary, you can modify the maximum limit for the size of the files that are created by pipelines that write to this data store by specifying the size (in bytes) in the **Max File Size Before Compression (Bytes)** field. The value applies to files before they are compressed. The Spark ETL engine partitions files on output, and the default maximum file size is 100 MB (uncompressed). The Sparkler ETL engine

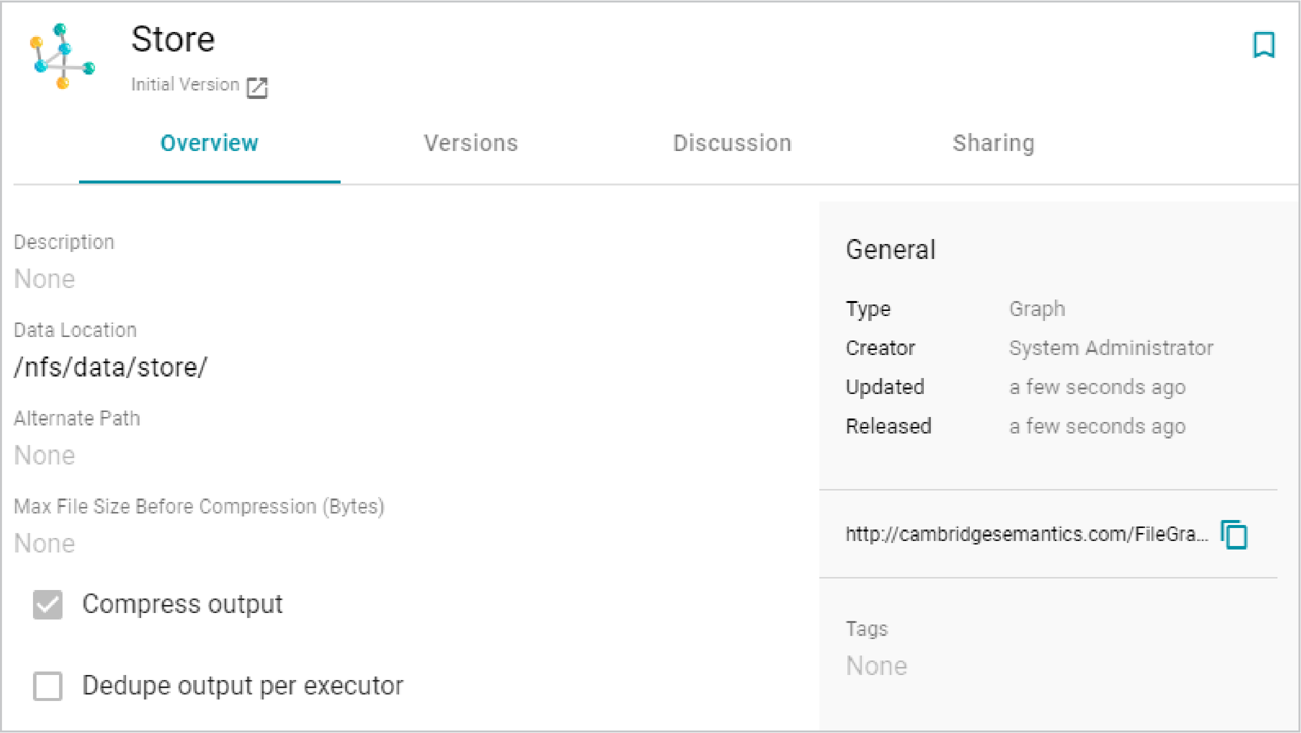
partitions files on input, and the default maximum file size is 128 MB (uncompressed). Since Sparkler files are partitioned on input, the resulting output FLDS files can be significantly larger than 128 MB since the source is converted to Turtle (TTL) format after it is partitioned.

> **Note**
> Cambridge Semantics recommends that you do not set this value unless instructed to do so by Cambridge Semantics Support.

7. Specify whether to compress the generated load files. By default, the **Compress output** checkbox is selected, indicating that Anzo generates .ttl.gz files when writing to this graph data source. If you clear the checkbox, Anzo generates uncompressed .ttl files. To preserve disk space and reduce read times when loading data into memory, Cambridge Semantics recommends that you accept the default configuration and compress load files.
8. The ETL engine does not remove duplicates by default when running structured pipelines. If the source contains a significant number of duplicate entities, you have two options for deduplicating the data:
    - **Deduplicate the data during the ETL process**: To deduplicate the data while running the jobs that will generate this graph source, select the **Dedupe output per executor** option. Enabling the dedupe option limits the number of duplicates to one duplicate per executor node. For example, if the Spark configuration has 10 executor nodes, the resulting data set can contain a maximum of 10 duplicate entities.

        > **Important**
        > Deduplication is based on primary keys and URI templates. If the source does not employ templating, do not enable the dedupe option. In addition, enabling this option substantially increases the time it takes to run the jobs for this data store.

    - **Deduplicate the data after loading it to AnzoGraph**: AnzoGraph deduplicates data during a "vacuum" process that runs automatically after data is loaded into memory. If you leave the **Dedupe output per executor** option disabled, duplicates will be removed by AnzoGraph.
9. Click **Save** to create the data store. Anzo saves the configuration and displays the details view. For example:



Store

Initial Version

Description
None

Data Location
/nfs/data/store/

Alternate Path
None

Max File Size Before Compression (Bytes)
None

☑ Compress output

☐ Dedupe output per executor

General

Type	Graph
Creator	System Administrator
Updated	a few seconds ago
Released	a few seconds ago

http://cambridgesemantics.com/FileGra...

Tags
None

You can click the Edit icon (✎) to modify any of the options. Click the check mark icon (✓) to save changes to an option, or click the X icon (✕) to clear the value for an option.

10. If you plan to load files into AnzoGraph from a location that is different than the **Data Location** that you specified, edit the **Alternate Data Location** field and select the location for AnzoGraph load files.

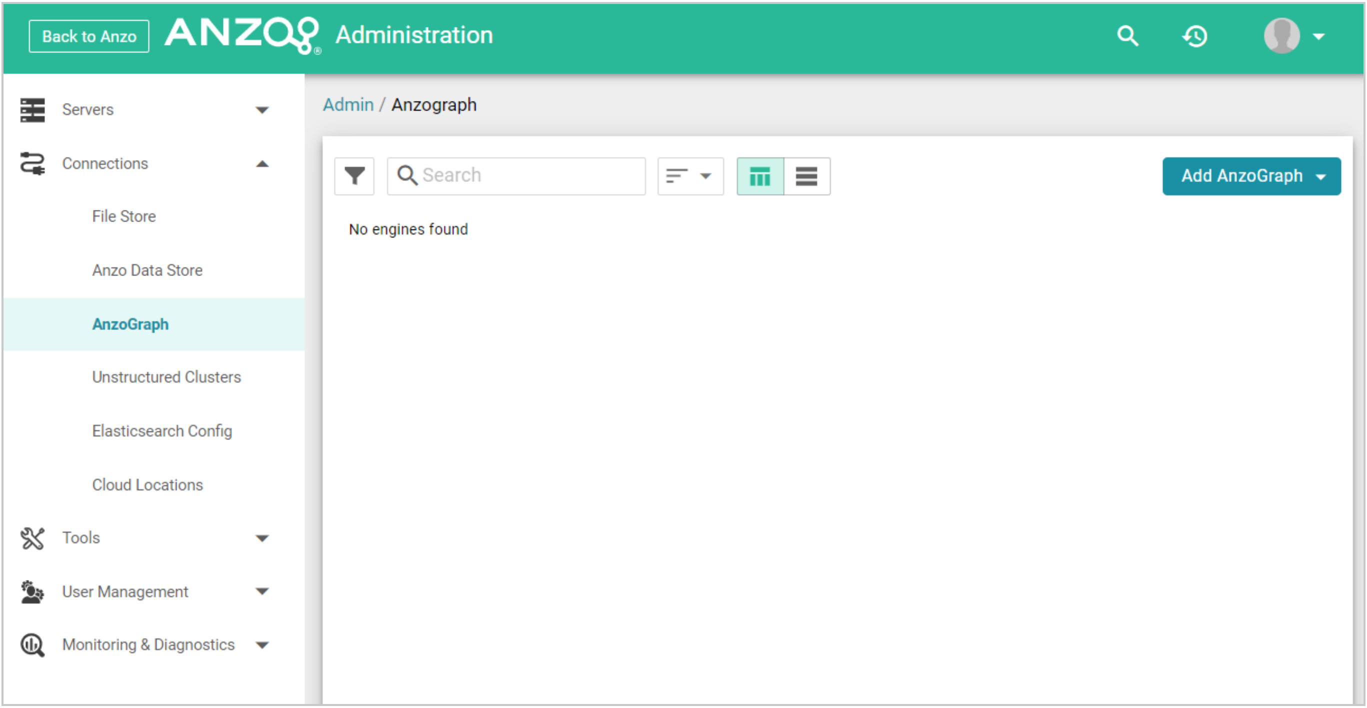
Once you have create the new data store, you can designate it as the default store so that it is automatically selected when users set up data onboarding workflows. See Configure the Default Anzo Data Store for instructions.

Connecting to AnzoGraph

This topic provides instructions for configuring the connection to AnzoGraph. For information about managing AnzoGraph servers, see AnzoGraph Server Administration.

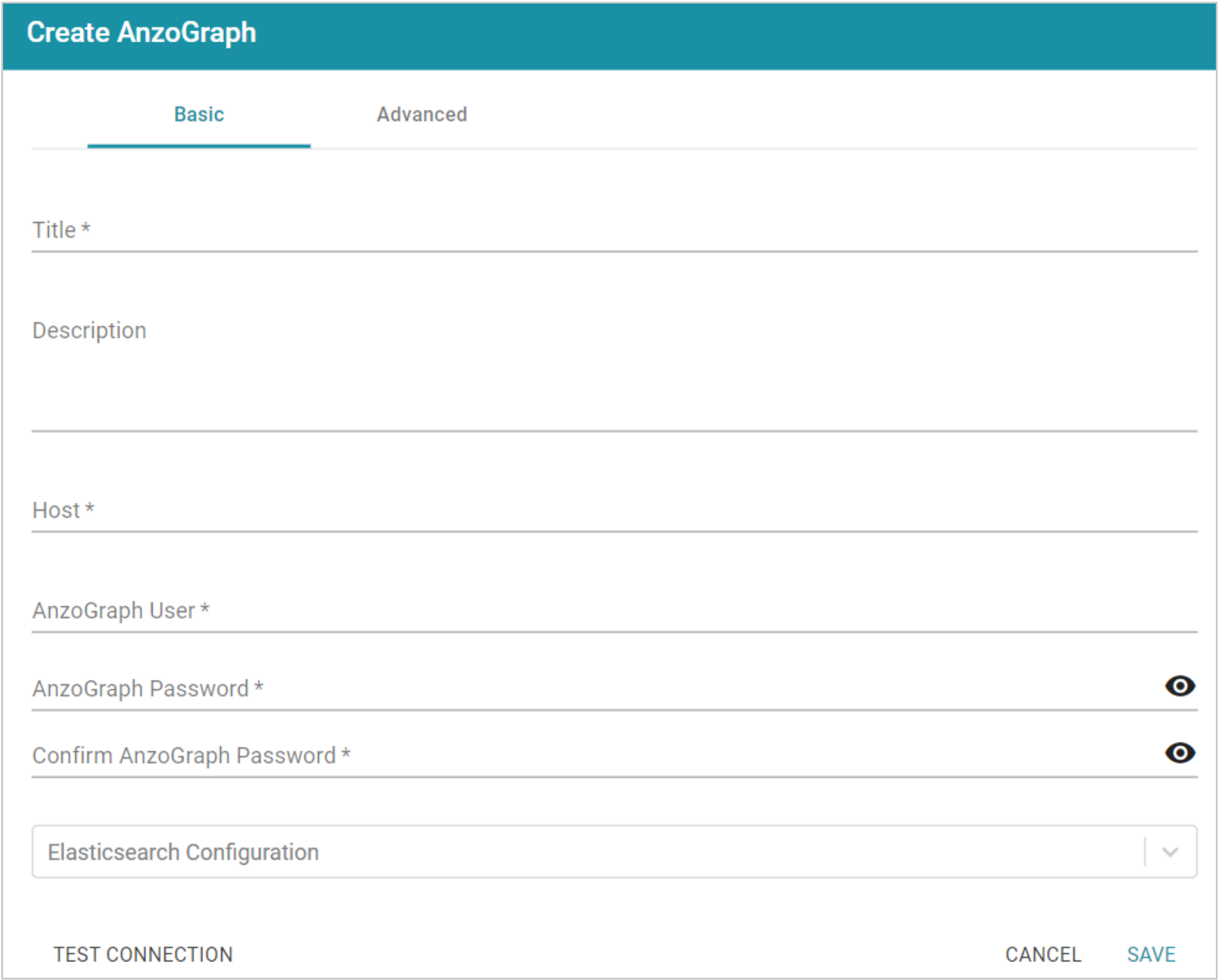
Important
Do not connect multiple Anzo instances to the same AnzoGraph instance. Since AnzoGraph is stateless and Anzo manages all of the data, connecting more than one Anzo instance to the same AnzoGraph instance causes severe data management conflicts that result in unexpected behavior. This type of configuration is not supported.

1. In the Administration application, expand the **Connections** menu and click **AnzoGraph**. Anzo opens the AnzoGraph connection overview screen, which lists any existing connections. For example:



No engines found

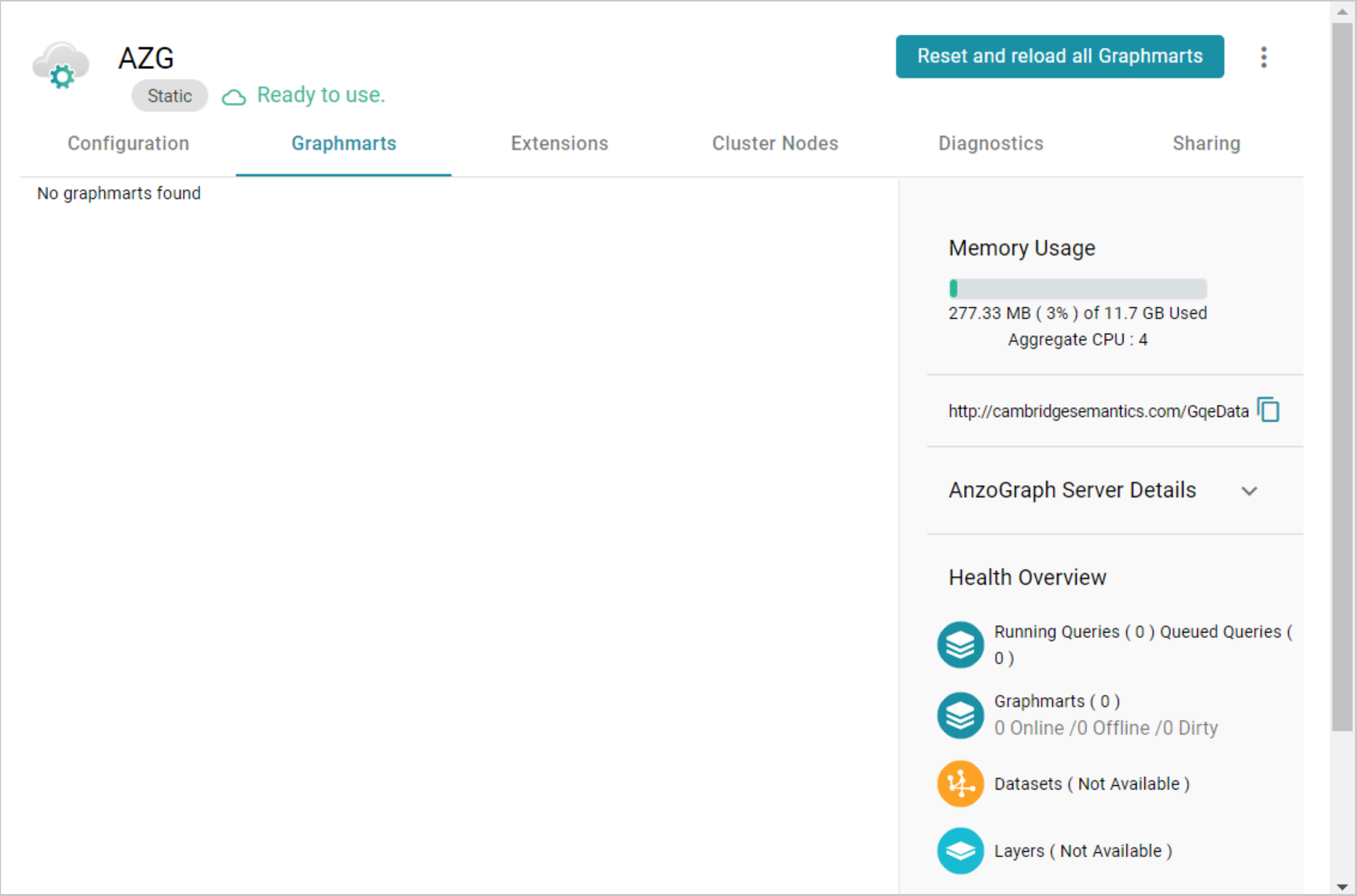
2. On the AnzoGraph screen, click **Add AnzoGraph** and select **Add AnzoGraph** from the drop-down list. Anzo displays the Create AnzoGraph dialog box.



3. On the Basic tab, type a name for the instance in the **Title** field.
4. In the optional **Description** field, type a description for the instance. If you leave this field blank, Anzo creates a description when you save the configuration.
5. In the **Host** field, type the AnzoGraph server host name or IP address. If you have a cluster, type the name or IP address of the leader server.
6. In the **AnzoGraph User** field, type the Admin username that was created when AnzoGraph was installed.
7. Type the password for the AnzoGraph user in the **AnzoGraph Password** and **Confirm Password** fields.
8. If this AnzoGraph instance will host data associated with Elasticsearch, click the **Elasticsearch Configuration** drop-down list and select the Elasticsearch instance to use with this AnzoGraph

connection. For information about configuring an Elasticsearch connection, see Connecting to Elasticsearch.

9. Click **Test Connection** to check if Anzo can connect to AnzoGraph. If the connection fails, make sure that AnzoGraph is running and that you typed the correct username and password.
10. **Optional**: Click the **Advanced** tab and configure any of the optional advanced settings. For details about the Advanced settings, see AnzoGraph Advanced Settings Reference.
11. Click **Save** to save the configuration. Anzo connects to AnzoGraph and opens the Graphmarts tab. For example:



No graphmarts found

Memory Usage

277.33 MB (3%) of 11.7 GB Used

Aggregate CPU : 4

http://cambridgesemantics.com/GqeData

Health Overview

Running Queries (0) Queued Queries (0)

Graphmarts (0)
0 Online /0 Offline /0 Dirty

Datasets (Not Available)

Layers (Not Available)

To change configuration details, click the **Configuration** tab and adjust values as needed. The right side of the screen shows connection status as well as memory usage details, overall data statistics, and Graphmart details.

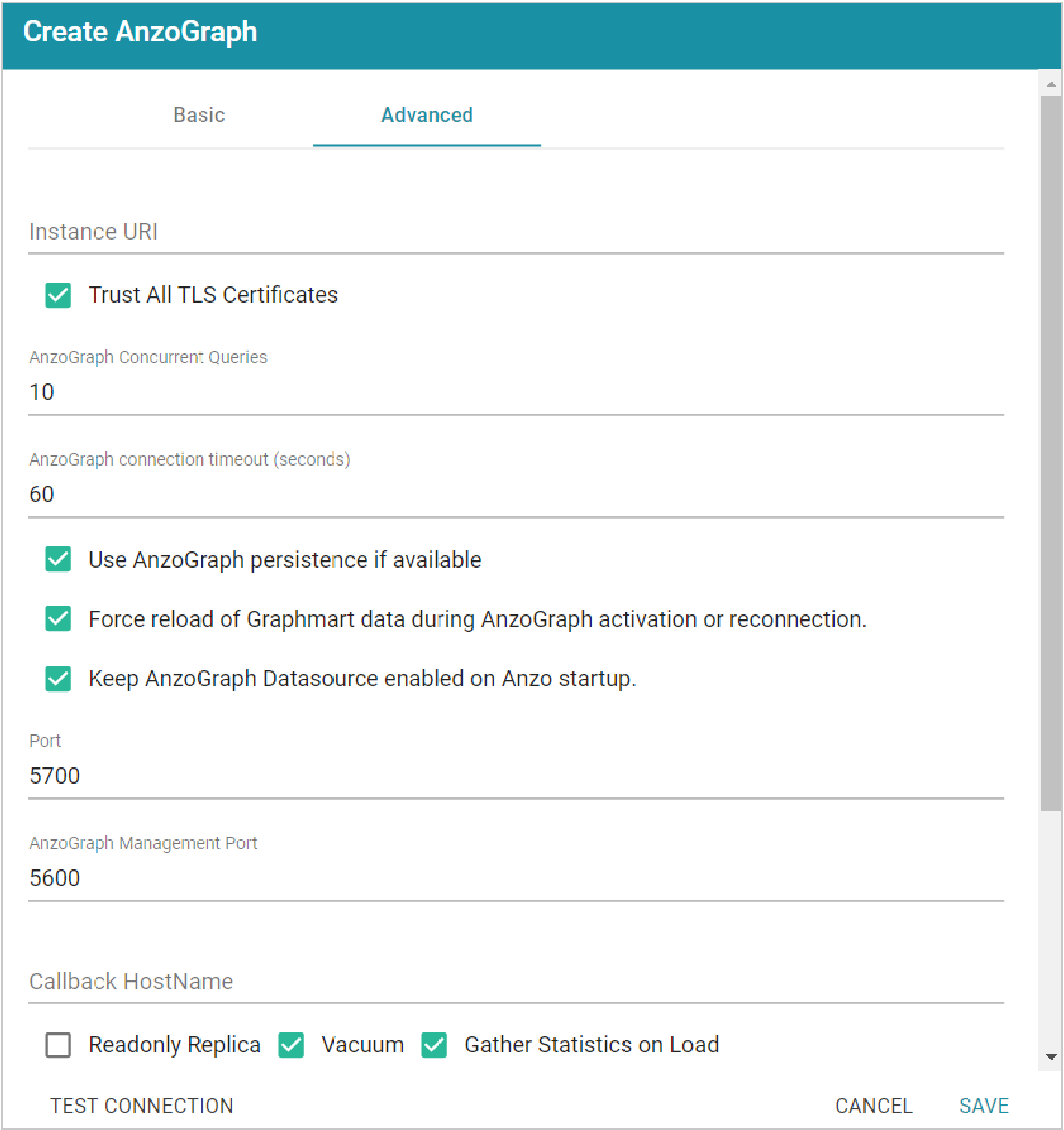
Related Topics

AnzoGraph Advanced Settings Reference

AnzoGraph Server Administration

AnzoGraph Advanced Settings Reference

This topic describes the Advanced AnzoGraph connection settings that are available on the Advanced tab when adding a static AnzoGraph connection or the Configuration tab when editing an existing connection.



Create AnzoGraph

Instance URI

☑ Trust All TLS Certificates

AnzoGraph Concurrent Queries: 10

AnzoGraph connection timeout (seconds): 60

☑ Use AnzoGraph persistence if available

☑ Force reload of Graphmart data during AnzoGraph activation or reconnection.

☑ Keep AnzoGraph Datasource enabled on Anzo startup.

Port: 5700

AnzoGraph Management Port: 5600

Callback HostName

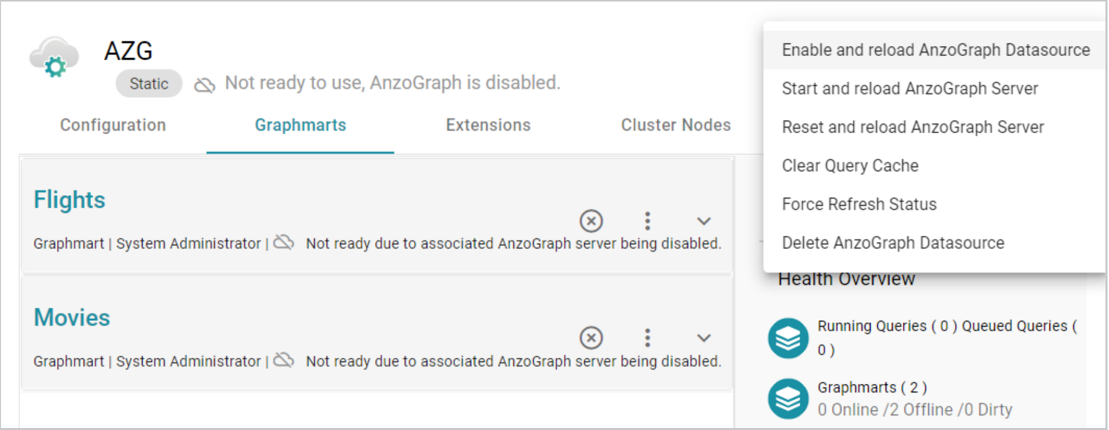
□ Readonly Replica ☑ Vacuum ☑ Gather Statistics on Load

- Instance URI
- Trust All TLS Certificates
- AnzoGraph Concurrent Queries

- AnzoGraph Connection Timeout
- Use AnzoGraph Persistence if Available
- Force Reload of Graphmart Data During AnzoGraph Activation or Reconnection
- Keep AnzoGraph Datasource Enabled on Anzo Startup
- Port
- AnzoGraph Management Port
- Callback HostName
- Readonly Replica
- Vacuum
- Gather Statistics on Load
- Use Priority Queue Query Manager
- Enable Detailed Query Timing
- Max Allowed Duration for System Operations
- Max Allowed Duration for Queries
- Use Minimal Number of SPARQL Rewriters

Setting	Description
Instance URI	Defines the URI for this AnzoGraph instance. When this setting is empty Anzo automatically assigns an instance URI. If you specify a custom URI, make sure that the URI is valid and unique.
Trust All TLS Certificates	Indicates whether Anzo should trust the AnzoGraph certificates for this connection. Cambridge Semantics recommends that you accept the default value of enabled.
AnzoGraph Concurrent Queries	Specifies the maximum number of queries that Anzo can send to AnzoGraph concurrently. The default value is **10** queries. Cambridge Semantics recommends that you accept the default value. If you want to increase the number of concurrent queries, Cambridge Semantics recommends that you choose a value between 10 and 20.
AnzoGraph Connection Timeout	Controls how often (in seconds) Anzo checks the status of the connection to this AnzoGraph instance. The connection is tested every *N* seconds, where *N* is the value of this setting. The default value is **60**. If the test fails, Anzo re-tests the connection

Setting	Description
	every 15 seconds for 2 minutes to rule out a brief network glitch. If the connection continues to fail after 2 minutes, the status is changed to "Offline." If the connection is re-established within the 2-minute window, Anzo determines whether the connection came back automatically or whether AnzoGraph was restarted.
Use AnzoGraph Persistence if Available	Controls how Anzo manages graphmart data if persistence is enabled for this data source and AnzoGraph is restarted. **Note** The **Use AnzoGraph Persistence if Available** setting is enabled by default but persistence is disabled for AnzoGraph by default. For information about how Anzo manages the data when persistence is enabled and for instructions on enabling persistence, see Using AnzoGraph Persistence (Preview).
Force Reload of Graphmart Data During AnzoGraph Activation or Reconnection	This option is enabled by default and means that Anzo forces a reload of active graphmarts when one of the following actions occur: 1. Anzo restarts and reconnects to AnzoGraph. 2. Anzo restarts and a user manually re-enables this data source by selecting **Enable and reload AnzoGraph Datasource** from the menu on the AnzoGraph administration screen. When this option is disabled and AnzoGraph persistence is also disabled, graphmarts must be reloaded by clicking the **Reset and Reload all Graphmarts** button on the AnzoGraph screen after the connection is re-established due to an AnzoGraph restart. **Note** If AnzoGraph persistence is enabled and **Force reload of Graphmart data...** is disabled, Anzo may force a reload if the last updated timestamp in AnzoGraph does not match the last updated value in Anzo.
Keep AnzoGraph	This option is enabled by default and means that Anzo leaves the AnzoGraph data

Setting	Description
Datasource Enabled on Anzo Startup	source online in a "Ready to use" state if Anzo is restarted (if this data source is online at the time Anzo is restarted). When this option is disabled, Anzo disables this data source when Anzo is restarted. When Anzo comes online, this source must be manually enabled by selecting **Enable and reload AnzoGraph Datasource** from the menu on the AnzoGraph administration screen. For example:  AZG Static Not ready to use, AnzoGraph is disabled. Configuration Graphmarts Extensions Cluster Nodes Flights Graphmart | System Administrator | Not ready due to associated AnzoGraph server being disabled. Movies Graphmart | System Administrator | Not ready due to associated AnzoGraph server being disabled. Enable and reload AnzoGraph Datasource Start and reload AnzoGraph Server Reset and reload AnzoGraph Server Clear Query Cache Force Refresh Status Delete AnzoGraph Datasource Health Overview Running Queries (0) Queued Queries (0) Graphmarts (2) 0 Online /2 Offline /0 Dirty
Port	The port to use for communication between AnzoGraph and Anzo. The default value is **5700**, the Anzo protocol (gRPC) port for secure communication. Do not change the value unless instructed by Cambridge Semantics Support.
AnzoGraph Management Port	The SSL system management port for AnzoGraph. It is the port that Anzo uses to connect to the system manager and, in a cluster, the AnzoGraph system managers use to communicate to each other across the cluster. The default value is **5600**. Do not change the value unless instructed by Cambridge Semantics Support.
Callback HostName	The Anzo instance to call when AnzoGraph makes service callbacks. If you have multiple Anzo servers and one or more of them are not routable by the AnzoGraph server, the Callback HostName is the Anzo host that AnzoGraph can target when making service calls.
Readonly Replica	This option is for use if you have multiple Anzo servers and only one of those servers loads graphmarts to AnzoGraph. When **Readonly Replica** is selected, Anzo treats this AnzoGraph instance as a read-only source so that Anzo can view the data in AnzoGraph but cannot change it.

Setting	Description
Vacuum	Controls whether Anzo initiates an AnzoGraph vacuum process after each data load. The vacuum process improves data organization in memory, deduplicates data, and reclaims memory after data is deleted. Completing a vacuum after update operations is extremely important for maintaining overall query performance and memory allocation accuracy. **Note** Do not disable vacuum unless you are instructed to do so by Cambridge Semantics Support.
Gather Statistics on Load	Controls whether Anzo initiates AnzoGraph's internal statistics gathering queries after loading data. Gathering statistics helps the query planner generate ideal query execution plans when queries are run. When this option is enabled, the AnzoGraph statistics queries are run immediately after a Graphmart is loaded. It increases Graphmart load time but reduces execution time for the first analytic queries, such as when a Hi-Res Analytics Dashboard is created. When this option is disabled (the checkbox is clear), AnzoGraph automatically performs statistics gathering when the first queries are run, increasing the execution time for the initial queries. **Note** Cambridge Semantics recommends that you leave Gather Statistics on Load enabled so that AnzoGraph gathers statistics at the end of a load rather than during query execution. Since loads take longer than queries, adding more time to the load is less noticeable than waiting for statistics to be generated during initial query execution.
Use Priority Queue Query Manager	Controls whether Anzo provides a view of the queries that are in the queue waiting to be run. The queued queries are displayed in the System Query Audit log. **Note** Enabling or disabling this option after saving the initial configuration requires a restart of Anzo.

Setting	Description
Enable Detailed Query Timing	When the Priority Queue Query Manager is enabled, this option controls whether Anzo obtains detailed timing statistics for every AnzoGraph query. If this option is enabled, Anzo sends additional statistics gathering queries to AnzoGraph for each user query. The extra query timing details, such as query compilation time, compilation statistics, and a query summary, are displayed in the System Query Audit log. For more information about this setting, see AnzoGraph Detailed Query Timing Reference. **Important** Enabling detailed query timing increases the AnzoGraph workload and may decrease overall query performance.
Max Allowed Duration for System Operations	Sets a limit on the duration of time Anzo waits for AnzoGraph to complete system operation related queries, such as queries for CPU and memory usage statistics. The default value is **2** minutes. If Anzo is waiting on system information from AnzoGraph and AnzoGraph does not respond within the specified time, Anzo cancels the request.
Max Allowed Duration for Queries	Sets a limit on the amount of time that Anzo waits for AnzoGraph to complete a user query (such as dashboard, data layer, or Query Builder queries). By default, Anzo waits indefinitely. To set a maximum duration, specify the amount of time in any combination of days, hours, and minutes. For example, specifying **1d** sets the maximum duration to one day. Specifying **10h**, sets the maximum duration to 10 hours, and specifying **1d12h30m** sets the duration to 1 day, 12 hours, and 30 minutes. If **Max Allowed Duration for Queries** is set and a query does not complete in the specified time, Anzo cancels the request regardless of whether AnzoGraph has returned partial results.
Use Minimal Number of SPARQL Rewriters	When Anzo processes SPARQL queries before sending them to AnzoGraph, there is a set of rewrites it makes to try to optimize the query execution. This setting controls whether Anzo performs the full set of rewrites to optimize the query or whether it performs only the minimal required modifications. When this setting is disabled (the default value) Anzo performs the full set of rewrites. When this setting is enabled, Anzo performs only a minimal set of rewrites.

Setting	Description
	Note Do not enable this setting unless you are instructed to do so by Cambridge Semantics Support.

Related Topics

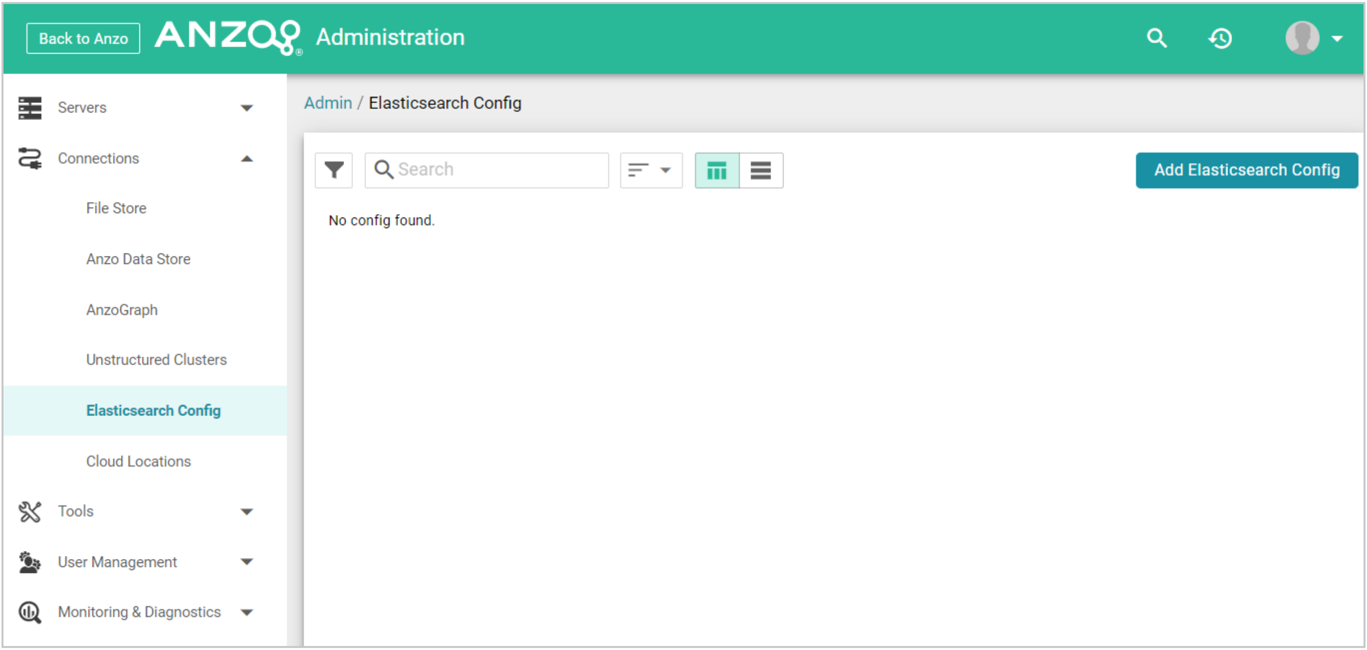
Connecting to AnzoGraph

AnzoGraph Server Administration

Connecting to Elasticsearch

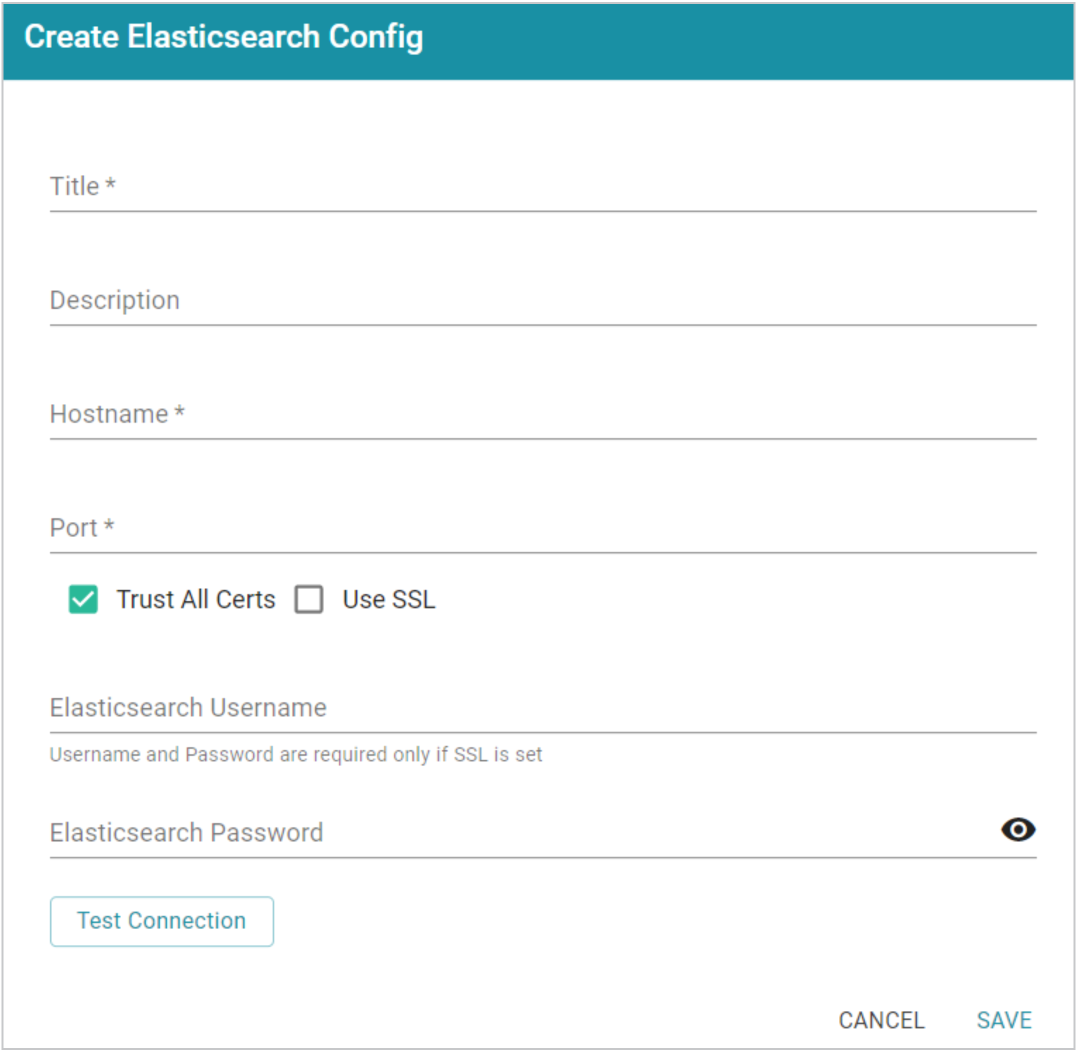
This topic provides instructions for configuring a connection to an Elasticsearch instance in the Administration application. For information about installing Elasticsearch, see Installing and Configuring Elasticsearch in the Deployment Guide.

1. In the Administration application, expand the **Connections** menu and click **Elasticsearch Config**. Anzo displays the Elasticsearch Config screen, which lists any existing Elasticsearch connections. For example:



No config found.

2. On the Elasticsearch Config screen, click the **Add Elasticsearch Config** button. Anzo opens the Create Elasticsearch Config dialog box.



Create Elasticsearch Config

Title *

Description

Hostname *

Port *

☑ Trust All Certs □ Use SSL

Elasticsearch Username

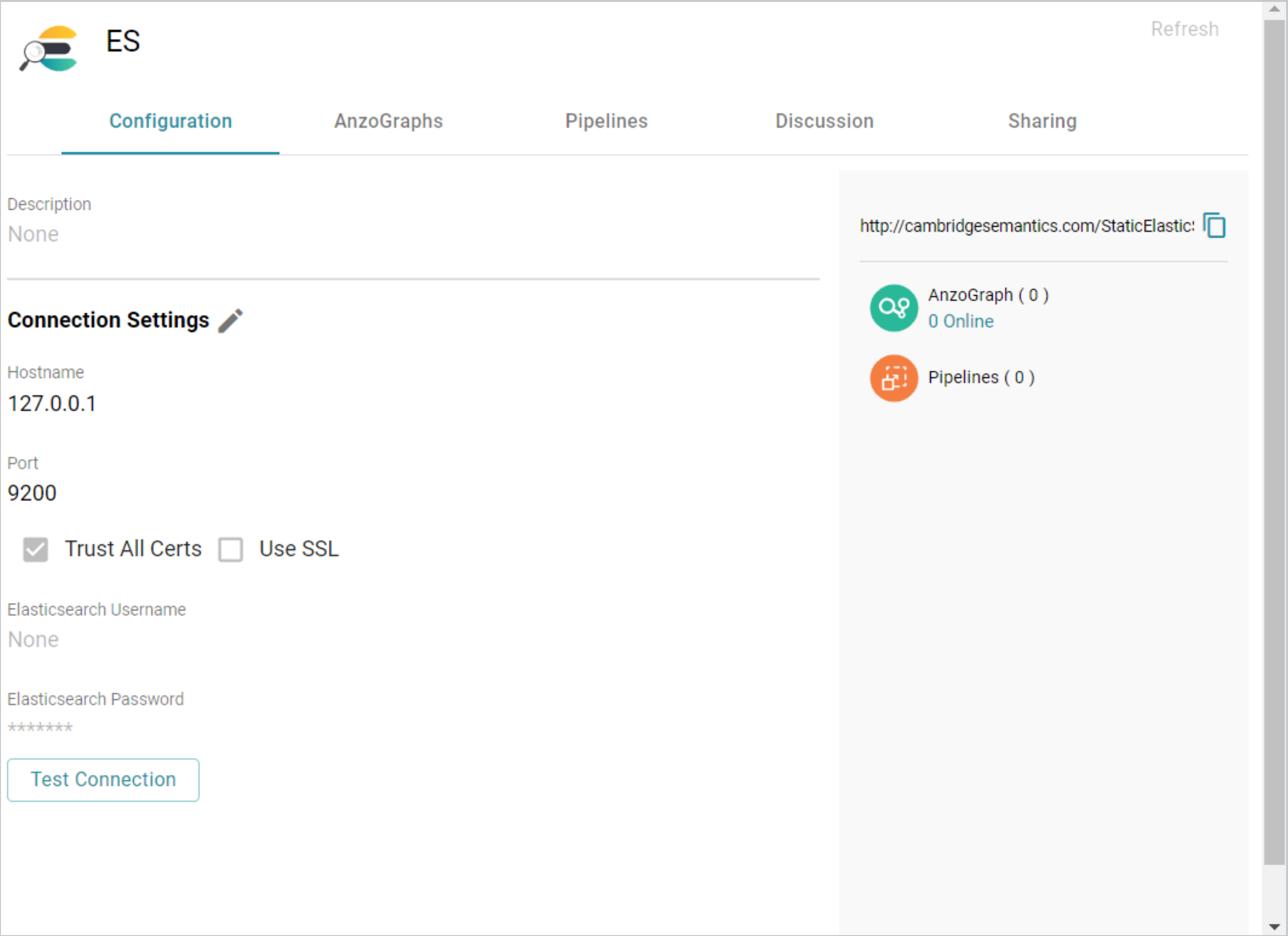
Username and Password are required only if SSL is set

Elasticsearch Password

3. On the Create Elasticsearch Config screen, provide the following details about the Elasticsearch instance:
   - **Title**: Type a name for this Elasticsearch connection.
   - **Description**: Optional description for this connection.
   - **Hostname**: Specify the IP address or hostname of the Elasticsearch server.
   - **Port**: Specify the port to use for the Elasticsearch connection. The default Elasticsearch port is **9200**.
   - **Trust All Certs**: Indicates whether Anzo should trust the Elasticsearch certificates for this connection. Cambridge Semantics recommends that you accept the default value of enabled.
   - **Use SSL**: If this Elasticsearch instance is configured for SSL authentication, select the **Use SSL** checkbox.

- **Elasticsearch Username**: If Use SSL is specified, type the user name to use to connect to Elasticsearch.
- **Elasticsearch Password**: If Use SSL is specified, type the password for the user name that you specified.

4. Click **Test Connection** to check if Anzo can connect to Elasticsearch. If the connection fails, make sure that Elasticsearch is running and that you entered the correct connection details.
5. Anzo displays a Connection Successful dialog box. Click **OK** to close the dialog, and then click **Save** to save the new connection. Anzo saves the connection and displays the Configuration overview screen. For example:



Description
None

Connection Settings

Hostname
127.0.0.1

Port
9200

☑ Trust All Certs ☐ Use SSL

Elasticsearch Username
None

Elasticsearch Password
****

http://cambridgesemantics.com/StaticElastic:

AnzoGraph (0)
0 Online

Pipelines (0)

You can adjust configuration details as needed. To connect this Elasticsearch instance to an AnzoGraph instance, view the configuration details for the AnzoGraph instance and choose this Elasticsearch connection in the **Elasticsearch Configuration** field.

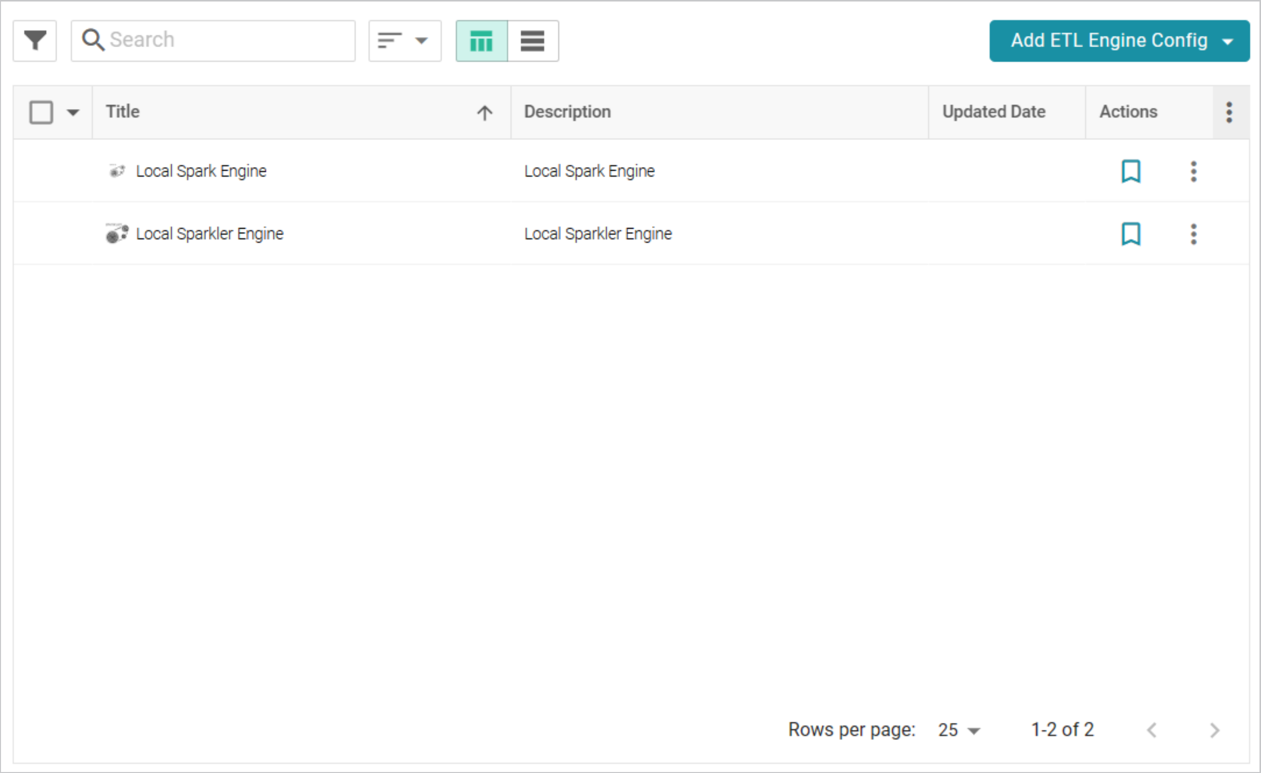
Connecting to an ETL Engine

The default Anzo installation includes an optional pre-configured local Spark ETL engine and Sparkler ETL compiler. Sparkler is Cambridge Semantics' Spark SPARQL interpreter. The Sparkler interpreter expresses Spark ingestion jobs as SPARQL, which adds benefits such as support for ingesting wide CSV files with a large number of columns. The topics in this section provide instructions for changing the configuration of the local engines or connecting to an alternate Spark ETL engine or Sparkler compiler.

Configuring a Spark ETL Engine

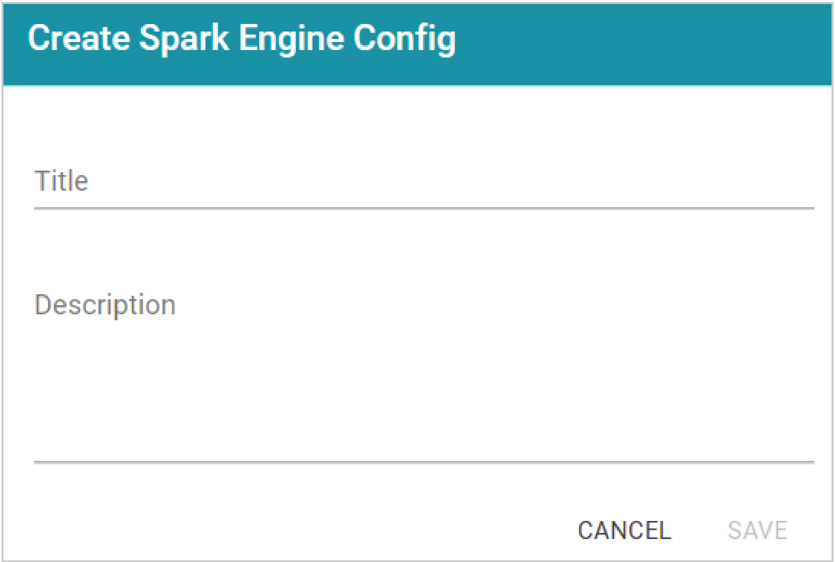
This topic provides instructions for configuring a connection to a Spark ETL engine.

1. In the Administration application, expand the **Connections** menu and click **ETL Engine Config**. Anzo displays the ETL Engine Config screen, which lists existing ETL engine connections. For example:

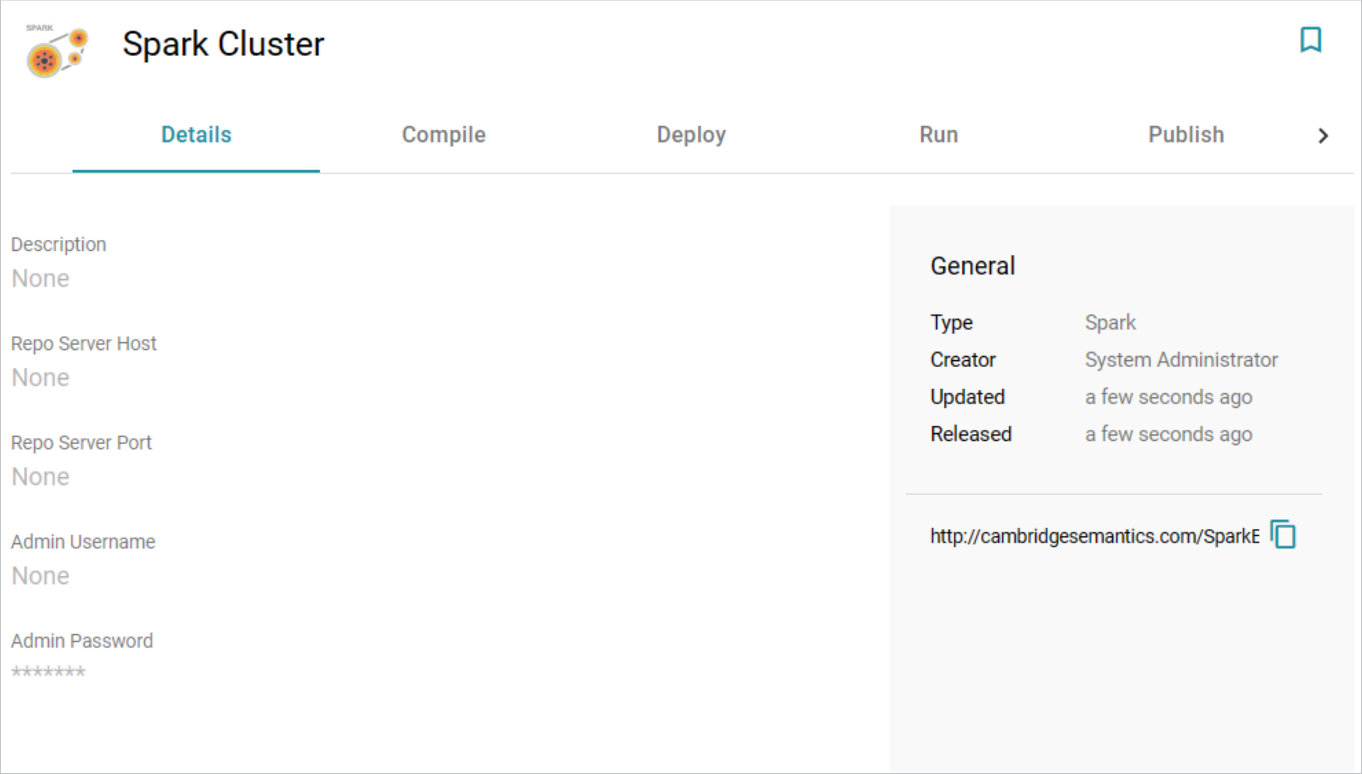


Title	Description	Updated Date	Actions
Local Spark Engine	Local Spark Engine		
Local Sparkler Engine	Local Sparkler Engine		

2. On the ETL Engine Config screen, click the **Add ETL Engine Config** button and select **Spark Engine Config**. Anzo displays the Create Spark Engine Config screen.



3. On the Create screen, type a **Title** and optional **Description** for the engine. Then click **Save**. Anzo displays the Details view for the new engine. For example:



Description
None

Repo Server Host
None

Repo Server Port
None

Admin Username
None

Admin Password
****

Type	Spark
Creator	System Administrator
Updated	a few seconds ago
Released	a few seconds ago

http://cambridgesemantics.com/SparkE

4. Configure the engine by completing the required fields and adding any optional values on the Details, Compile, Deploy and Run tabs. To edit a field, click a value to make the field editable or click the edit icon (✎). Click the check mark icon (✓) to save changes to an option, or click the X icon (✕) to clear

the value for an option. See the Spark Settings Reference section below for descriptions of the settings.

Spark Settings Reference

This section provides reference information for the Spark ETL engine settings on each of the tabs.

Details Tab

- **Repo Server Host**: Leave this field blank.
- **Repo Server Port**: Leave this field blank.
- **Admin Username**: Not currently used.
- **Admin Password**: Not currently used.

Compile Tab

The Compile settings control where Anzo saves the compiled Scala .jar files for the Spark job.

- **Remote Server**: The host name or IP address of the server where the compilation will be performed.
- **Target Folder**: The path and directory on the server where Anzo can stage temporary artifacts created during the compilation and upload process. The location must be a valid path on the Anzo server that the user running the ETL job has access to.

Deploy Tab

The Deploy step is performed after the job is compiled locally and before the job is submitted to Spark. The Deploy settings control how and where the job's .jar files will be copied from the Anzo server to a file system that Spark can access.

- **Deployment Working Dir**: The directory that the Anzo server should use when executing the deploy commands.
- **Deploy Command**: The command line script that the deploy step should run.

Run Tab

- **Job Runner Endpoint**: The HTTP endpoint used to reach the Livy server. For example, when using the local Anzo Spark engine, the endpoint is localhost:8998.
- **SDI Jobs Dir**: The file system location where the Spark engine will look for the compiled .jar files. This field is required when working with a remote Spark server. It can be left blank when using the local Spark engine.

- **SDI Dependencies Dir**: The file system location where the Spark engine will look for the dependency .jar files, **sdi-full-deps.jar** and **sdi-deps.jar**. If you are using a remote Spark cluster, sdi-full-deps.jar and sdi-deps.jar can be copied to the Spark master node from the `<install_path>/Server/data/sdiScripts/<Spark_version>/compile/dependencies-lib` directory on the Anzo server.
- **Additional Jars**: For relational database sources, this field lists the file system location for the JDBC driver .jar file or files that are used to connect to the source. All paths must be absolute. For multiple jar files, specify a comma-separated list. Do not include a space after the commas.

  For RDBs whose drivers are installed with Anzo, such as MSSQL (com.springsource.net.sourceforge.jtds_1.2.2.jar), Oracle (oracle.jdbc_11.2.0.3.jar), Amazon Redshift (org.postgresql.osgi.redshift_9.3.702.jar), and PostgreSQL (com.springsource.org.postgresql.jdbc3_8.3.603.jar), you can find the driver jar files in the `<install_path>/Server/plugins` directory.

  - If you use the local Spark ETL engine, the Additional Jars field should list the path to the jar files in the Anzo plugins directory. For example, `/opt/Anzo/Server/plugins/org.postgresql.osgi.redshift_9.3.702.jar`.
  - If you use a remote Spark cluster in **cluster mode**, the driver jar files need to be copied onto the HDFS. If Spark is running in **client mode**, jar files can be copied to the Hadoop/Spark master node file system. Specify the path to the copied jar files in the Additional Jars field.

  > **Note**
  > If a driver is uploaded to Anzo as described in Uploading a Plugin, the driver will be in the `<install_path>/Server/dropins` directory. For example,
  > `/opt/Anzo/Server/dropins/com.springsource.com.mysql.jdbc-5.1.6.jar`

- **Execute Locally**: Select this option for local Spark engines on the Anzo server. Make sure this option is not selected when using a remote Spark server.
- **Do Callback**: Select this option when you want Anzo to create a new data set in the Dataset catalog and generate load files for the graph source.
- **Run with Yarn**: Employs the Spark YARN cluster manager when running ETL jobs.
- **Callback URL**: When **Do Callback** is selected, enter one of the following URLs:

  ```
  http://Anzo_hostname_or_IP:Anzo_app_HTTP_port/anzoclient/call
  ```

  ```
  https://Anzo_hostname_or_IP:Anzo_app_HTTPS_port/anzoclient/call
  ```

For example:

```
https://10.100.0.1:8443/anzoclient/call
```

Publish Tab

The Publish tab controls the action of the **Publish All** button when a pipeline is published.

Sharing Tab

The Sharing tab enables you to share or restrict access to this ETL engine.

When the configuration is complete, Anzo provides this ETL engine as a choice to select when ingesting data and configuring pipelines. If you want to specify the default ETL engine to use automatically any time a pipeline is configured, see Configure the Default ETL Engine.

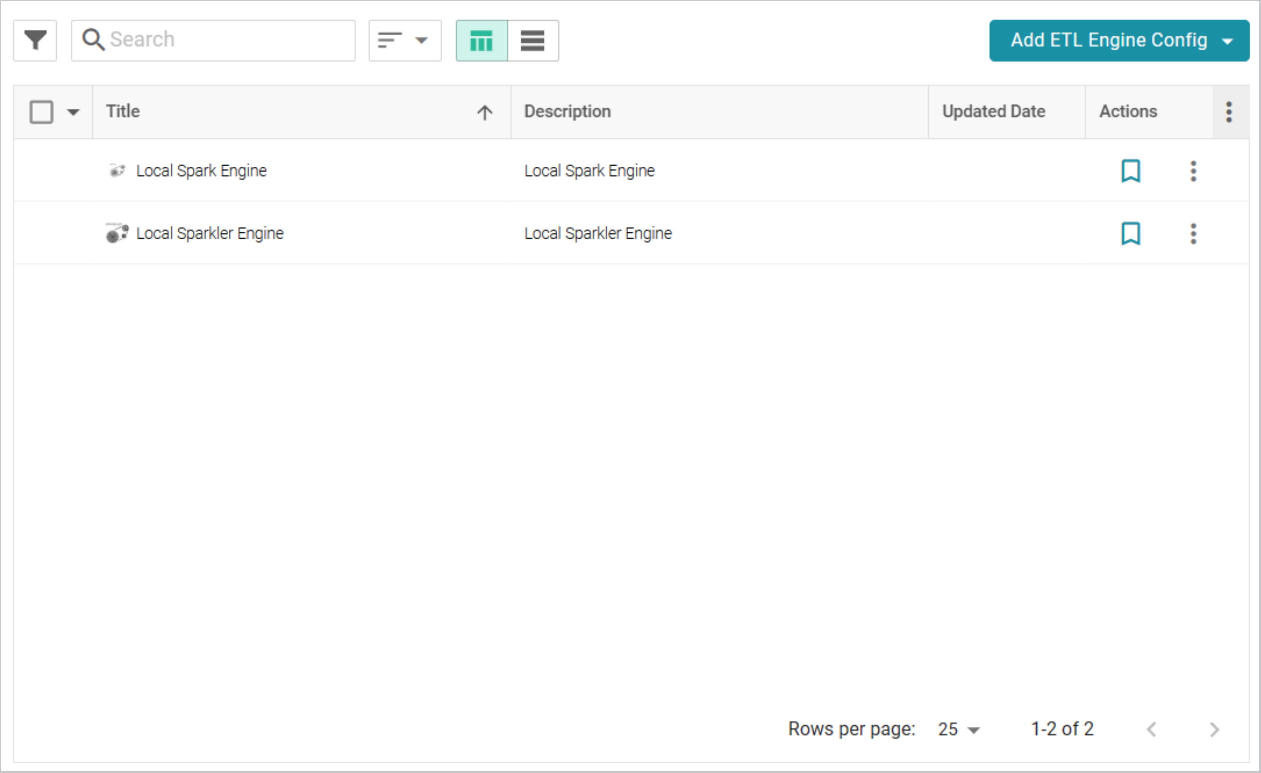
Related Topics

Configuring a Sparkler Engine

Configuring a Sparkler Engine

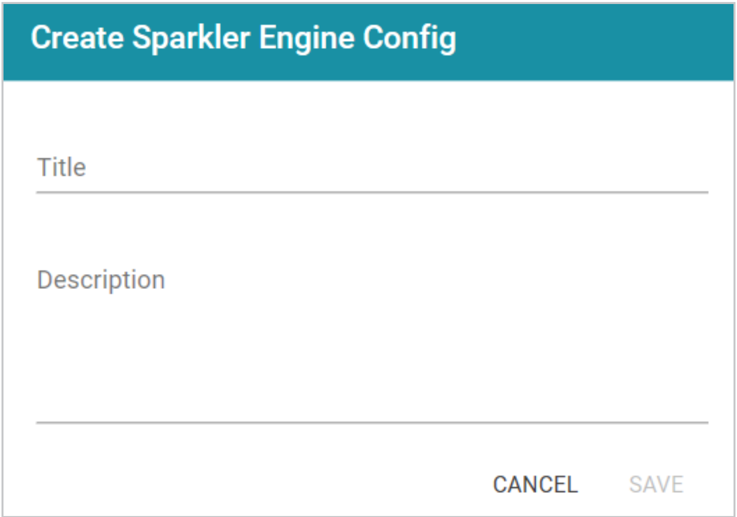
This topic provides instructions for configuring a connection to a Sparkler compiler. Sparkler is Cambridge Semantics' Spark SPARQL interpreter. Sparkler expresses Spark ingestion jobs as SPARQL, and Sparkler jobs are executed by Spark. They are submitted to Spark using Livy interactive sessions.

1. In the Administration application, expand the **Connections** menu and click **ETL Engine Config**. Anzo displays the ETL Engine Config screen, which lists existing ETL engine connections. For example:



Title	Description	Updated Date	Actions
Local Spark Engine	Local Spark Engine		
Local Sparkler Engine	Local Sparkler Engine		

2. On the ETL Engine Config screen, click the **Add ETL Engine Config** button and select **Sparkler Engine Config**. Anzo displays the Create Sparkler Engine Config screen.



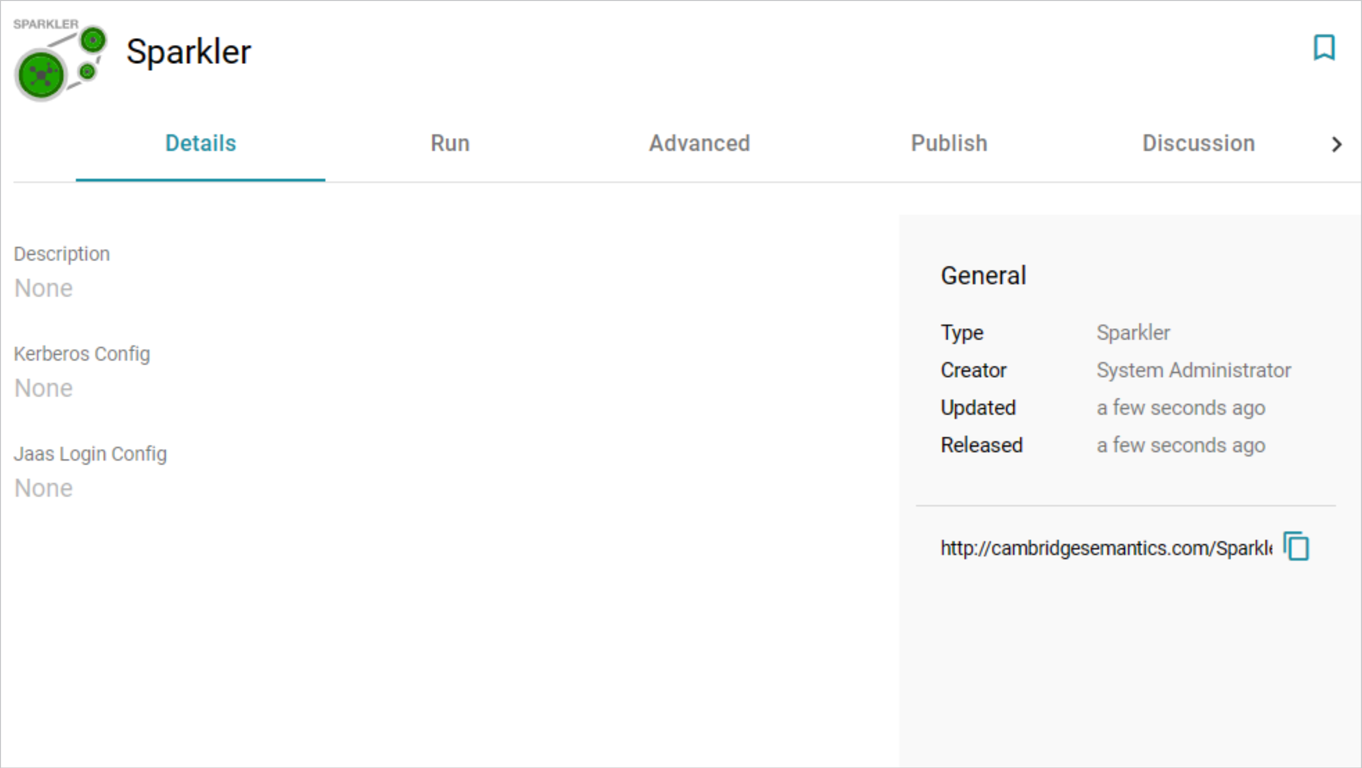
Create Sparkler Engine Config

Title

Description

CANCEL SAVE

3. On the Create screen, type a **Title** and optional **Description** for the engine. Then click **Save**. Anzo displays the Details view for the new engine. For example:



Sparkler

Details | Run | Advanced | Publish | Discussion

Description
None

Kerberos Config
None

Jaas Login Config
None

General

Type	Sparkler
Creator	System Administrator
Updated	a few seconds ago
Released	a few seconds ago

http://cambridgesemantics.com/Sparkl

4. Configure the engine by completing the required fields and adding any optional values on the Run, Advanced, and Publish tabs. To edit a field, click a value to make the field editable or click the edit icon (✎). Click the check mark icon (✓) to save changes to an option, or click the X icon (✕) to clear the value for an option. See the Sparkler Settings Reference section below for descriptions of the settings.

Sparkler Settings Reference

This section provides reference information for the Sparkler ETL engine settings on each of the tabs.

Run Tab

- **Remote Server Name**: The host name or IP address of the server where the compilation will be performed.
- **Job Runner Endpoint**: The HTTP endpoint used to reach the Livy server. For example, when using the local Anzo Sparkler engine, the endpoint is localhost:8998.
- **Target Folder Name**: The path and directory on the host where temporary artifacts can be created during the compilation and upload process. The location must be a valid path on the server that the user running the ETL job has access to.
- **Sparkler Home**: The path and directory where the Sparkler compiler is installed on the host server.
- **SDI Dependencies Dir**: The file system location where the Spark engine will look for the dependency .jar files, **sdi-full-deps.jar** and **sdi-deps.jar**. If you are using a remote Spark cluster, sdi-full-deps.jar and sdi-deps.jar can be copied to the Spark master node from the `<install_path>/Server/data/sdiScripts/<Spark_version>/compile/dependencies-lib` directory on the Anzo server.
- **Additional Jars**: For relational database sources, this field lists the file system location for the JDBC driver .jar file or files that are used to connect to the source. All paths must be absolute. For multiple jar files, specify a comma-separated list. Do not include a space after the commas.

  For RDBs whose drivers are installed with Anzo, such as MSSQL (com.springsource.net.sourceforge.jtds_1.2.2.jar), Oracle (oracle.jdbc_11.2.0.3.jar), Amazon Redshift (org.postgresql.osgi.redshift_9.3.702.jar), and PostgreSQL (com.springsource.org.postgresql.jdbc3_8.3.603.jar), you can find the driver jar files in the `<install_path>/Server/plugins` directory.

  - If you use the local Sparkler ETL engine, specifying the path to drivers in the Additional Jars field is not mandatory. Anzo will automatically locate the drivers in the plugins directory. If you do list the path to the jar files, specify the path to the Anzo plugins directory. For example, `/opt/Anzo/Server/plugins/org.postgresql.osgi.redshift_9.3.702.jar`.
  - If you use a remote Spark cluster in **cluster mode**, the driver jar files need to be copied onto the HDFS. If Spark is running in **client mode**, jar files can be copied to the Hadoop/Spark master node file system. Specify the path to the copied jar files in the Additional Jars field.

> **Note**
>
> If a driver is uploaded to Anzo as described in Uploading a Plugin, the driver will be in the `<install_path>/Server/dropins` directory. For example, `/opt/Anzo/Server/dropins/com.springsource.com.mysql.jdbc-5.1.6.jar`

- **Execute Locally**: Select this option for local Sparkler engines on the Anzo server. Make sure this option is not selected when using a remote Sparkler server.
- **Do Callback**: Select this option when you want Anzo to create a new data set in the Dataset catalog and generate load files for the graph source.
- **Run with Yarn**: Employs the Spark YARN cluster manager when running ETL jobs.
- **Callback URL**: When **Do Callback** is selected, enter one of the following URLs:

  ```
  http://Anzo_hostname_or_IP:Anzo_app_HTTP_port/anzoclient/call
  ```

  ```
  https://Anzo_hostname_or_IP:Anzo_app_HTTPS_port/anzoclient/call
  ```

  For example:

  ```
  https://10.100.0.1:8443/anzoclient/call
  ```

Advanced Tab

The options on this tab enable users with advanced Spark expertise to customize the values that are passed to Spark.

- **Enable CSV Error Reporting**: Controls whether detailed CSV errors are displayed in the Anzo user interface.
- **Input Database Partition Default**: By default, Sparkler attempts to partition relational database tables if the table has a primary column with an integer data type and the source data has been profiled (as described in Generating a Source Data Profile in the User Guide). When **Input Database Partition Default** is enabled, Sparkler attempts to partition RDBMS tables when they have a primary column with an integer type even if a data source profile has not been generated.
- **Enable Hive Context (Enable in Livy Conf for Spark 2)**: Controls Hive context for Spark version 1.6. Selecting this setting enables the Hive context for Spark 1.6.
- **Redirect Graph Output to Hive**: Controls whether the ETL process writes data to Hive or a file-based linked data set (FLDS). When this option is disabled (the default configuration) data is written to an

FLDS that can be added to a graphmart and loaded to AnzoGraph. When this option is enabled, the ETL process writes data to Hive rather than creating an FLDS.

- **Run As User**: Specifies the user to impersonate when starting the Livy session.
- **Max Graph Output File Size Default (Bytes)**: The maximum number of bytes to limit graph output files to.
- **Max Input File Partition Size (Bytes)**: The maximum number of bytes to pack into a partition when reading files. Maps to the `spark.files.maxPartitionBytes` Spark configuration setting.
- **Spark Job Driver Cores**: The number of cores to use for the driver process. Maps to the `spark.driver.cores` Spark configuration setting.
- **Spark Job Driver Memory**: The amount of memory to use for the driver process. Maps to the `spark.driver.memory` Spark configuration setting.
- **Number of Executors Per Spark Job**: The number of executors to request per Spark job. Maps to the `spark.executor.instances` Spark configuration setting.
- **Spark Job Cores Per Executor**: The number of cores to use on each executor. Maps to the `spark.executor.cores` Spark configuration setting.
- **Spark Job Memory Per Executor**: The amount of memory to use per executor process. Maps to the `spark.executor.memory` Spark configuration setting.
- **Off Heap Size (Bytes)**: The amount of memory in bytes that can be used for off-heap allocation. Maps to the `spark.memory.offHeap.size` Spark configuration setting.
- **Job Dependencies (Maven Package Coordinate)**: The comma-separated list of Maven jar coordinates to include on the driver and executor classpaths. Maps to the `spark.jars.packages` Spark configuration setting.
- **Maven Package Excludes**: To avoid dependency conflicts, this is the comma-separated list of **groupId:artifactId** to exclude while resolving the dependencies listed in spark.jars.packages. Maps to the `spark.jars.excludes` Spark configuration setting.
- **Maven Repositories**: A comma-separated list of additional remote repositories to search for the maven coordinates from the Job Dependencies setting. Maps to the `spark.jars.repositories` Spark configuration setting.
- **Spark Job Deploy Mode (Livy Config has Precedence)**: The deploy mode of the Spark driver program. If this value is set in the Livy configuration, the Livy value takes precedence. Maps to the `spark.submit.deployMode` Spark configuration setting.

Publish Tab

The Publish tab controls the action of the **Publish All** button when a pipeline is published.

Sharing Tab

The Sharing tab enables you to share or restrict access to this ETL engine.

When the configuration is complete, Anzo provides this ETL engine as a choice to select when ingesting data and configuring pipelines. If you want to specify the default ETL engine to use automatically any time a pipeline is configured, see Configure the Default ETL Engine.

Related Topics

Limiting Job Concurrency on a Remote Sparkler Engine

Limiting Job Concurrency on a Remote Sparkler Engine

When compiling ETL jobs on a remote Sparkler engine, all jobs are executed simultaneously. For pipelines with more than 110 jobs, running all jobs concurrently can consume all of ports in the default port range and cause the pipeline to fail. To limit the number of jobs that can be executed concurrently on a remote Spark cluster with Sparkler, you can add a configuration file to the cluster and specify the maximum number of jobs that can be executed at the same time. When the number of jobs exceeds the limit, additional jobs are queued and then executed as resources are freed. Follow the instructions below to configure the limit.

1. If necessary, run the following command to stop the remote Sparkler server:

   ```
   ./<install_path>/sparkler/bin/sparkler-server stop
   ```

2. The Anzo embedded Sparkler engine includes a configuration file template, **application.conf.template**, that you can copy to the remote cluster. If needed, you can retrieve **application.conf.template** from the following directory on the Anzo server:

   ```
   <install_path>/Server/data/sdiScripts/spark-2.2/compile/dependencies-
   lib/sparkler/conf
   ```

3. Rename application.conf.template to **application.conf** and place applicaton.conf in the `<install_path>/sparkler/conf/` directory on the remote cluster.
4. Open application.conf in an editor. At the top of the file under server options, change the value for **maxActiveJobs** to the maximum number of jobs that you want Sparkler to execute concurrently. The setting and default value are shown in bold below:

   ```
   server {
     actorSystemName = "SparklerServerSystem"
     actorName = "SparklerJobActor"
     retryDelay = "3 seconds"
     maxRetries = 5
     maxActiveJobs = 1
   ...
   ```

5. Save and close application.conf, and then run the following command to restart the Sparkler server:

   ```
   ./<install_path>/sparkler/bin/sparkler-server start
   ```

Related Topics

Configuring a Sparkler Engine

Connecting to a Cloud Location

A Cloud Location is a connection between Anzo and the Kubernetes (K8s) cluster that will host the dynamic Anzo Agent and Anzo Unstructured, AnzoGraph, Spark, and Elasticsearch applications. When you create a Cloud Location, Anzo discovers the K8s cluster and any internal container registries, authenticates the K8s API services, obtains the node pool or group specifications and retrieves pricing information from the Cloud Service Provider for the configured compute instances, and maps the node pool specifications to Launch Configurations in Anzo.

> **Tip**
> For instructions on deploying the K8s infrastructure to support Cloud Locations, see Using K8s for Dynamic Deployments of Anzo Components in the Deployment Guide.

The topics in this section provide instructions on setting up the NFS configuration for the dynamically deployed applications and creating a Cloud Location.

Importing the NFS Configuration

Before creating a Cloud Location in the Administration application, the configuration details for the NFS server or servers need to be imported into Anzo. This is a one-time procedure; the configuration that you import is used for all Cloud Locations. Anzo will automatically mount the NFS server to any nodes that are provisioned when applications are deployed.

> **Tip** For information about the NFS requirements, see NFS Guidelines in the Deployment Guide.

Create the NFS Configuration File

The NFS configuration details need to be specified in TriG format. The TriG file is imported to Anzo using the Anzo Admin CLI. Use the following contents as a template to create a .trig file on the Anzo server. If you have multiple NFS servers for different regions, you can configure each server in the same configuration file. The objects to supply values for are described below:

```
@prefix : <http://cambridgesemantics.com/ontologies/cloud/deployment/config#> .
@prefix nfsmountconfig:
<http://cambridgesemantics.com/ontologies/CloudDeployment/NFSMountConfiguratio
n/> .
@prefix deployment: <http://cambridgesemantics.com/ontologies/CloudDeployment/>
.
@prefix anzo: <http://openanzo.org/ontologies/2008/07/Anzo#> .
@prefix int:  <http://openanzo.org/system/internal/> .
@prefix role: <http://openanzo.org/Role/> .

#Mode:REPLACE
:nfsMountConfig1
{
  :nfsMountConfig1 a deployment:NFSMountConfiguration,
deployment:MountConfiguration;
     nfsmountconfig:NFSfqdn "NFSfqdn" ;
     nfsmountconfig:NFSMountDir "NFSMountDir" ;
     nfsmountconfig:NFSMountOptions "NFSMountOptions" ;
     nfsmountconfig:NFSSharedDir "NFSSharedDir" .
}
# :nfsMountConfig2
# {
#  :nfsMountConfig2 a deployment:NFSMountConfiguration,
deployment:MountConfiguration;
#     nfsmountconfig:NFSfqdn "NFSfqdn2" ;
#     nfsmountconfig:NFSMountDir "NFSMountDir2" ;
#     nfsmountconfig:NFSMountOptions "NFSMountOptions2" ;
#     nfsmountconfig:NFSSharedDir "NFSSharedDir2" .
# }
# ...
```

NFSfqdn

The IP address for the NFS server.

NFSMountDir

The NFS mount location on the Anzo server. The same mount location will be used to mount the NFS when dynamic resources are provisioned.

NFSMountOptions

The mount options to use when mounting the NFS.

NFSSharedDir

The NFS directory to share between Anzo and the dynamic resources.

For example:

```
# nfs-config.trig
@prefix : <http://cambridgesemantics.com/ontologies/cloud/deployment/config#> .
@prefix nfsmountconfig:
<http://cambridgesemantics.com/ontologies/CloudDeployment/NFSMountConfiguratio
n/> .
@prefix deployment: <http://cambridgesemantics.com/ontologies/CloudDeployment/>
.
@prefix anzo: <http://openanzo.org/ontologies/2008/07/Anzo#> .
@prefix int:  <http://openanzo.org/system/internal/> .
@prefix role: <http://openanzo.org/Role/> .

#Mode:REPLACE
:nfsMountConfig1
{
  :nfsMountConfig1 a deployment:NFSMountConfiguration,
deployment:MountConfiguration;
```

```
    nfsmountconfig:isTransferFiles false ;
    nfsmountconfig:NFSfqdn "10.104.0.6" ;
    nfsmountconfig:NFSMountDir "/private/var/nfsshare_dev" ;
    nfsmountconfig:NFSMountOptions "hard,nfsvers=4.1" ;
    nfsmountconfig:NFSSharedDir "/global/nfs/data" .
}
```

Import the NFS Configuration to Anzo

Once the NFS configuration file is created, run the following command to import the file to Anzo with the Anzo Admin CLI:

```
<install_path>/Client/anzo <file_path>/<filename>.trig -u sysadmin --useModes
```

For example:

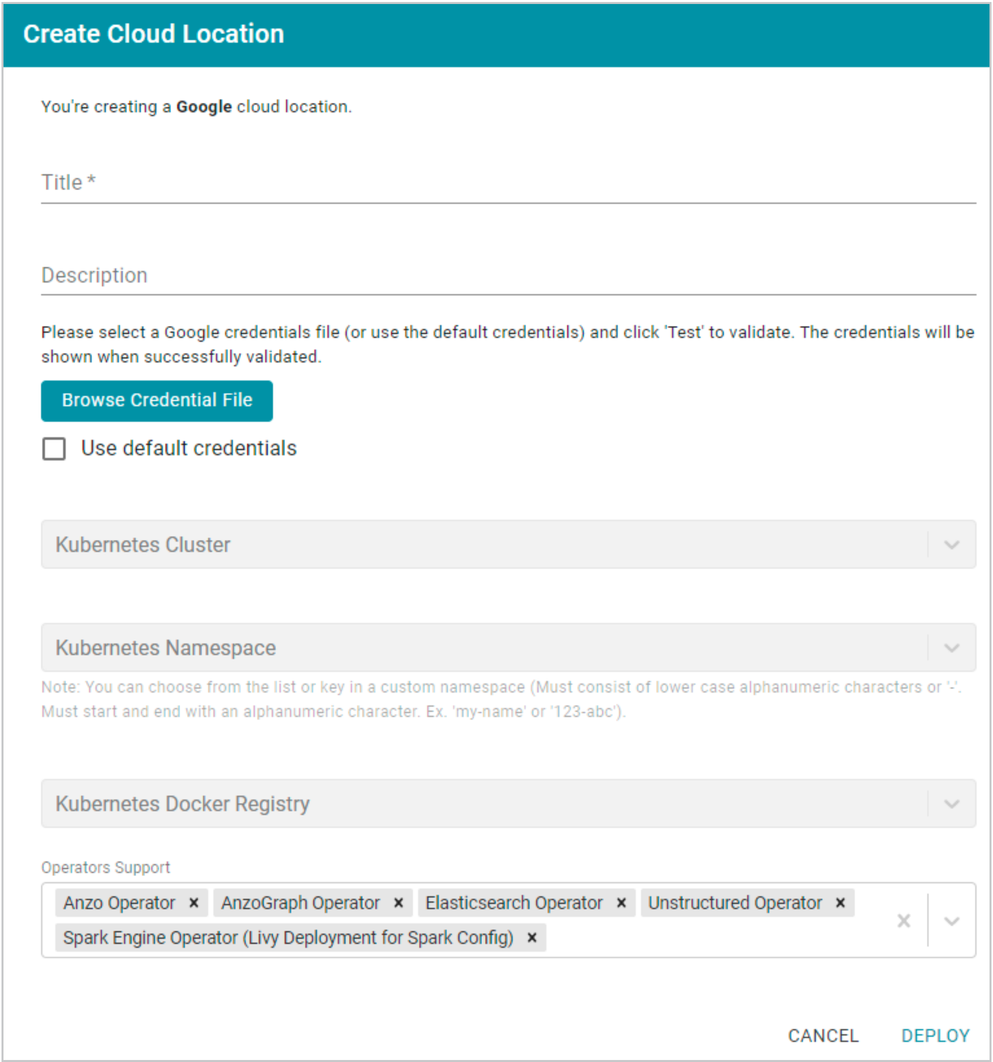
```
/opt/Anzo/Client/anzo import nfs-config.trig -u sysadmin --useModes
```

When the NFS configuration details have been imported to Anzo, see Creating a Cloud Location for next steps.

Creating a Cloud Location

Follow the instructions below to create a Cloud Location. Note that the steps below are in progress and more details are forthcoming.

1. In the Administration application, expand the **Connections** menu and click **Cloud Locations**.
2. On the Cloud Locations screen, click the **Add Cloud Location** button and select the Cloud Service Provider that hosts your Kubernetes (K8s) cluster. The Create Cloud Location dialog box is displayed. For example, the image below shows the Create Cloud Location screen for Google:



Create Cloud Location

You're creating a **Google** cloud location.

Title *

Description

Please select a Google credentials file (or use the default credentials) and click 'Test' to validate. The credentials will be shown when successfully validated.

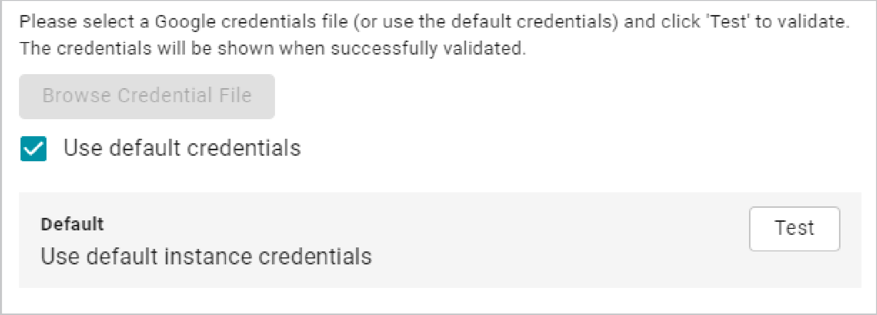
Note: You can choose from the list or key in a custom namespace (Must consist of lower case alphanumeric characters or '-'. Must start and end with an alphanumeric character. Ex. 'my-name' or '123-abc').

Operators Support: Anzo Operator ×, AnzoGraph Operator ×, Elasticsearch Operator ×, Unstructured Operator ×, Spark Engine Operator (Livy Deployment for Spark Config) ×

3. At the top of the screen, specify a **Title** for this Cloud Location and type an optional **Description**.
4. Next, specify the credentials that have permission to connect to the Cloud Service Provider (CSP) API and deploy resources in the Kubernetes cluster. There are two options, depending on the user that is

running Anzo and the Service Account, Principal, or Group that was assigned the K8s Cluster Developer IAM policy when the K8s infrastructure was set up:

- Since the Anzo Service Account, Principal, or Group is typically running Anzo, and the K8s Cluster Developer IAM policy was assigned to that account when the K8s infrastructure was set up, the appropriate credentials are already applied to this Anzo instance. In this case, select the **Use Default Credentials** checkbox. The dialog box indicates that the default instance credentials will be used and presents a **Test** button (shown in the image below).

  Please select a Google credentials file (or use the default credentials) and click 'Test' to validate. The credentials will be shown when successfully validated.

  ☑ Use default credentials

  Default
  Use default instance credentials

  Click **Test** to retrieve the credentials and test that they are valid.

- If another user is running Anzo, and that account does not have the Cluster Developer IAM permissions, retrieve from your CSP the JSON configuration file for the account that is assigned the Cluster Developer IAM policy. Then click the **Browse Credential File** button and upload the JSON credentials file that you downloaded.

Administration Tools

Anzo's Administration Tools aid administrators in performing repetitive, bulk operations.

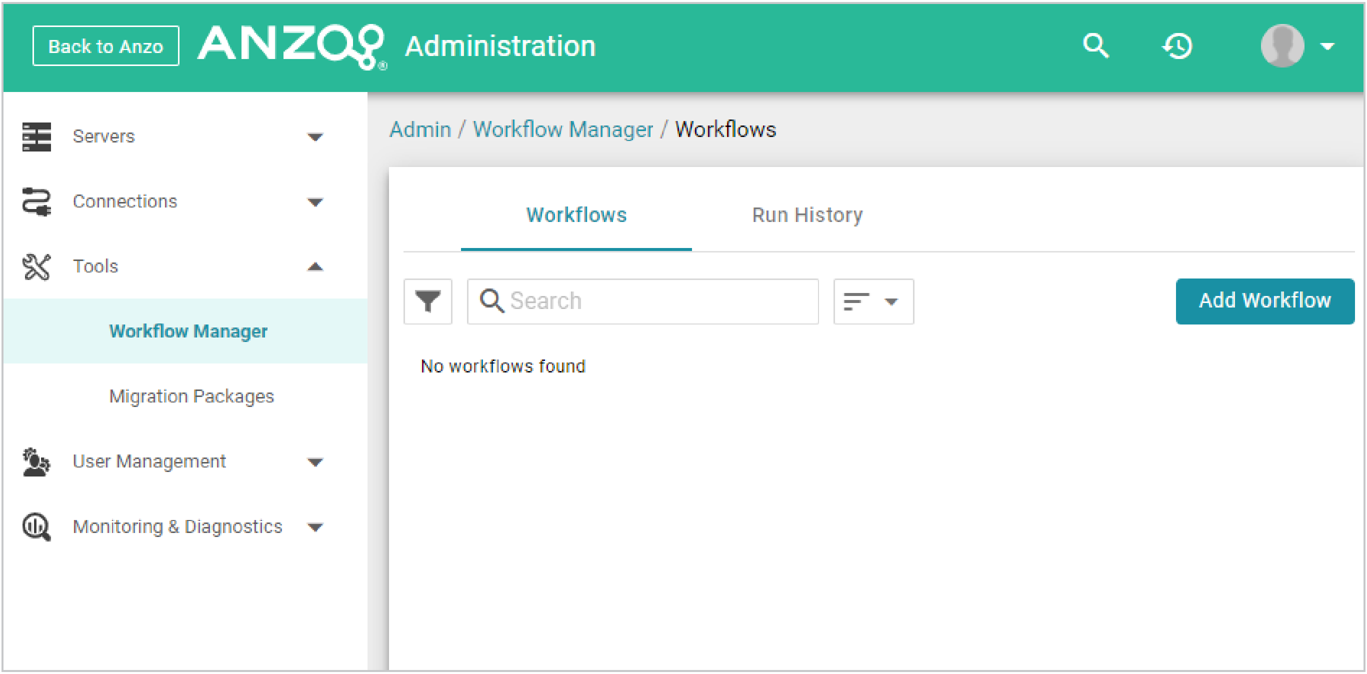
Workflow Manager

The Workflow Manager is used to manage tasks such as Structured or Unstructured Pipeline runs and Graphmart loads. Workflows can be triggered from the Anzo Admin CLI, and the CLI call can be automated by setting up cron jobs. The topics in this section provide instructions for creating Workflows, adding Tasks, and configuring cron jobs.

Adding a Workflow

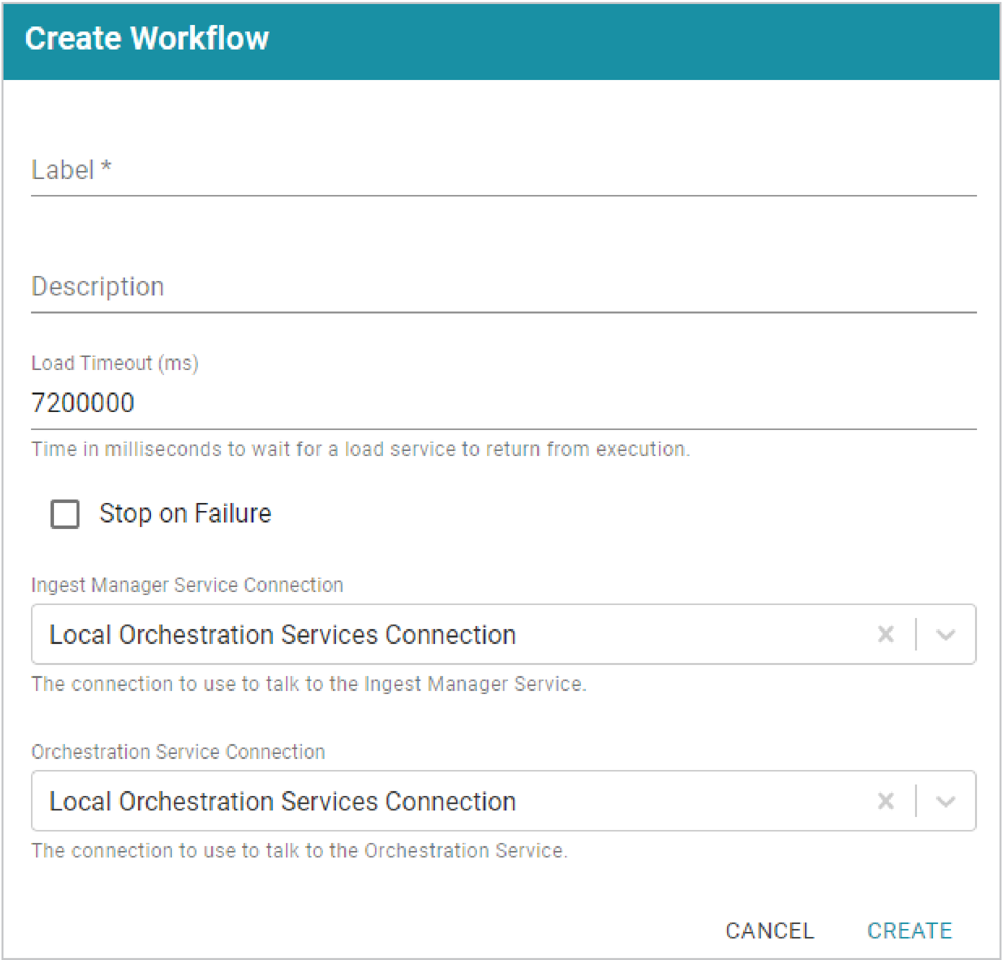
A Workflow is a container for tasks. Running a workflow runs all of the tasks in that workflow. Consider whether any tasks have dependencies when determining the number and type of tasks to group in one workflow.

1. In the Administration application, expand the **Tools** menu and click **Workflow Manager**. Anzo displays the Workflows screen, which lists any existing workflows. The image below shows the Workflows screen on an environment without any existing workflows.



No workflows found

2. Click **Add Workflow**. The Create Workflow dialog box is displayed:



Create Workflow

Label *

Description

Load Timeout (ms)

7200000

Time in milliseconds to wait for a load service to return from execution.

□ Stop on Failure

Ingest Manager Service Connection

Local Orchestration Services Connection

The connection to use to talk to the Ingest Manager Service.

Orchestration Service Connection

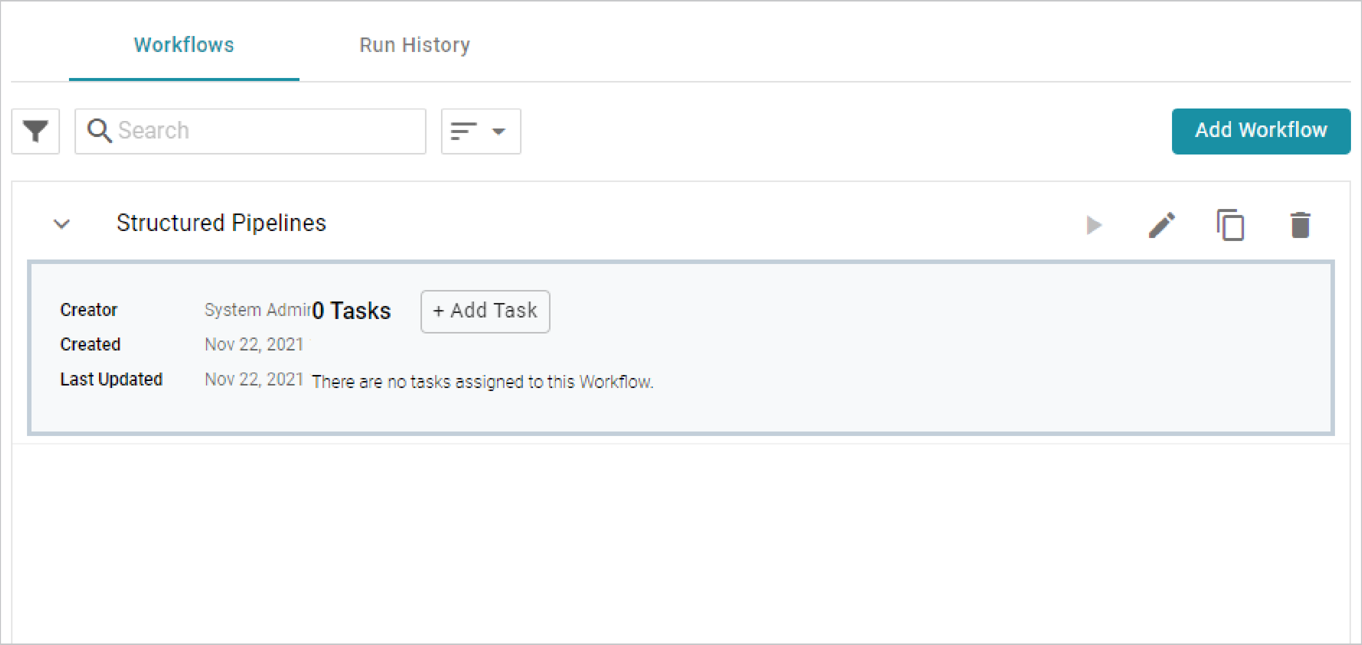
Local Orchestration Services Connection

The connection to use to talk to the Orchestration Service.

3. Configure the workflow by completing the following fields as needed. Only **Label** is a required field.
   - **Label**: This field specifies the name of the workflow.
   - **Description**: This field specifies an optional description for the workflow.
   - **Load Timeout (ms)**: This field specifies the time limit (in milliseconds) for the workflow to complete. The default value is 7200000 milliseconds (120 minutes). If all of the tasks in the workflow are not finished before the load timeout, the workflow will be stopped.
   - **Stop on Failure**: This option controls whether the workflow is stopped if one of the tasks fails or whether the workflow continues to process the rest of the tasks if there is a failure.
   - **Ingest Manager Service Connection**: This field specifies the Ingest Manager connection to use for this Workflow. The field defaults to the local Orchestration Services connection. If you have registered additional connections, you can select an alternate connection.

- **Orchestration Service Connection**: This field specifies the Orchestration Service connection to use for this workflow. The field defaults to the local Orchestration Services connection. If you have registered additional connections, you can select an alternate connection.

4. Click **Create** to add the workflow. The new workflow is added to the list of workflows on the Workflows screen. For example, the image below shows that there is one new workflow without any tasks.



Structured Pipelines

Creator System Admir

Created Nov 22, 2021

Last Updated Nov 22, 2021

0 Tasks

There are no tasks assigned to this Workflow.

Once the workflow is configured, you can add any number of tasks that the workflow should run. For instructions, see Adding a Task to a Workflow.

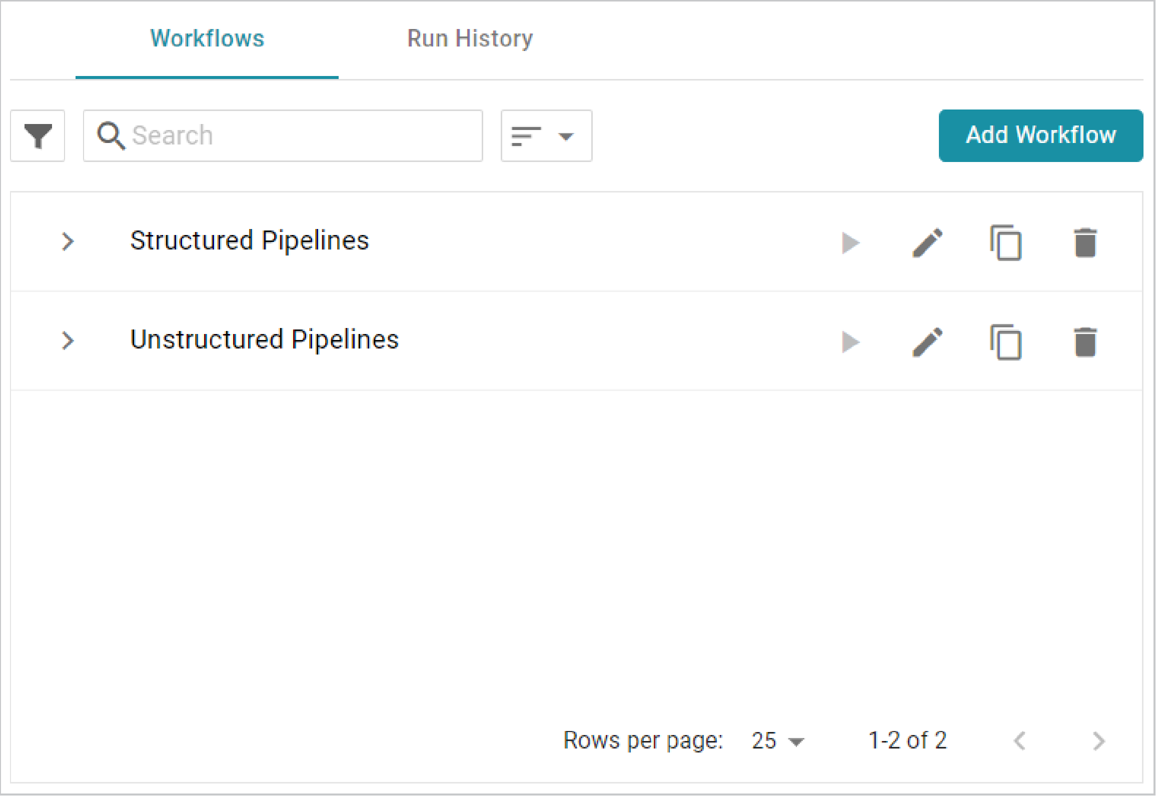
Adding a Task to a Workflow

The topics in this section provide instructions for configuring each type of task that is available for adding to a workflow.

Adding a Task that Runs an Unstructured Pipeline

Follow the instructions below to add a task that runs an unstructured pipeline.

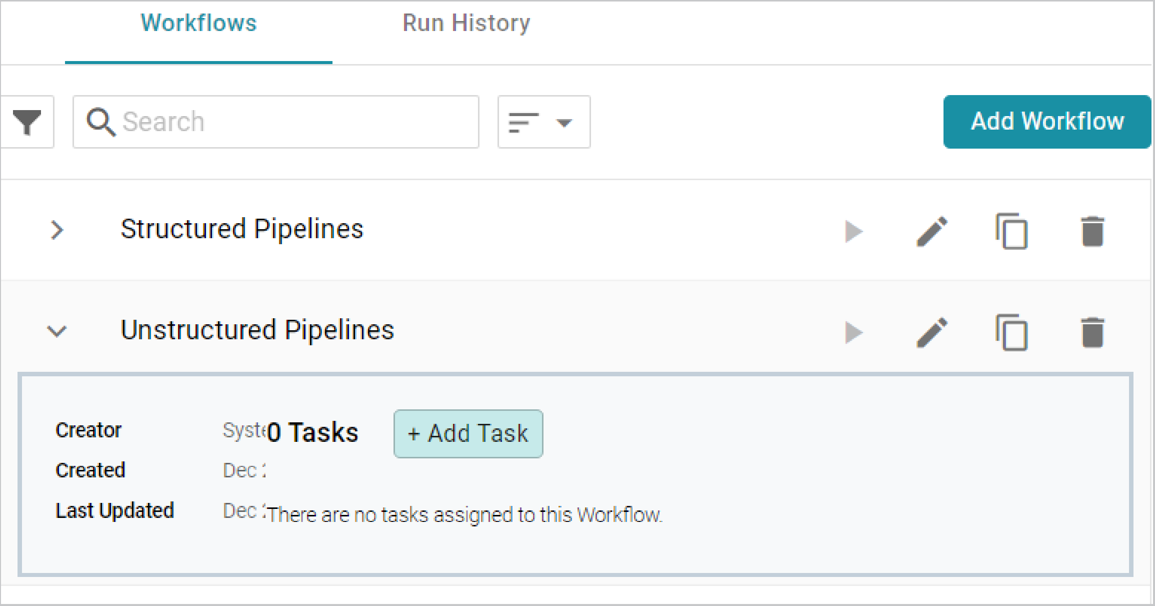
1. In the Administration application, expand the **Tools** menu and click **Workflow Manager**. Anzo displays the Workflows screen, which lists any existing workflows. For example:



Structured Pipelines

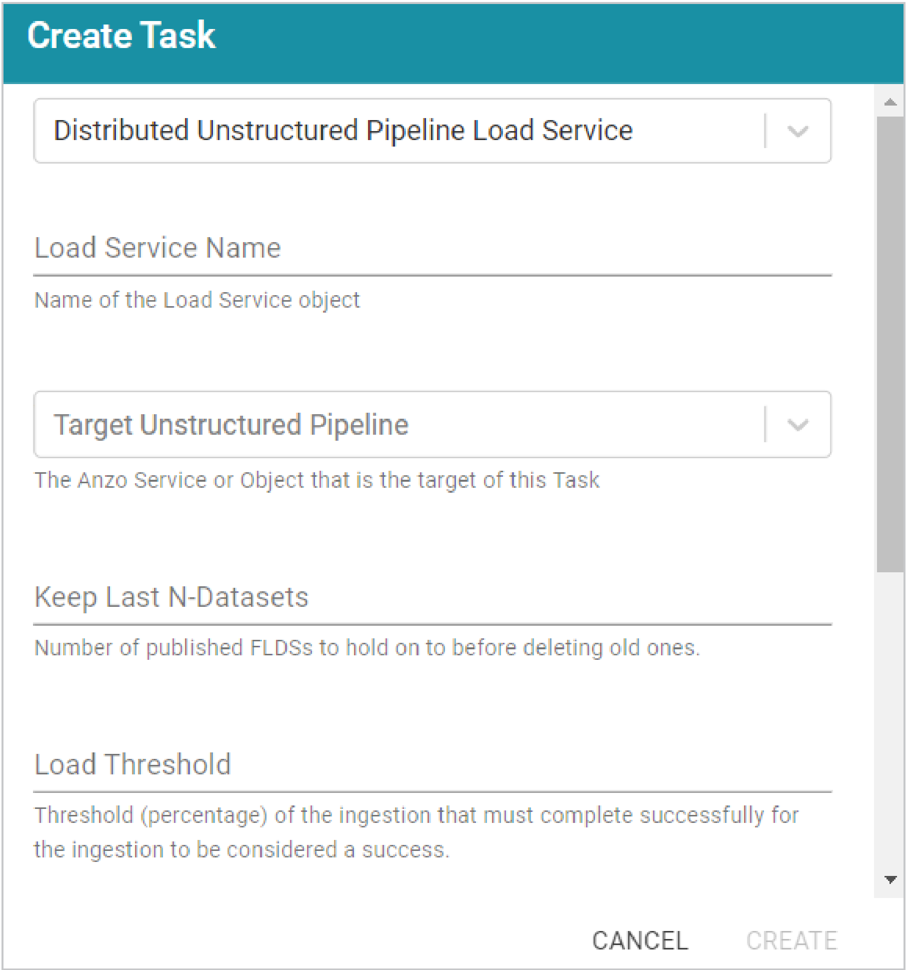
Unstructured Pipelines

2. Expand the workflow that you want to add a task to. For example:



Creator Syst0 Tasks
Created Dec :
Last Updated Dec :There are no tasks assigned to this Workflow.

3. Click **Add Task**. The Create Task dialog box is displayed:



Distributed Unstructured Pipeline Load Service

Load Service Name

Name of the Load Service object

Target Unstructured Pipeline

The Anzo Service or Object that is the target of this Task

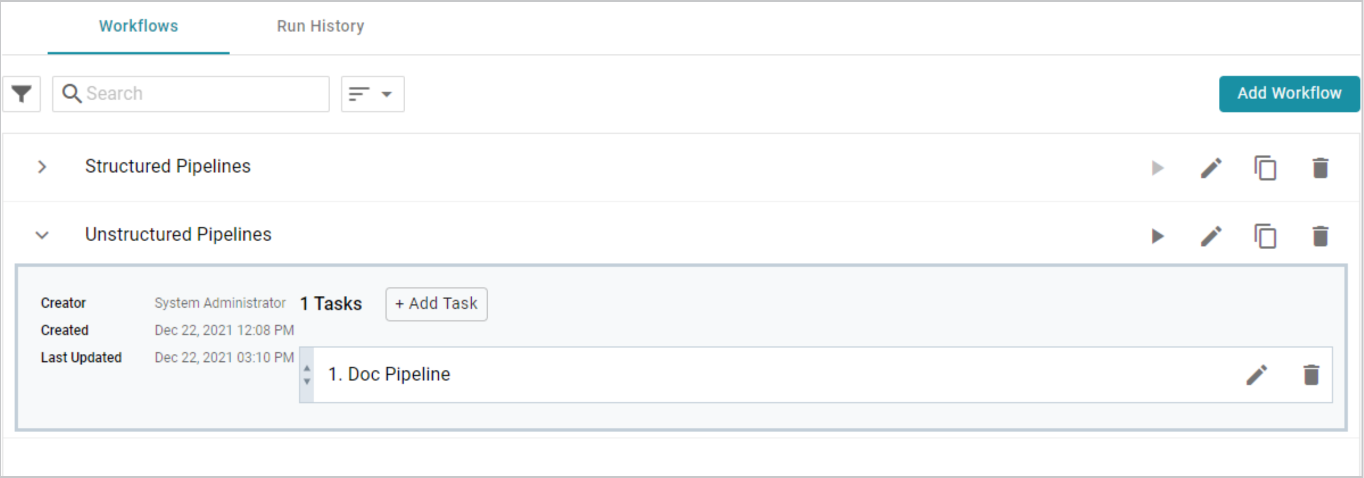
Keep Last N-Datasets

Number of published FLDSs to hold on to before deleting old ones.

Load Threshold

Threshold (percentage) of the ingestion that must complete successfully for the ingestion to be considered a success.

4. Configure the Task by completing the following fields as needed:
    - **Task Type**: The drop-down list at the top of the dialog box specifies the type of task to create. **Distributed Unstructured Pipeline Load Service** is selected by default. Accept the default value.
    - **Load Service Name**: This field specifies the name for the task.
    - **Target Unstructured Pipeline**: This field specifies the unstructured pipeline that this task should run. Click the drop-down list and select the desired pipeline.
    - **Keep Last N-Datasets**: This field specifies the number of file-based linked data sets (FLDS) from this pipeline to retain on disk before deleting the oldest ones.
    - **Load Threshold**: This field specifies the percentage of the pipeline that must complete successfully for the ingestion to be considered a success.
    - **Distributed Unstructured Pipeline Stop Timeout**: This field specifies the number of milliseconds to wait for an unstructured pipeline to stop.
    - **Distributed Unstructured Pipeline Percent Timeout**: This field specifies the number of milliseconds to wait before timing out if there is no change in the percentage of documents processed.
    - **Index**: This field specifies a numeric value that represents the order in which this task should run in the workflow.
5. Click **Create** to add the task to the workflow. For example, the image below shows a workflow with one task.



Creator System Administrator
Created Dec 22, 2021 12:08 PM
Last Updated Dec 22, 2021 03:10 PM

1 Tasks

1. Doc Pipeline

You can repeat this process to add tasks that run additional unstructured pipelines.

Related Topics

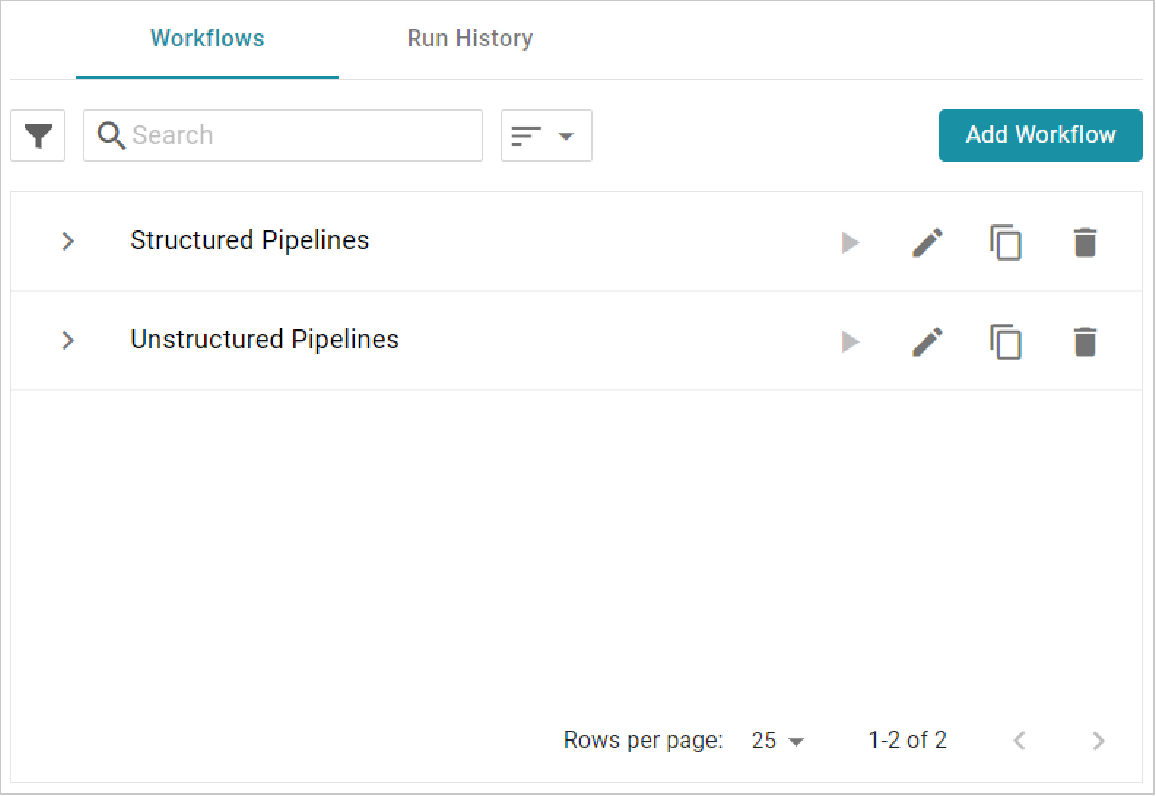
Adding a Task that Runs a Job in a Structured Pipeline

Adding a Task that Refreshes or Reloads a Graphmart

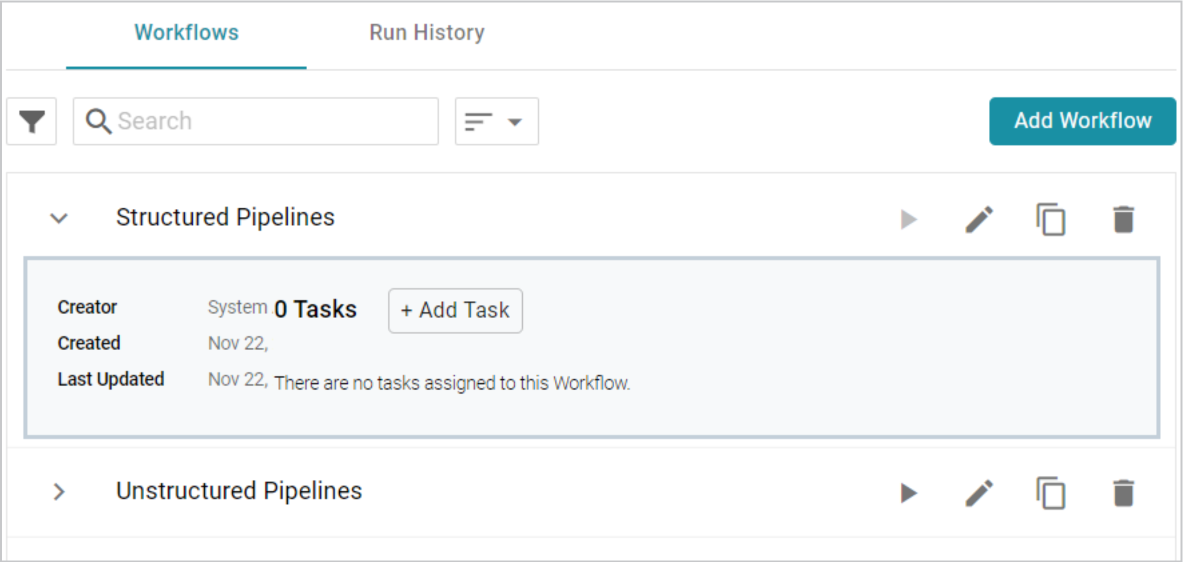
Adding a Task that Runs a Job in a Structured Pipeline

Follow the instructions below to add a task that runs a job in a structured pipeline.

1. In the Administration application, expand the **Tools** menu and click **Workflow Manager**. Anzo displays the Workflows screen, which lists any existing workflows. For example:

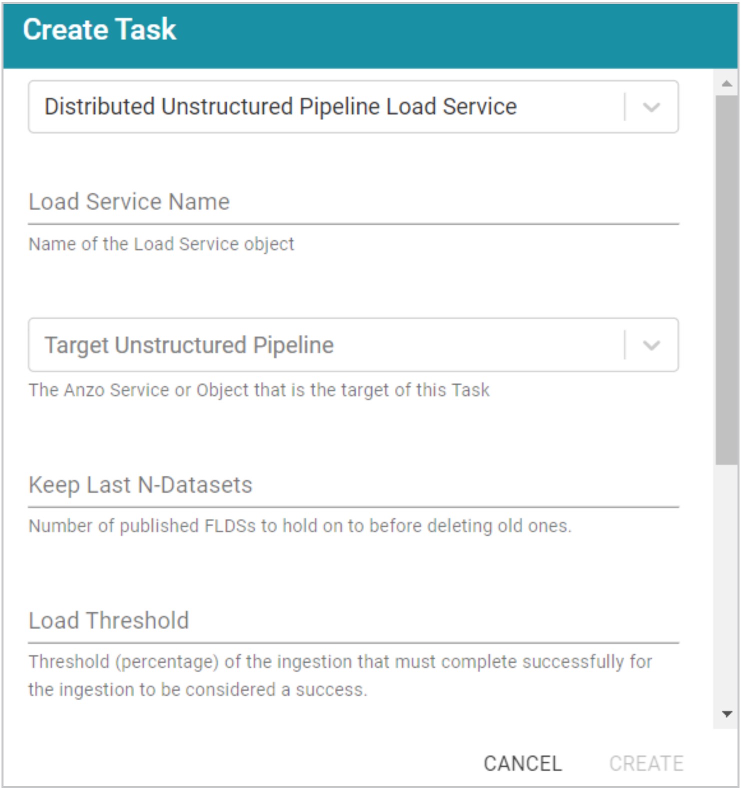


2. Expand the workflow that you want to add a task to. For example:



Creator	System	0 Tasks
Created	Nov 22,	
Last Updated	Nov 22,	There are no tasks assigned to this Workflow.

3. Click **Add Task**. The Create Task dialog box is displayed:


Create Task

Distributed Unstructured Pipeline Load Service

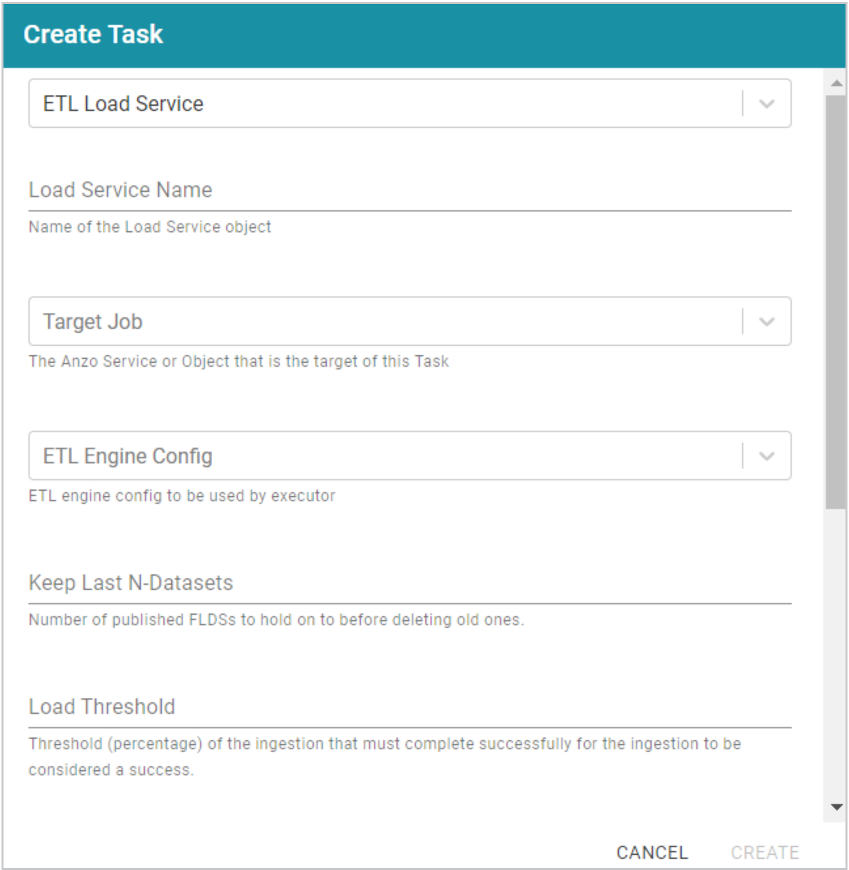
Load Service Name
Name of the Load Service object

Target Unstructured Pipeline
The Anzo Service or Object that is the target of this Task

Keep Last N-Datasets
Number of published FLDSs to hold on to before deleting old ones.

Load Threshold
Threshold (percentage) of the ingestion that must complete successfully for the ingestion to be considered a success.

4. At the top of the dialog box, click the drop-down list and select **ETL Load Service** to set up a task that runs a structured job. The dialog box presents the options that are valid for ETL Load Service Tasks:



Create Task

ETL Load Service

Load Service Name

Name of the Load Service object

Target Job

The Anzo Service or Object that is the target of this Task

ETL Engine Config

ETL engine config to be used by executor

Keep Last N-Datasets

Number of published FLDSs to hold on to before deleting old ones.

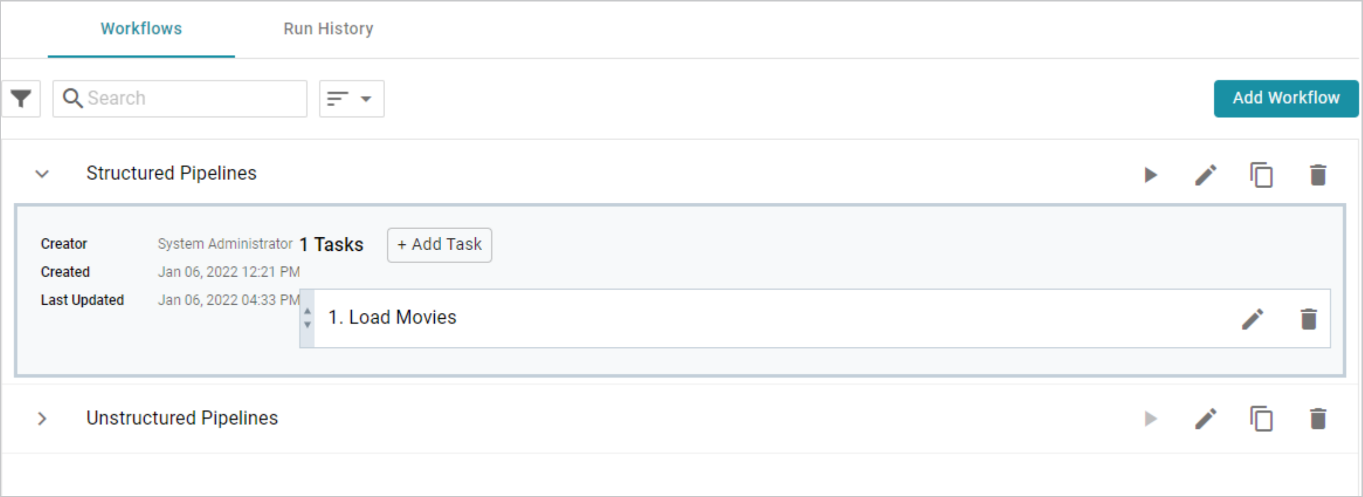
Load Threshold

Threshold (percentage) of the ingestion that must complete successfully for the ingestion to be considered a success.

5. Configure the task by completing the following fields as needed:
    - **Load Service Name**: This field specifies the name for the task.
    - **Target Job**: This field specifies the job that this task should run. Click the drop-down list and select the desired job.
    - **ETL Engine Config**: This field specifies the ETL engine to use for publishing the job.
    - **Keep Last N-Datasets**: This field specifies the number of file-based linked data sets (FLDS) from this pipeline to retain on disk before deleting the oldest ones.
    - **Load Threshold**: This field specifies the percentage of the job that must complete successfully for the ingestion to be considered a success.
    - **Job Execution Timeout**: This field specifies the number of milliseconds to wait for job executions to run to completion.
    - **Number Status Checks**: This field specifies the number of times to perform a status check on the job run.
    - **Wait Time Between Status Checks**: This field specifies the number of milliseconds to wait between status checks.

- **Include Preceding Stages**: This field indicates whether to run the all of the stages in the publishing process (generate, compile, deploy, and run) or whether to complete the run step only. Selecting **Include Preceding Stages** runs all of the steps,
- **Index**: This field specifies a numeric value that represents the order in which this task should run in the workflow.

6. Click **Create** to add the task to the workflow. For example, the image below shows a workflow with one Task.



You can repeat this process to add tasks that run additional jobs.

Related Topics

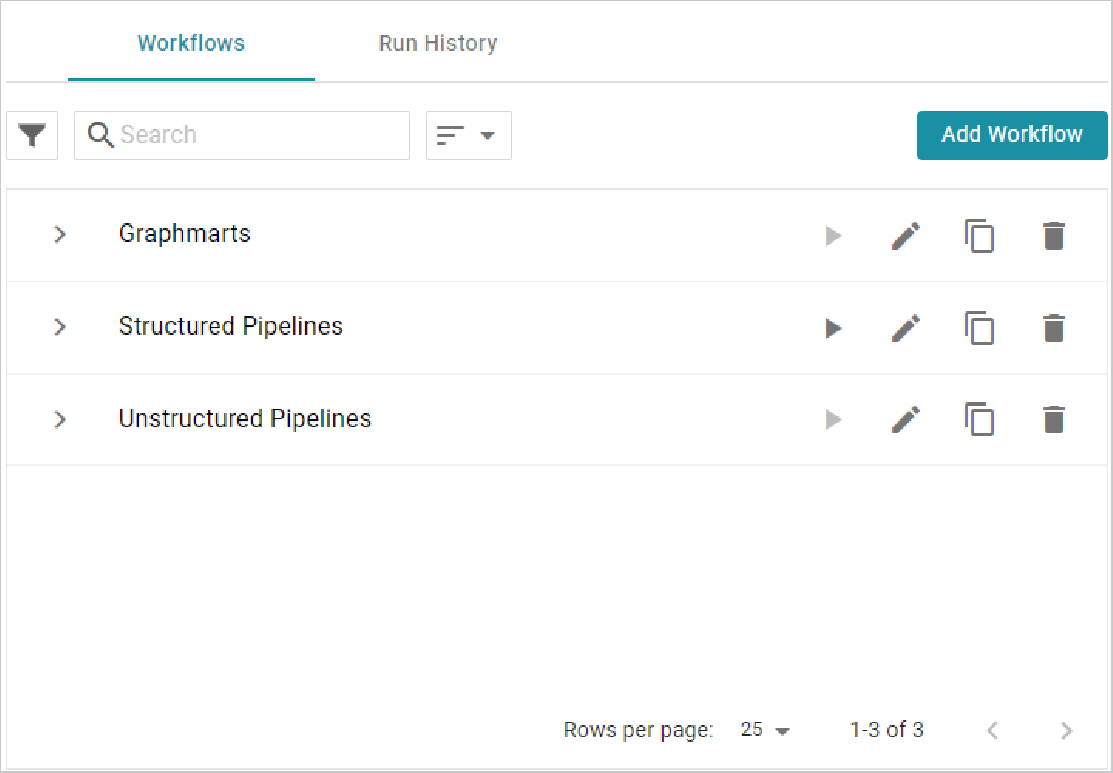
Adding a Task that Runs an Unstructured Pipeline

Adding a Task that Refreshes or Reloads a Graphmart

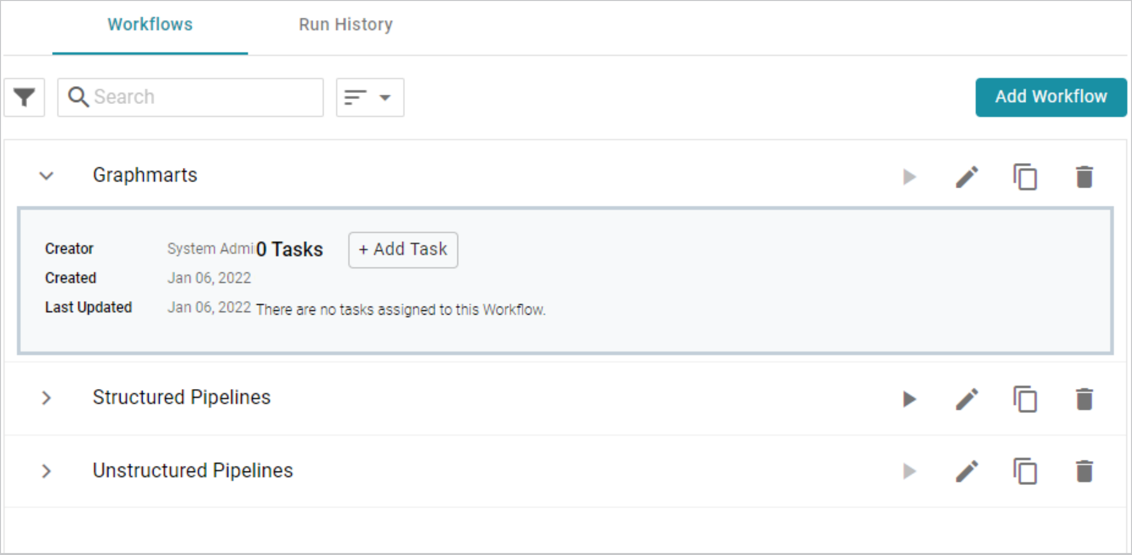
Adding a Task that Refreshes or Reloads a Graphmart

Follow the instructions below to add a task that refreshes or reloads a graphmart.

1. In the Administration application, expand the **Tools** menu and click **Workflow Manager**. Anzo displays the Workflows screen, which lists any existing workflows. For example:



2. Expand the workflow that you want to add a task to. For example:

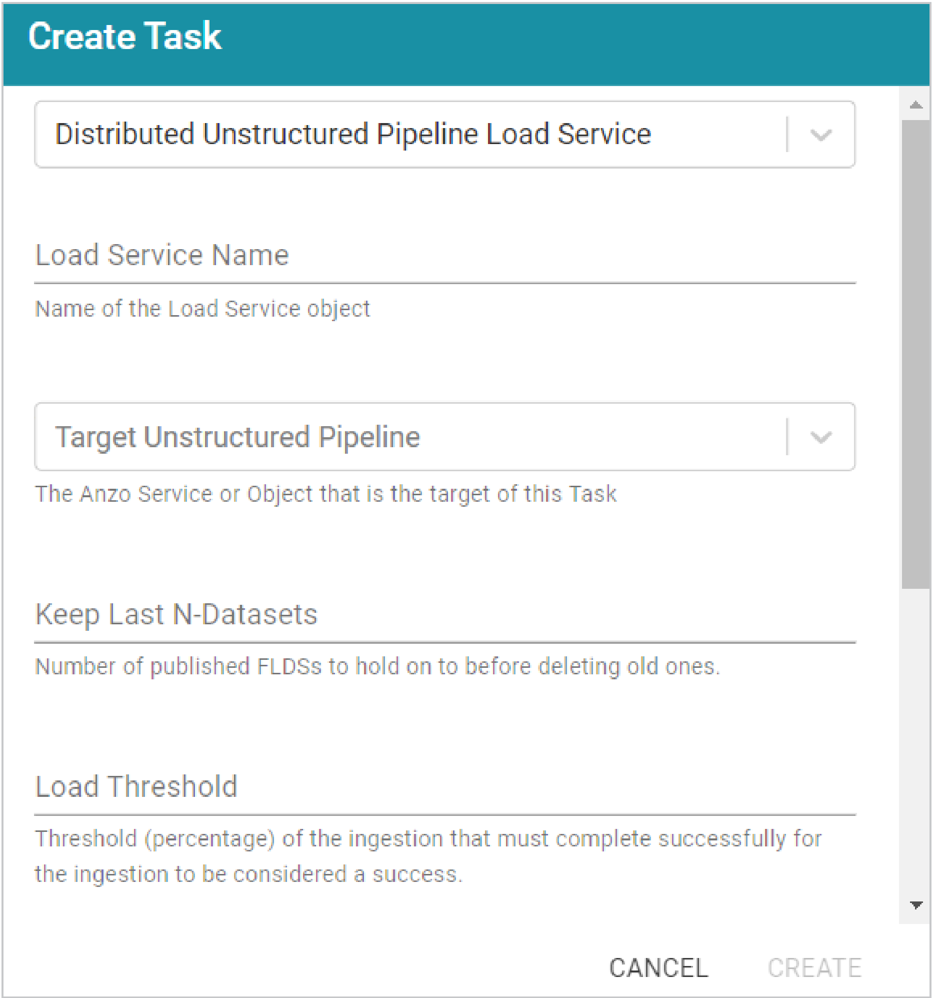


Creator System Admi 0 Tasks + Add Task

Created Jan 06, 2022

Last Updated Jan 06, 2022 There are no tasks assigned to this Workflow.

3. Click **Add Task**. The Create Task dialog box is displayed:



Create Task

Distributed Unstructured Pipeline Load Service

Load Service Name

Name of the Load Service object

Target Unstructured Pipeline

The Anzo Service or Object that is the target of this Task

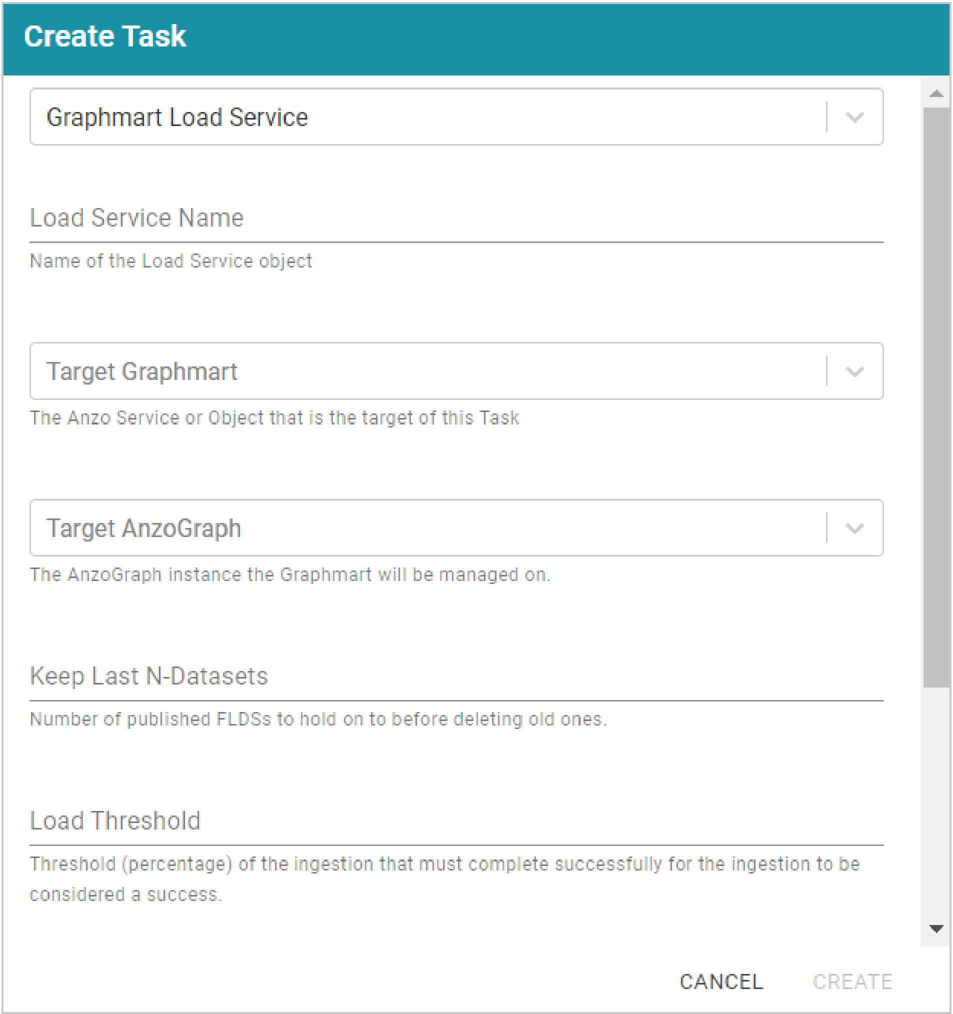
Keep Last N-Datasets

Number of published FLDSs to hold on to before deleting old ones.

Load Threshold

Threshold (percentage) of the ingestion that must complete successfully for the ingestion to be considered a success.

4. At the top of the dialog box, click the drop-down list and select **Graphmart Load Service** to set up a task that reloads or refreshes a graphmart. The dialog box presents the options that are valid for Graphmart Load Service Tasks:



Create Task

Graphmart Load Service

Load Service Name

Name of the Load Service object

Target Graphmart

The Anzo Service or Object that is the target of this Task

Target AnzoGraph

The AnzoGraph instance the Graphmart will be managed on.

Keep Last N-Datasets

Number of published FLDSs to hold on to before deleting old ones.

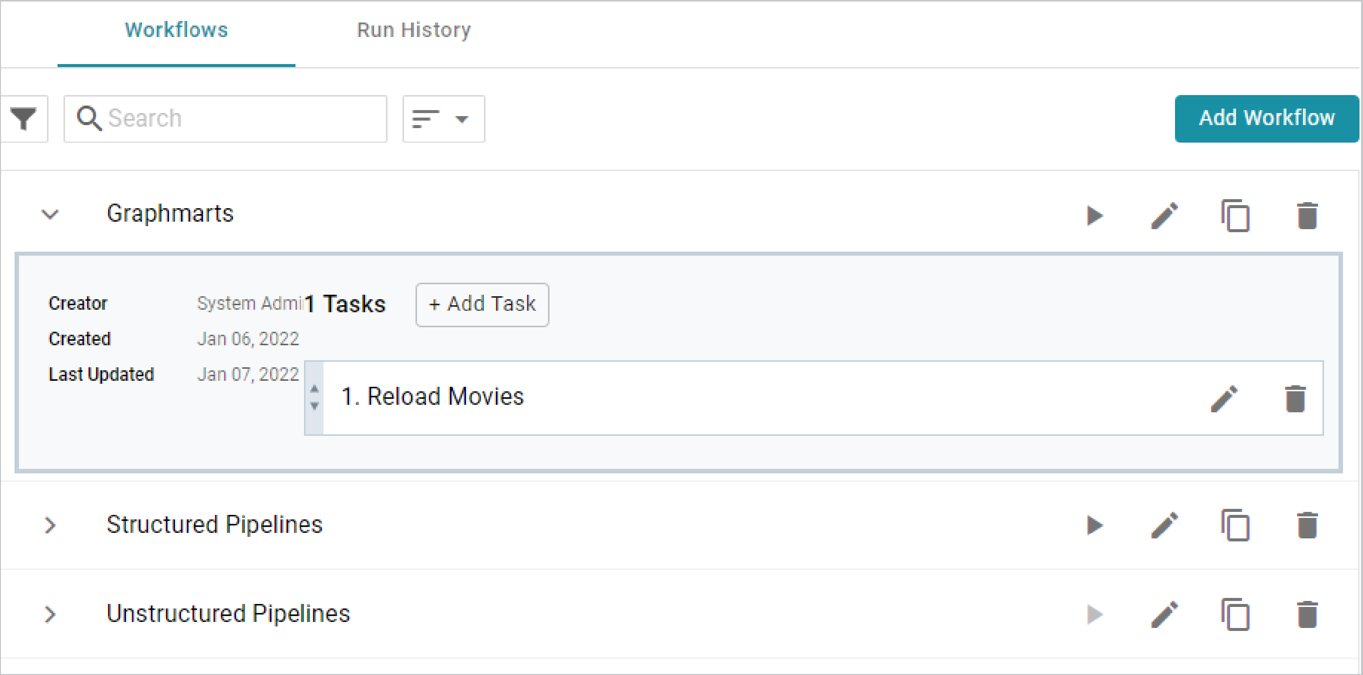
Load Threshold

Threshold (percentage) of the ingestion that must complete successfully for the ingestion to be considered a success.

5. Configure the task by completing the following fields as needed:
   - **Load Service Name**: This field specifies the name for the task.
   - **Target Graphmart**: This field specifies the graphmart that this task should reload or refresh. Click the drop-down list and select the desired graphmart.
   - **Target AnzoGraph**: This field specifies the AnzoGraph instance that hosts this graphmart.
   - **Keep Last N-Datasets**: This field is not relevant for Graphmart Load Service Tasks.
   - **Load Threshold**: This field is not relevant for Graphmart Load Service Tasks.
   - **Graphmart Action**: This field specifies whether to refresh or reload the target graphmart. For refresh, click the drop-down list and select **Refresh Target Graphmart**. To perform a reload, click the drop-down list and select **Reload Target Graphmart**.
   - **Activate**: This option indicates whether the target graphmart needs to be activated before the refresh or reload is attempted. If the target graphmart is offline when the workflow is run, this task will fail unless **Activate** is enabled.

- **Deactivate**: This option indicates whether to deactivate the graphmart after the task is complete. If you want Anzo to deactivate the target graphmart after the reload or refresh is complete, select the **Deactivate** checkbox.
- **Index**: This field specifies a numeric value that represents the order in which this task should run in the workflow.

6. Click **Create** to add the task to the workflow. For example, the image below shows a workflow with one task.



You can repeat this process to add tasks that refresh or reload additional graphmarts.

Related Topics

Adding a Task that Runs an Unstructured Pipeline

Adding a Task that Runs a Job in a Structured Pipeline

Running a Workflow

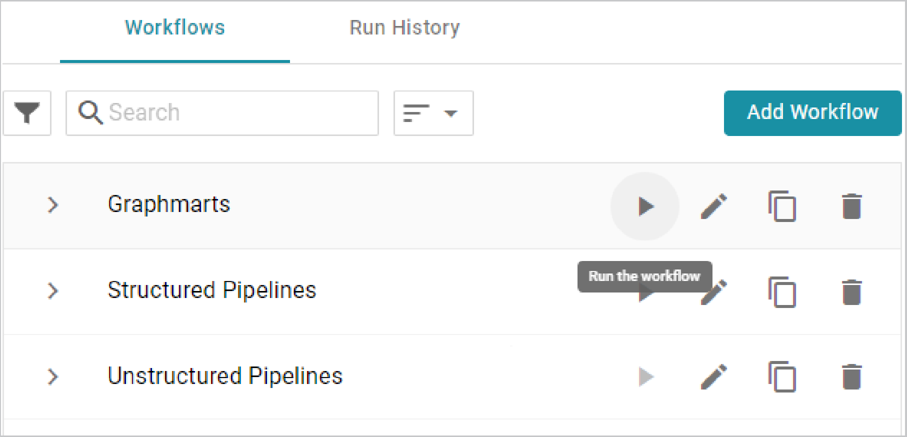
There are multiple ways to run workflows. You can initiate a workflow manually from the Administration application or the Anzo Admin CLI. You can also automate workflows by using the Linux Cron utility or a similar application to schedule them. This topic provides instructions for running a workflow manually and gives an example of a cron job that runs a workflow on a schedule.

- Running a Workflow Manually
- Scheduling a Workflow to Run Automatically

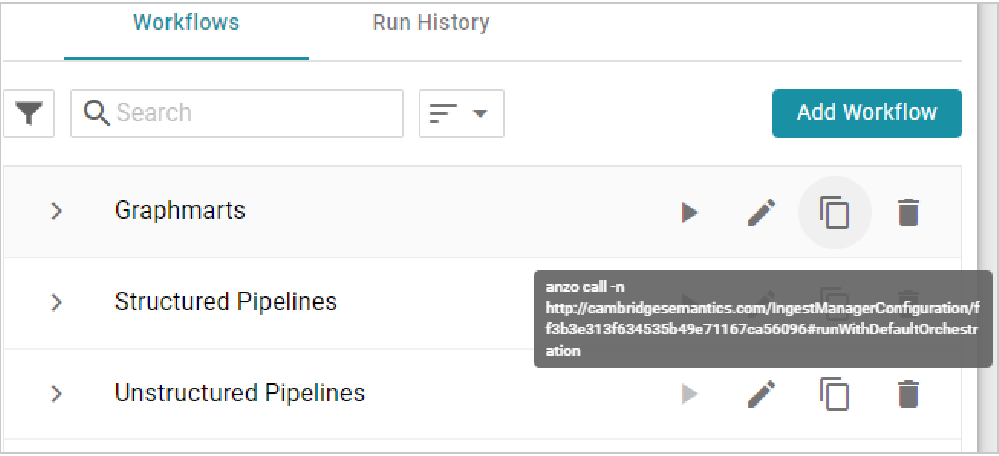
Running a Workflow Manually

There are two ways to run a workflow manually:

1. You can click the run icon (▶) for the workflow in the Administration application.



2. You can click the copy icon (⧉) to copy the `anzo call` statement for the workflow and run it with the Admin CLI.

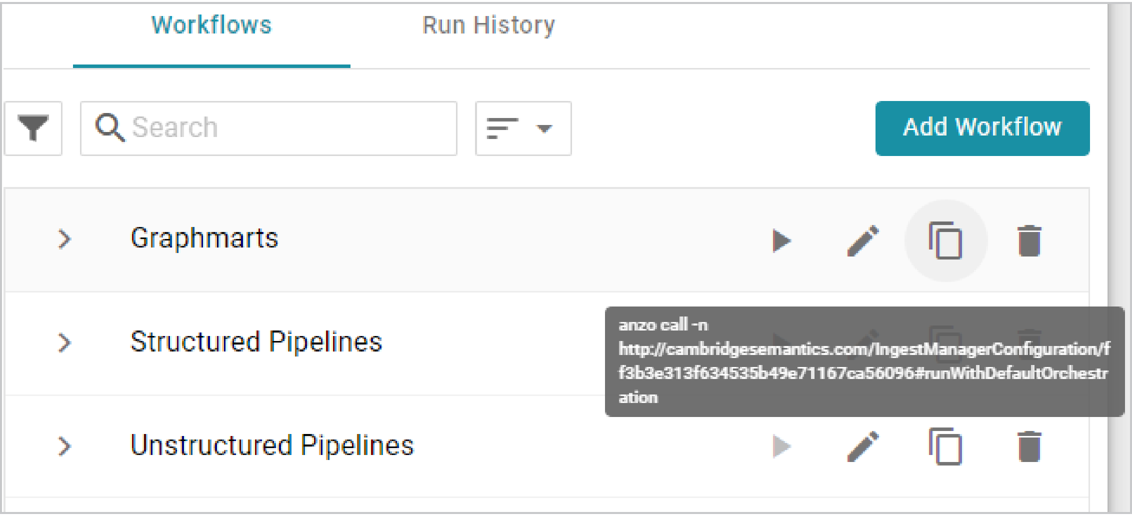


```
anzo call -n
http://cambridgesemantics.com/IngestManagerConfiguration/f
f3b3e313f634535b49e71167ca56096#runWithDefaultOrchestr
ation
```

Scheduling a Workflow to Run Automatically

To automate the running of a workflow, you can set up a cron job that runs the `anzo call` statement on a schedule. This section gives example steps to follow to set up a cron job that schedules a single workflow.

1. First, find the `anzo call` statement for the workflow that you want to schedule. As shown in the image below, you can click the copy icon (⧉) for the workflow to copy the statement.

   ```
   anzo call -n
   http://cambridgesemantics.com/IngestManagerConfiguration/f
   f3b3e313f634535b49e71167ca56096#runWithDefaultOrchestr
   ation
   ```

2. On the Anzo server, run the following command to open a crontab:

   ```
   sudo crontab -e -u <user_name>
   ```

   For example, the following command opens a crontab as the Anzo service user:

   ```
   sudo crontab -e -u anzo
   ```

3. Add contents to the file using the following syntax. Use an asterisk in place of options that you do not want to set:

   ```
   <minute> <hour> <day_of_month> <month> <day_of_week> <absolute_path_to_
   client/anzo_call_statement>
   ```

   For example, the following contents run the Workflow every day at 8:00 AM:

   ```
   0 8 * * * /opt/Anzo/Client/anzo call -n
   http://cambridgesemantics.com/IngestManagerConfiguration/ff3b3e313f634535
   b49e71167ca56096#runWithDefaultOrchestration
   ```

4. Save and close the crontab.

Migration Packages

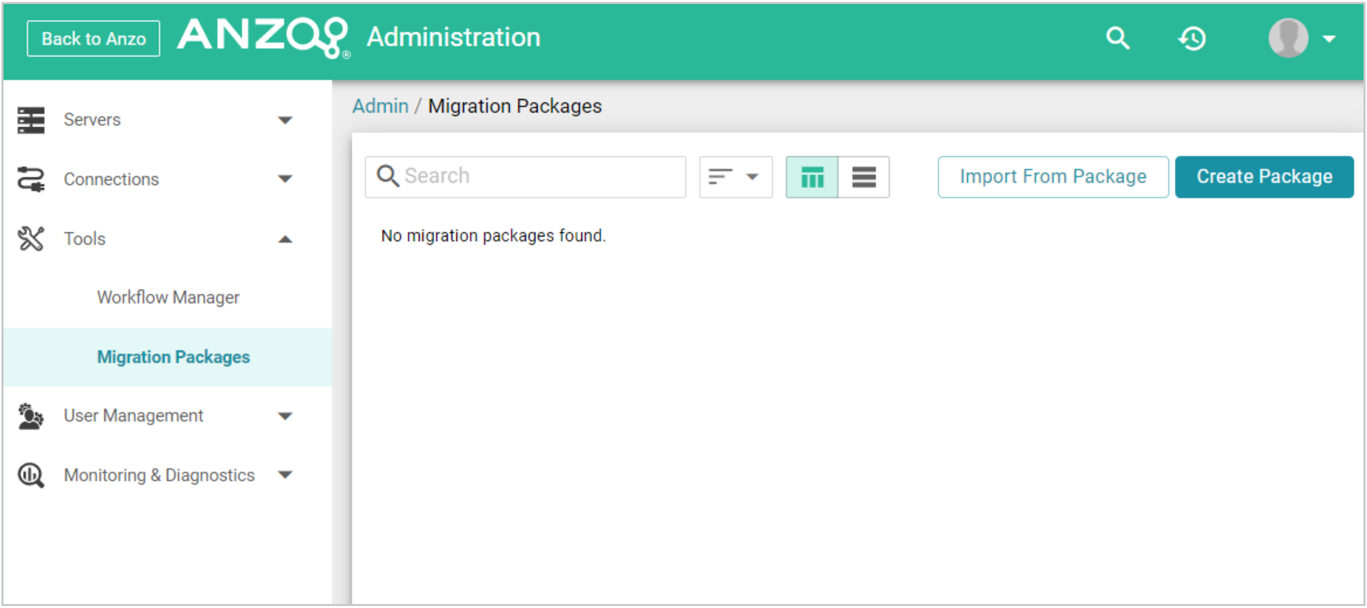
When migrating artifacts between environments, administrators can perform a bulk export (and import) by assembling a Migration Package that includes any number and type of artifacts and their related entities. The export configuration is maintained at the package level and applied to all of the contained artifacts, which means the configuration can be reused as artifacts are added to or removed from the package. The topics in this section provide instructions on creating and configuring Migration Packages as well as exporting and importing packages.

Creating a Migration Package

Follow the instructions below to create a new Migration Package, add artifacts to the package, and configure the export options.

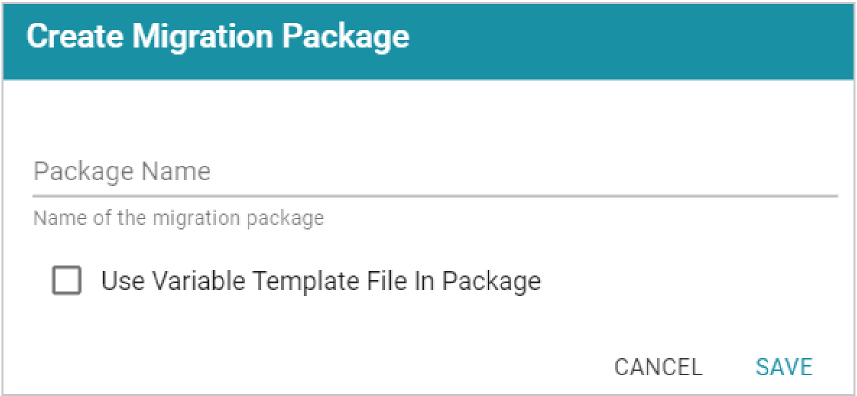
> **Note**
> There are two permissions that control access to export, modify, and import migration packages: **Manage Migration Packages** and **Perform Migration Package Operations As Sysadmin**. If a user has only the Manage Migration Packages permission, they cannot modify, export, or import artifacts in packages unless they have the appropriate permissions on the artifacts. If a user has Perform Migration Package Operations As Sysadmin, that means they can modify, export, and import migration packages that include artifacts they may not otherwise have permission to operate on.

1. In the Administration application, expand the **Tools** menu and click **Migration Packages**. Anzo displays the Migration Packages screen, which lists any existing packages. The image below shows the Migration Packages screen on an environment without any existing packages.



No migration packages found.

2. Click the **Create Package** button. The Create Migration Package dialog box is displayed.



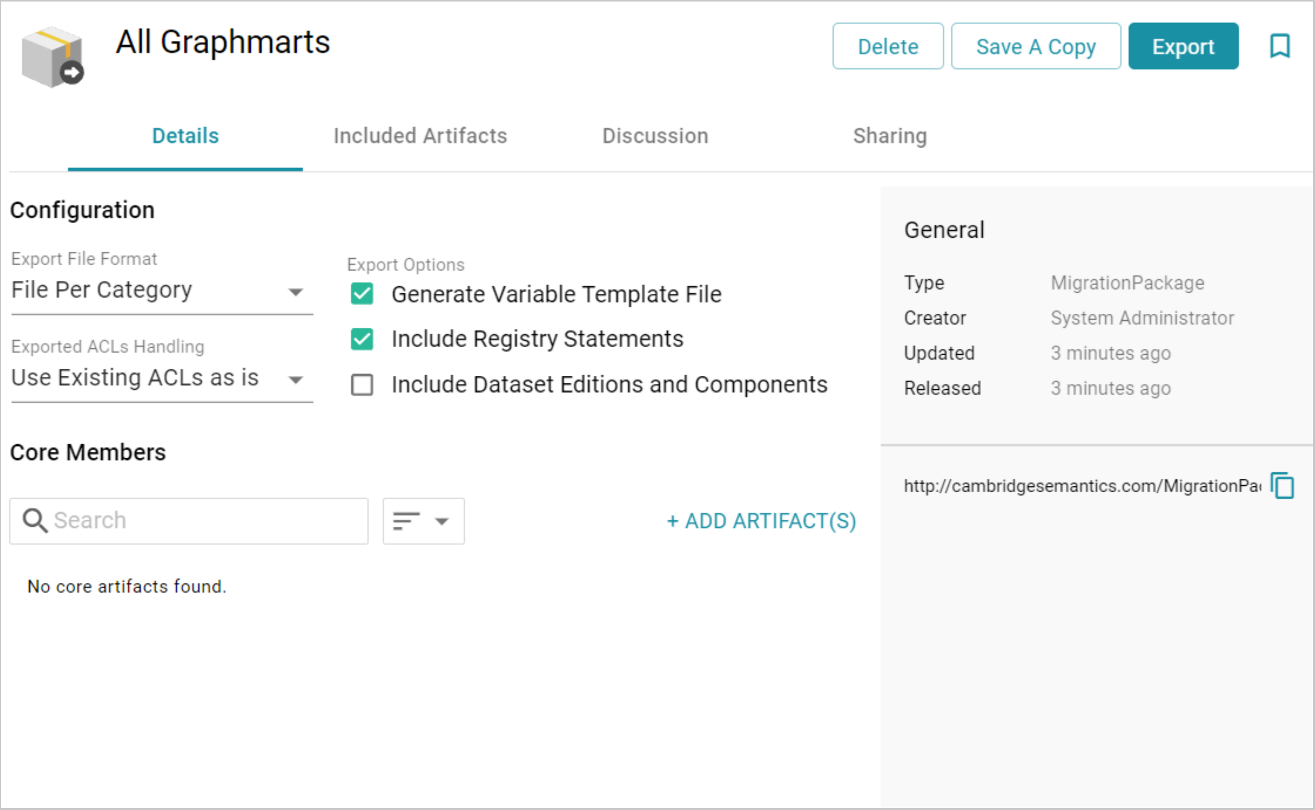
Create Migration Package

Package Name

Name of the migration package

☐ Use Variable Template File In Package

3. On the Create Migration Package screen, type a name for the package in the **Package Name** field.
4. Next, determine whether you want to add a Variable Template File to the package. A Variable Template File is a TriG file that contains statements for all of the properties that have replaceable values for each artifact included in the package. Properties with replaceable values are objects such as file paths and Anzo Data Store locations, which might differ on the source and target Anzo servers. The template that is generated has placeholder text that you replace with the desired values for the target server. To generate a Variable Template File with the package, select the **Use Variable Template File In Package** checkbox. Leave the checkbox blank if you do not want to generate the file.
5. Click **Save** to save the package. Anzo creates the package and displays the Details tab where you can add artifacts and configure additional options. For example, the image below shows a new package called All Graphmarts.


All Graphmarts

Configuration

Export File Format: File Per Category

Exported ACLs Handling: Use Existing ACLs as is

Export Options

☑ Generate Variable Template File

☑ Include Registry Statements

□ Include Dataset Editions and Components

Core Members

No core artifacts found.

General

Type: MigrationPackage

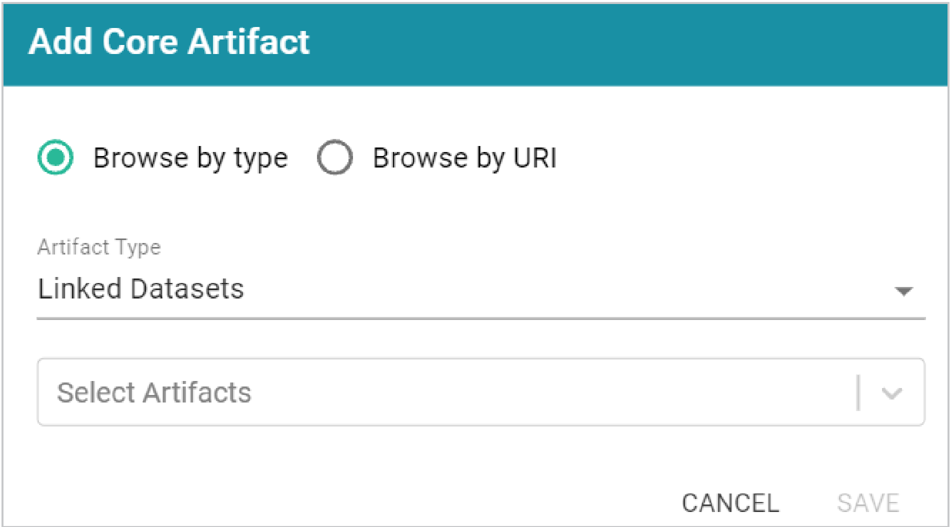
Creator: System Administrator

Updated: 3 minutes ago

Released: 3 minutes ago

http://cambridgesemantics.com/MigrationPa

6. First, determine the artifacts to add to the package. Click **Add Artifacts** under Core Members. The Add Core Artifact dialog box is displayed:


Add Core Artifact

● Browse by type ○ Browse by URI

Artifact Type: Linked Datasets

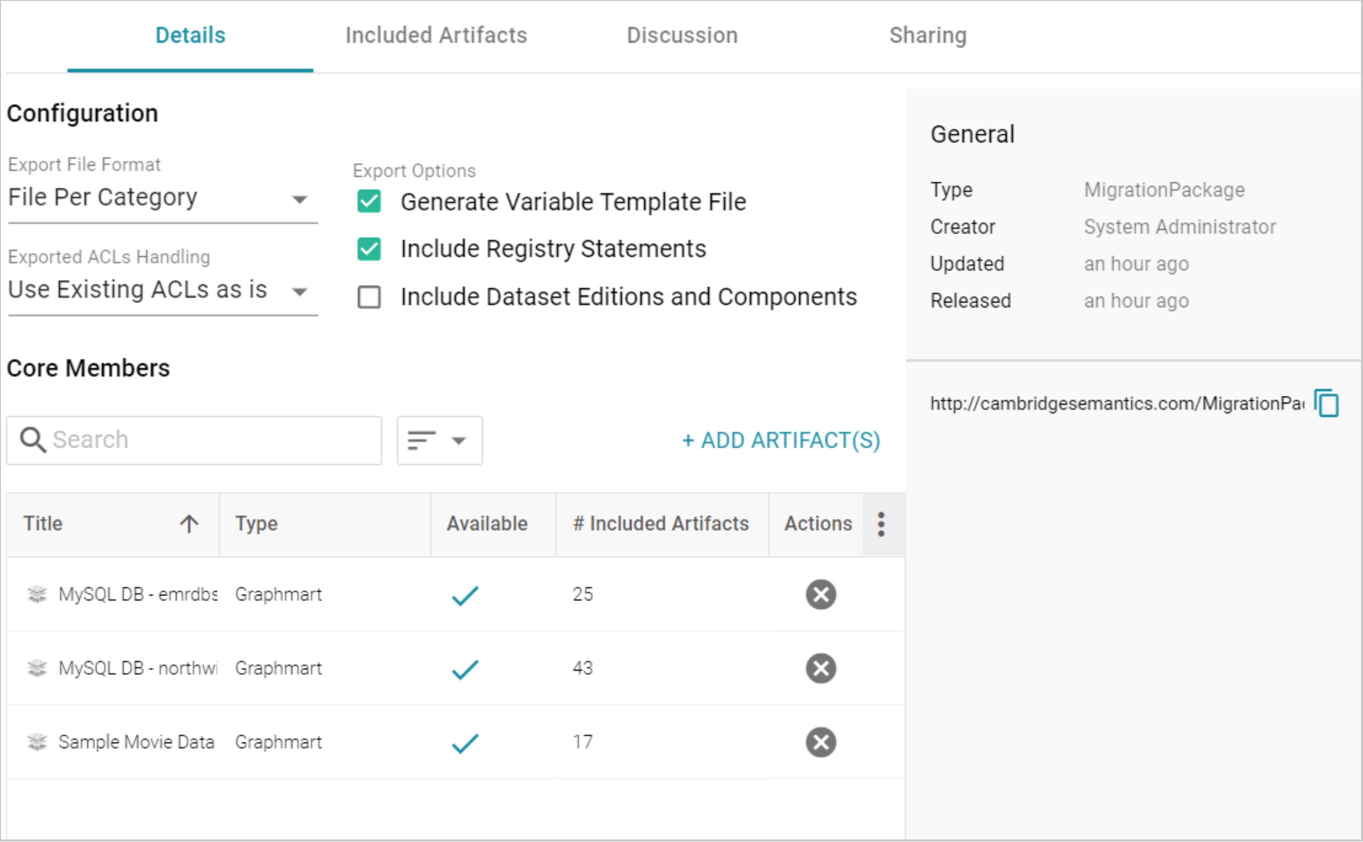
7. By default, the dialog box is set to **Browse by Type** of artifact, such as Linked Dataset, Data Source, Mapping, or Graphmart. To choose a type, click the **Artifact Type** drop-down list and select a type to filter by. The **Select Artifacts** list is filtered to show only the selected type of artifact.

> **Tip**
>
> If you have a list of artifacts that you want to find by URI, you can select the **Browse by URI** radio button and then specify a URI.

8. Click **Select Artifacts** and select an artifact from the list. Repeat this step to select multiple artifacts of the same type.

> **Tip**
>
> All related entities for the selected artifact are automatically added to the package. You do not need to find and select each related artifact individually. If you change the Artifact Type, any selections will be cleared from the Select Artifacts field.

9. When you have finished selecting artifacts, click **Save** to add the artifacts to the package. The artifacts are added to the list at the bottom of the screen. For example, the package shown in the image contains three Graphmarts. The **Included Artifacts** column shows the total number of artifacts that are related to the core member and are also included in the package.



Configuration

Export File Format: File Per Category

Exported ACLs Handling: Use Existing ACLs as is

Export Options

☑ Generate Variable Template File

☑ Include Registry Statements

☐ Include Dataset Editions and Components

General

Type	MigrationPackage
Creator	System Administrator
Updated	an hour ago
Released	an hour ago

http://cambridgesemantics.com/MigrationPa

Core Members

Title	Type	Available	# Included Artifacts	Actions
MySQL DB - emrdbs	Graphmart	✓	25	⊗
MySQL DB - northwi	Graphmart	✓	43	⊗
Sample Movie Data	Graphmart	✓	17	⊗

> **Tip**
>
> You can view specifics about the included artifacts on the **Included Artifacts** tab. More information about the tab is included in Editing Migration Package Template Files.

10. Once the desired artifacts have been added to the package, review the export options at the top of the screen and make adjustments as needed. For details about each of the settings, see Export Configuration Settings Reference.

Once the Migration Package includes the desired artifacts and the export options are configured, the package can be exported. For instructions, see Exporting a Migration Package.

Related Topics

Exporting a Migration Package

Editing Migration Package Template Files

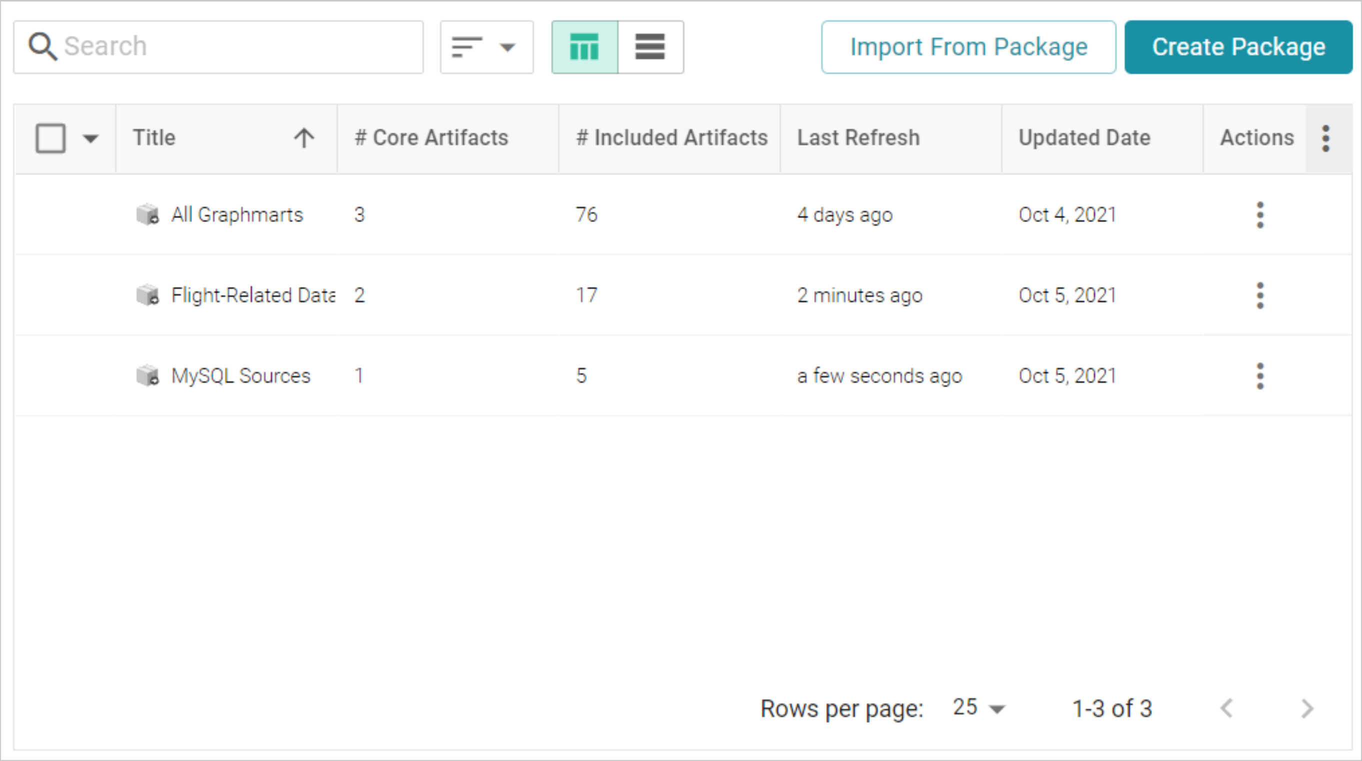
Export Configuration Settings Reference

Exporting a Migration Package

Follow the steps below to export a Migration Package.

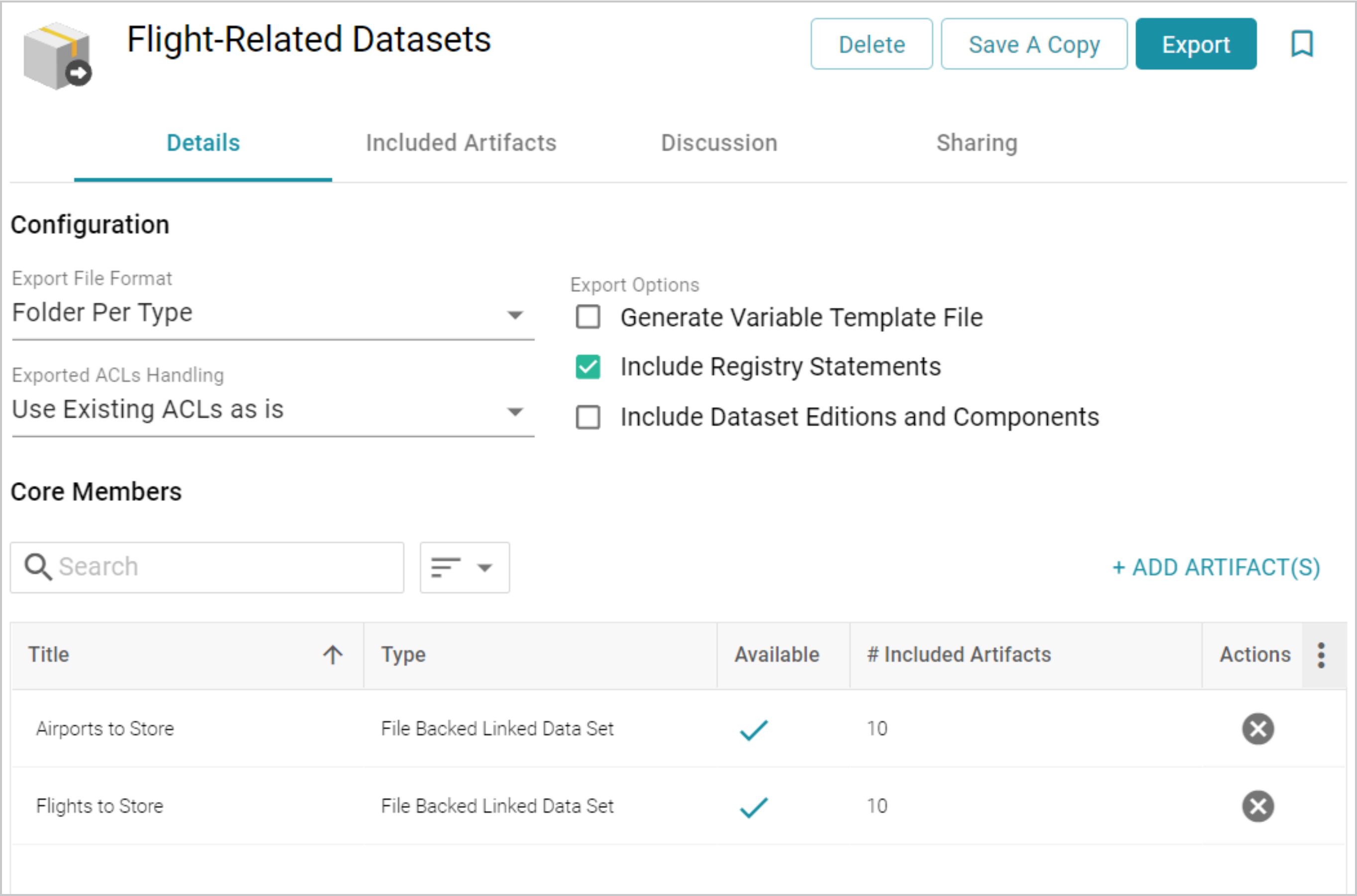
> **Note**
> There are two permissions that control access to export, modify, and import migration packages: **Manage Migration Packages** and **Perform Migration Package Operations As Sysadmin**. If a user has only the Manage Migration Packages permission, they cannot modify, export, or import artifacts in packages unless they have the appropriate permissions on the artifacts. If a user has Perform Migration Package Operations As Sysadmin, that means they can modify, export, and import migration packages that include artifacts they may not otherwise have permission to operate on.

1. In the Administration application, expand the **Tools** menu and click **Migration Packages**. Anzo displays the Migration Packages screen, which lists any existing packages. For example:



□	Title	# Core Artifacts	# Included Artifacts	Last Refresh	Updated Date	Actions
	All Graphmarts	3	76	4 days ago	Oct 4, 2021	⋮
	Flight-Related Date	2	17	2 minutes ago	Oct 5, 2021	⋮
	MySQL Sources	1	5	a few seconds ago	Oct 5, 2021	⋮

2. Click the name of the Migration Package that you want to export. The Details tab for the package is displayed. For example:



Flight-Related Datasets

Configuration

Export File Format
Folder Per Type

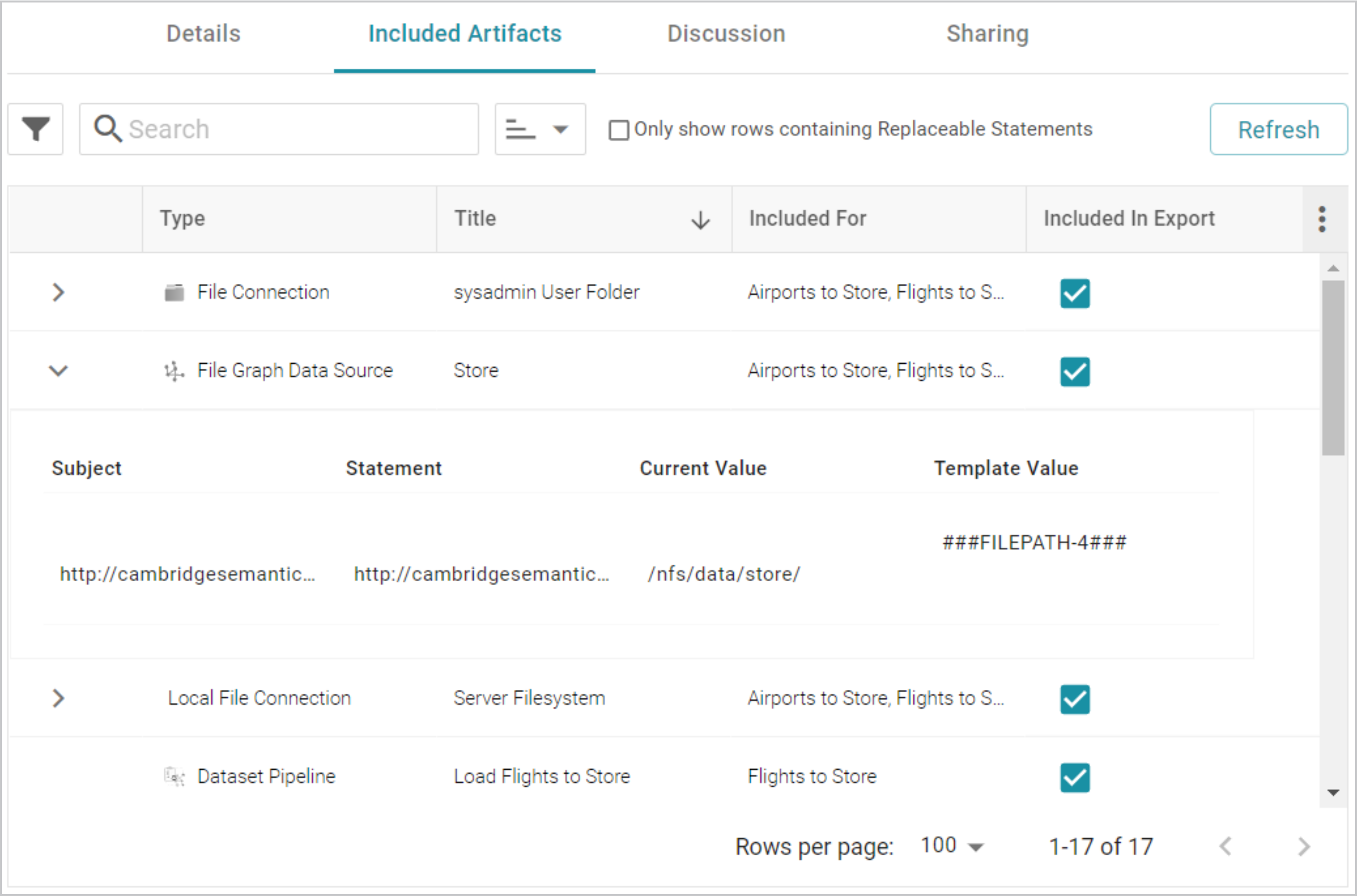
Exported ACLs Handling
Use Existing ACLs as is

Export Options
- ☐ Generate Variable Template File
- ☑ Include Registry Statements
- ☐ Include Dataset Editions and Components

Core Members

Title	Type	Available	# Included Artifacts	Actions
Airports to Store	File Backed Linked Data Set	✓	10	⊗
Flights to Store	File Backed Linked Data Set	✓	10	⊗

3. If desired, you can change the export configuration by adjusting the Configuration settings at the top of the screen. For details about the options, refer to Export Configuration Settings Reference.
4. If **Generate Variable Template File** is enabled and you want to change replaceable property values before performing the export and generating the template, you can click the **Included Artifacts** tab. The properties with editable values can be expanded by clicking the > character next to the artifact. For example:



	Type	Title	Included For	Included In Export
>	File Connection	sysadmin User Folder	Airports to Store, Flights to S...	☑
⌄	File Graph Data Source	Store	Airports to Store, Flights to S...	☑

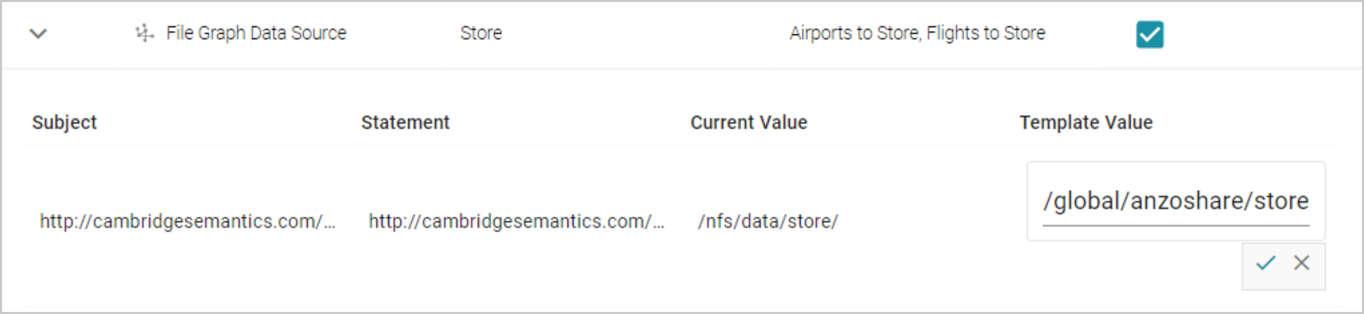
Subject	Statement	Current Value	Template Value
http://cambridgesemantic...	http://cambridgesemantic...	/nfs/data/store/	###FILEPATH-4###

	Type	Title	Included For	Included In Export
>	Local File Connection	Server Filesystem	Airports to Store, Flights to S...	☑
	Dataset Pipeline	Load Flights to Store	Flights to Store	☑

Tip

To filter the list to show only the rows that have replaceable values, you can select the **Only show rows containing Replaceable Statements** checkbox at the top of the screen.

5. To edit a statement, click the value in the **Template Value** column and replace the placeholder text with the desired value. Then click the checkmark icon (✓) to save the change. Any changes you make on the Included Artifacts tab will be included in the variable template that is generated during the export. For example, in the image below the Anzo Data Store placeholder is replaced with the path on the target server.



	Type	Title	Included For	Included In Export
⌄	File Graph Data Source	Store	Airports to Store, Flights to Store	☑

Subject	Statement	Current Value	Template Value
http://cambridgesemantics.com/...	http://cambridgesemantics.com/...	/nfs/data/store/	/global/anzoshare/store

6. When you are ready to export the package, click the **Export** button. Anzo exports each of the included artifacts as TriG files and packages the TriG files into a .zip file. The contents of the .zip file are laid out according to the specified Export File Format. Once the package is assembled, the .zip file is automatically downloaded to your computer.

> **Note**
> If changes were made to the artifacts since they were added to the package and the package was not refreshed before the export, Anzo automatically creates a Version of the changed artifacts.

Once the package is exported, you can extract the file to access any generated templates and to place the artifacts in source control if that is part of your organization's process. For information about working with the generated templates, see Editing Migration Package Template Files. When you are ready to import the package into the target server, see Importing a Migration Package.

Related Topics

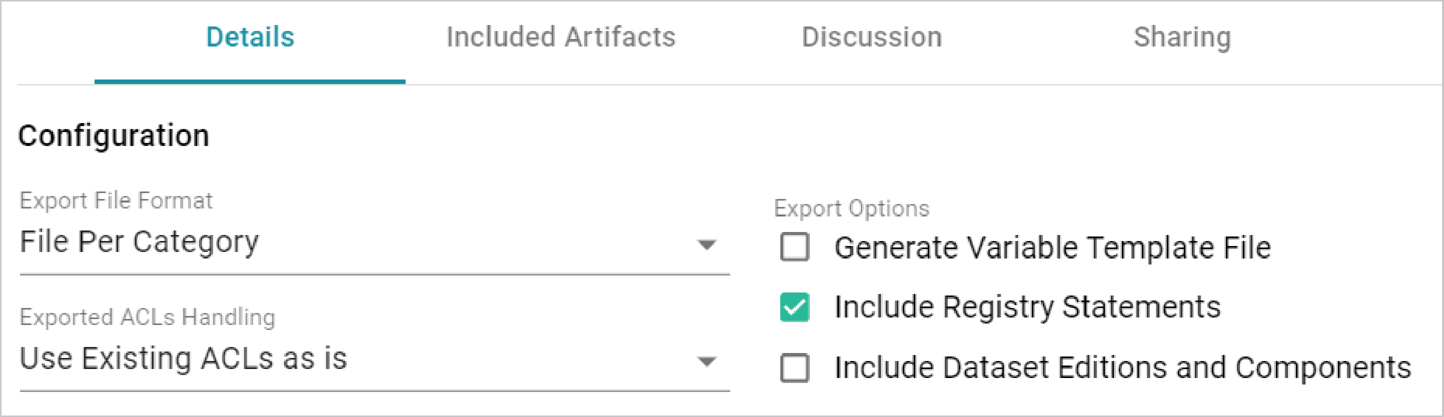
Export Configuration Settings Reference

Editing Migration Package Template Files

Importing a Migration Package

Export Configuration Settings Reference

This topic describes the Export Configuration options that are available on the Details tab when creating or configuring a migration package.



Configuration

Export File Format
File Per Category

Exported ACLs Handling
Use Existing ACLs as is

Export Options
☐ Generate Variable Template File
☑ Include Registry Statements
☐ Include Dataset Editions and Components

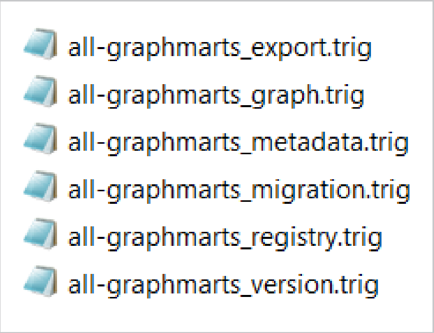
- Export File Format
- Exported ACLs Handling
- Generate Variable Template File
- Include Registry Statements
- Include Dataset Editions and Components

Export File Format

This option configures the file structure of the exported .zip package. There are three options to choose from:

File Per Category

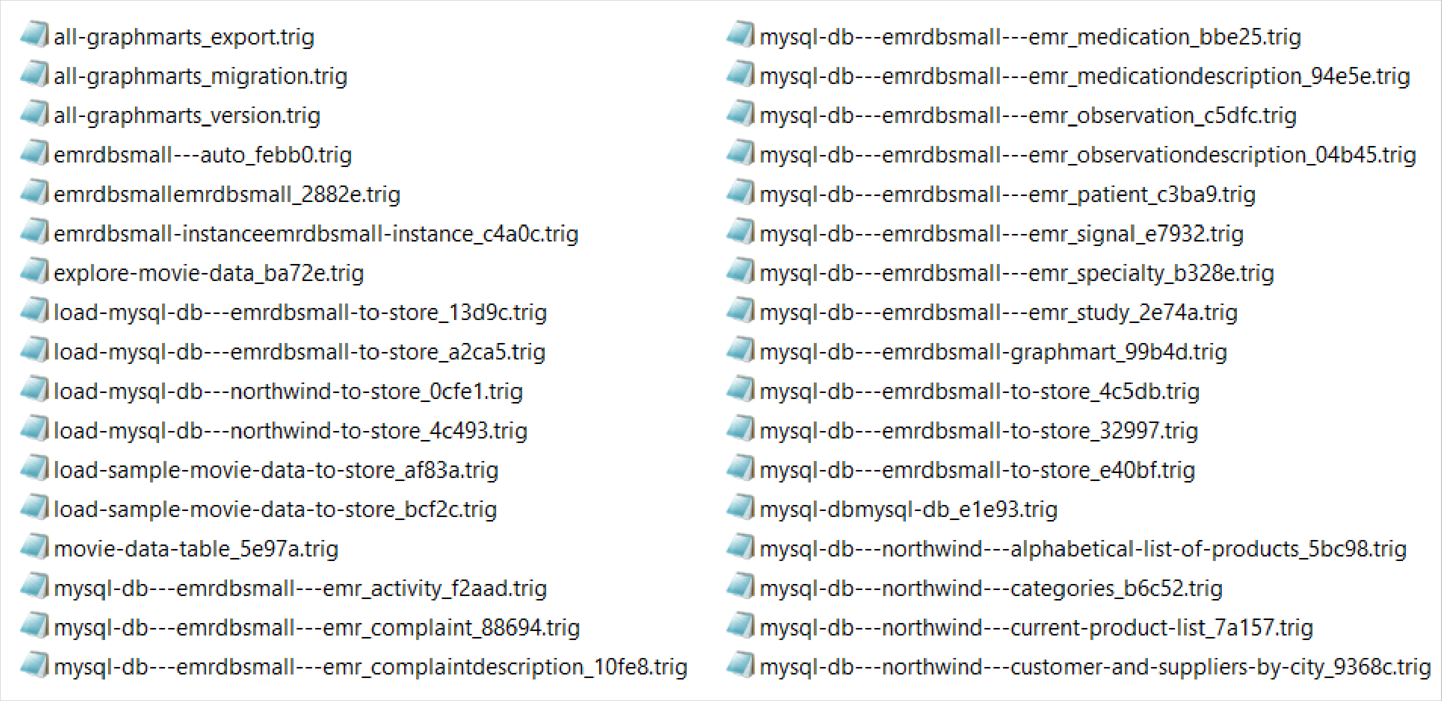
This option (the default setting) creates one TriG file per type or category of information that is included in the export. This is the same as the layout of files that results when you export an artifact from the Versions tab in the Anzo application. If **File Per Category** is selected, the exported package contains one file per each of the following categories: Export, Migration, Versions, Metadata, Registries, and Graph. The files that are generated depend on the chosen Export Options. The relevant information for all of the included artifacts is written to the same category file. For example, the image below shows the contents of a package that was exported with Export File Format set to **File Per Category**. The package name is "All Graphmarts."



```
all-graphmarts_export.trig
all-graphmarts_graph.trig
all-graphmarts_metadata.trig
all-graphmarts_migration.trig
all-graphmarts_registry.trig
all-graphmarts_version.trig
```

File Per Graph

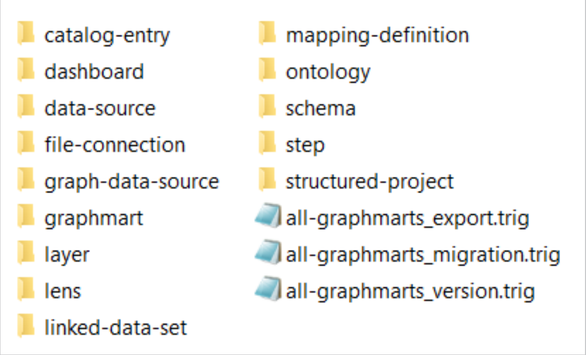
This option creates one TriG file per graph. Unlike the Files Per Category option, where data is separated by type of information, each graph file contains all of the data that is related to that graph, such as the metadata and registry information. For example, the image below shows the contents of a package that was exported with Export File Format set to **File Per Graph**. There is a TriG file for each data source graph, mapping graph, pipeline graph, dashboard graph, etc. The package name is "All Graphmarts."



```
all-graphmarts_export.trig
all-graphmarts_migration.trig
all-graphmarts_version.trig
emrdbsmall---auto_febb0.trig
emrdbsmallemrdbsmall_2882e.trig
emrdbsmall-instanceemrdbsmall-instance_c4a0c.trig
explore-movie-data_ba72e.trig
load-mysql-db---emrdbsmall-to-store_13d9c.trig
load-mysql-db---emrdbsmall-to-store_a2ca5.trig
load-mysql-db---northwind-to-store_0cfe1.trig
load-mysql-db---northwind-to-store_4c493.trig
load-sample-movie-data-to-store_af83a.trig
load-sample-movie-data-to-store_bcf2c.trig
movie-data-table_5e97a.trig
mysql-db---emrdbsmall---emr_activity_f2aad.trig
mysql-db---emrdbsmall---emr_complaint_88694.trig
mysql-db---emrdbsmall---emr_complaintdescription_10fe8.trig
mysql-db---emrdbsmall---emr_medication_bbe25.trig
mysql-db---emrdbsmall---emr_medicationdescription_94e5e.trig
mysql-db---emrdbsmall---emr_observation_c5dfc.trig
mysql-db---emrdbsmall---emr_observationdescription_04b45.trig
mysql-db---emrdbsmall---emr_patient_c3ba9.trig
mysql-db---emrdbsmall---emr_signal_e7932.trig
mysql-db---emrdbsmall---emr_specialty_b328e.trig
mysql-db---emrdbsmall---emr_study_2e74a.trig
mysql-db---emrdbsmall-graphmart_99b4d.trig
mysql-db---emrdbsmall-to-store_4c5db.trig
mysql-db---emrdbsmall-to-store_32997.trig
mysql-db---emrdbsmall-to-store_e40bf.trig
mysql-dbmysql-db_e1e93.trig
mysql-db---northwind---alphabetical-list-of-products_5bc98.trig
mysql-db---northwind---categories_b6c52.trig
mysql-db---northwind---current-product-list_7a157.trig
mysql-db---northwind---customer-and-suppliers-by-city_9368c.trig
```

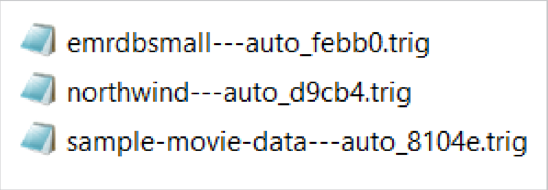
Folder Per Type

Like the File Per Graph option, this option creates one TriG file per graph, where each graph file contains all of the data that is related to the graph, such as the metadata and registry information. However, the graph files are organized into subdirectories by base graph type, such as data source, graphmart, layer, schema, etc. For example, the image below shows the contents of a package that was exported with Export File Format set to **Folder Per Type**. The package name is "All Graphmarts."



catalog-entry
dashboard
data-source
file-connection
graph-data-source
graphmart
layer
lens
linked-data-set
mapping-definition
ontology
schema
step
structured-project
all-graphmarts_export.trig
all-graphmarts_migration.trig
all-graphmarts_version.trig

The folders contain all graphs of that type for all of the included artifacts. For example, the ontology folder shown below contains the ontology graphs for the three graphmarts that are included in the package.



emrdbsmall---auto_febb0.trig
northwind---auto_d9cb4.trig
sample-movie-data---auto_8104e.trig

Exported ACLs Handling

This option determines how to handle the ACL configuration for the artifacts in the package. There are two options to choose from:

Use Existing ACLs as is

This option exports the ACL metadata for all of the artifacts as-is. No template file will be generated and the artifacts will be imported into the target system with the same permissions as the artifacts on the source system.

Generate Access Control Template File

This option generates a template file that contains access control statements with placeholder values in the objects. You replace the placeholder values with the Group or User URIs that should have permission to access all of the artifacts in the Migration Package. For more information about the template, see Editing Migration Package Template Files.

Generate Variable Template File

This option indicates whether to generate a Variable Template File in the export package. A Variable Template File is a TriG file that contains statements for all of the properties that have replaceable values. Properties with replaceable values are objects such as file paths and Anzo Data Store locations, which might differ on the source and target Anzo servers. The template that is generated has placeholder text that you replace with the desired values for the target server. If you want a template to be generated, select the **Generate Variable Template File** checkbox. If you do not want to make changes to artifacts before they are imported to the target system, clear the **Generate Variable Template File** checkbox. For more information about the template, see Editing Migration Package Template Files.

> Tip
>
> Enabling the **Generate Variable Template File** setting also makes the replaceable properties editable on the **Included Artifacts** tab. Properties are not editable on the Included Artifacts tab when Generate Variable Template Files is disabled.

Include Registry Statements

This option is selected by default and indicates whether to export the registry statements for the artifacts in the package. A registry is like a container for all artifacts of a certain type. For example, the Data Sources Registry stores information about all of the data sources. Registry statements should be included in exports except in rare cases when you do not intend to import the migration package back into Anzo. When registry statements are not included in an export, the imported artifacts are not displayed in Anzo. For example, if a data source artifact is imported without registry statements, it would not be added to the Data Sources Registry and therefore not be displayed in the list of data sources in the Anzo application.

Include Dataset Editions and Components

This option specifies whether the export includes all of the editions for each dataset in the package. When **Include Dataset Editions and Components** is selected, the exported package includes the Managed and Saved Editions and all of their components for each dataset.

Related Topics

Creating a Migration Package

Exporting a Migration Package

Editing Migration Package Template Files

Editing Migration Package Template Files

If a Variable Template File and/or Access Control Template File is included in a Migration Package export, the files must be edited to replace all of the placeholder values before the package can be imported to the target server. This topic describes the template files and provides guidance on editing the templates.

- Editing a Variable Template File
- Editing an Access Control Template File

Editing a Variable Template File

A Variable Template File is a TriG file that is used to define the values to use in Replaceable Statements. Properties with replaceable values are objects such as File Connection paths, Anzo Data Store locations, File-Backed Linked Data Set locations, and file locations for file-based Data Sources, which might differ on the source and target Anzo servers. The template that is generated has placeholder text that you replace with the desired values for the target server. **The values that you specify are applied to all artifacts included the Migration Package**.

When you open a Variable Template File, the placeholder values are denoted by three hash characters (`###`) and all capital letters, for example, `###FILEPATH-1###`. The example below shows a snippet of a Variable Template File. The placeholder text is shown in **bold**:

```
<http://openanzo.org/ReplacementObject/10441bd7-03fb-494a-b56a-cc0eea32aed2> {

<http://cambridgesemantics.com/PathConnection/6bd218a15c644045ba43f007c824d830>
<http://cambridgesemantics.com/ontologies/DataSources#filePath> "###FILEPATH-
2###" .

  <http://openanzo.org/ReplacementObject/10441bd7-03fb-494a-b56a-cc0eea32aed2> a
<http://cambridgesemantics.com/ontologies/2021/06/Migration#ReplacementObject> ;
    <http://cambridgesemantics.com/ontologies/2021/06/Migration#forGraph>
<http://cambridgesemantics.com/CSVDataSource/f9a54e23d83549999536782b5de1981c> .
}

<http://openanzo.org/ReplacementObject/1c79ae8a-c570-4170-a9f8-f1a5dd967c6d> {
  <http://csi.com/DataLocation/157ae35ecab30f803c754d314be18e44>
<http://cambridgesemantics.com/ontologies/DataSources#filePath> "###FILEPATH-
9###" .
```

```
  <http://openanzo.org/ReplacementObject/1c79ae8a-c570-4170-a9f8-f1a5dd967c6d> a
<http://cambridgesemantics.com/ontologies/2021/06/Migration#ReplacementObject> ;
    <http://cambridgesemantics.com/ontologies/2021/06/Migration#forGraph>
<http://csi.com/FileBasedLinkedDataSet/157ae35ecab30f803c754d314be18e44> .
}

<http://openanzo.org/ReplacementObject/21f24dda-fba6-47d7-bb6d-90e1fafec623> {
  <http://csi.com/DataLocation/7214e9ec270347dabeccbfc7328b4bed>
<http://cambridgesemantics.com/ontologies/DataSources#filePath> "###FILEPATH-
7###" .

  <http://openanzo.org/ReplacementObject/21f24dda-fba6-47d7-bb6d-90e1fafec623> a
<http://cambridgesemantics.com/ontologies/2021/06/Migration#ReplacementObject> ;
    <http://cambridgesemantics.com/ontologies/2021/06/Migration#forGraph>
<http://csi.com/FileBasedLinkedDataSet/7214e9ec270347dabeccbfc7328b4bed> .
}
```

When replacing the placeholder text, edit the text inside the quotation marks. All objects should retain the quotes. If the replacement value is a URI, place the URI inside the quotation marks.

Editing an Access Control Template File

The Access Control Template is a TriG file that is used to define the permissions to be assigned **on all artifacts included in the Migration Package**. The template contains two sets of statements, one set for the artifact graphs (`<http://openanzo.org/namedGraphs/reserved/graphs/defaultGraphTemplate>`) and one for the artifact metadata graphs (`<http://openanzo.org/namedGraphs/reserved/graphs/defaultMetadataGraphTemplate>`). The objects in the template are placeholder URIs that must be replaced with the Group and/or User URIs on the target server. A copy of the template is shown below. The placeholder URIs are shown in **bold**:

```
<http://openanzo.org/namedGraphs/AclTemplate> {
  <http://openanzo.org/namedGraphs/reserved/graphs/defaultGraphTemplate>
<http://openanzo.org/ontologies/2008/07/Anzo#canBeAddedToBy> <urn://ACL-ADD-
ROLE-PLACEHOLDER> ;
    <http://openanzo.org/ontologies/2008/07/Anzo#canBeReadBy> <urn://ACL-READ-
ROLE-PLACEHOLDER> ;
    <http://openanzo.org/ontologies/2008/07/Anzo#canBeRemovedFromBy> <urn://ACL-
REMOVE-ROLE-PLACEHOLDER> .

  <http://openanzo.org/namedGraphs/reserved/graphs/defaultMetadataGraphTemplate>
<http://openanzo.org/ontologies/2008/07/Anzo#canBeAddedToBy> <urn://ACL-METAADD-
ROLE-PLACEHOLDER> ;
    <http://openanzo.org/ontologies/2008/07/Anzo#canBeReadBy> <urn://ACL-
METAREAD-ROLE-PLACEHOLDER> ;
    <http://openanzo.org/ontologies/2008/07/Anzo#canBeRemovedFromBy> <urn://ACL-
METAREMOVE-ROLE-PLACEHOLDER> .
}
```

The **defaultGraphTemplate** statements configure who can view, modify, and delete the artifact. The **defaultMetadataGraphTemplate** statements configure who can view, modify, and delete artifact metadata, such as an artifact's permissions. The list below describes how the template properties map to permissions:

canBeReadBy

This property assigns **View** and **Meta View** permissions. On the **defaultGraphTemplate**, this property assigns **View**, which grants access to see the artifact but not change it. On the **defaultMetadataGraphTemplate**, this property assigns **Meta View**, which grants access to see the artifact's permissions but not change them.

canBeAddedToBy

This property assigns **Add/Edit** and **Meta Add/Edit** permissions. On the **defaultGraphTemplate**, this property assigns **Add/Edit**, which grants permission to change the artifact or add an entity to it, such as to add a Schema to a Data Source. On the **defaultMetadataGraphTemplate**, this property assigns **Meta Add/Edit**, which grants permission to change the artifact's permissions.

canBeRemovedFromBy

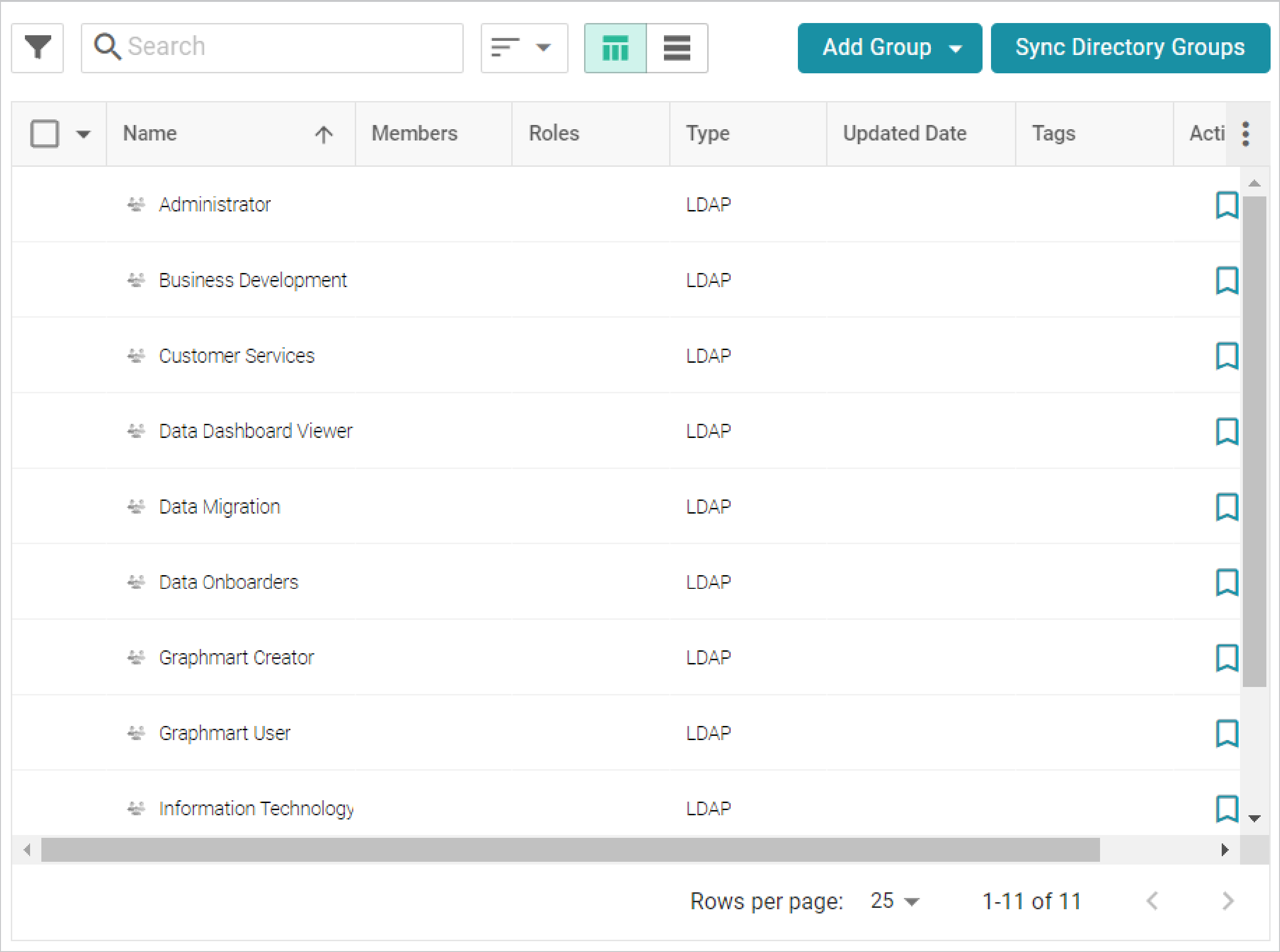
This property assigns **Delete** and **Meta Delete** permissions. On the **defaultGraphTemplate**, this property assigns **Delete**, which grants permission to delete an entity from an artifact, such as to delete a Data Layer from a Graphmart. On the **defaultMetadataGraphTemplate**, this property assigns **Meta Delete**, which grants permission to delete the parent artifact and change the artifact's permissions.

> **Tip**
> For more information about artifact permissions, see Permission Settings in the User Guide.

Finding Group and User URIs

In order to complete the Access Control Template and give groups access to the artifacts in the package, you need to find the Group and/or User URIs on the target server to add as objects to the template properties (canBeReadBy, canBeAddedToBy, and canBeRemovedFromBy).

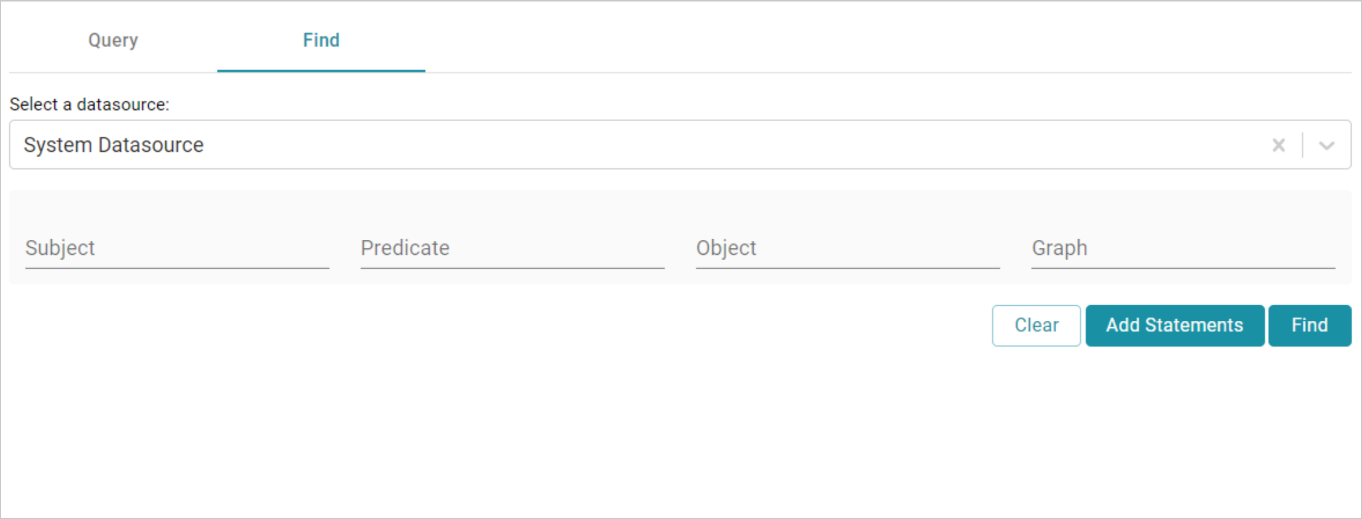
1. If you need to review a list of the Groups that are available on the target system, open the Administration application on that server. To access the Group names, expand the **User Management** menu and click **Groups**. For example:



Name	Members	Roles	Type	Updated Date	Tags	Acti
Administrator			LDAP			
Business Development			LDAP			
Customer Services			LDAP			
Data Dashboard Viewer			LDAP			
Data Migration			LDAP			
Data Onboarders			LDAP			
Graphmart Creator			LDAP			
Graphmart User			LDAP			
Information Technology			LDAP			

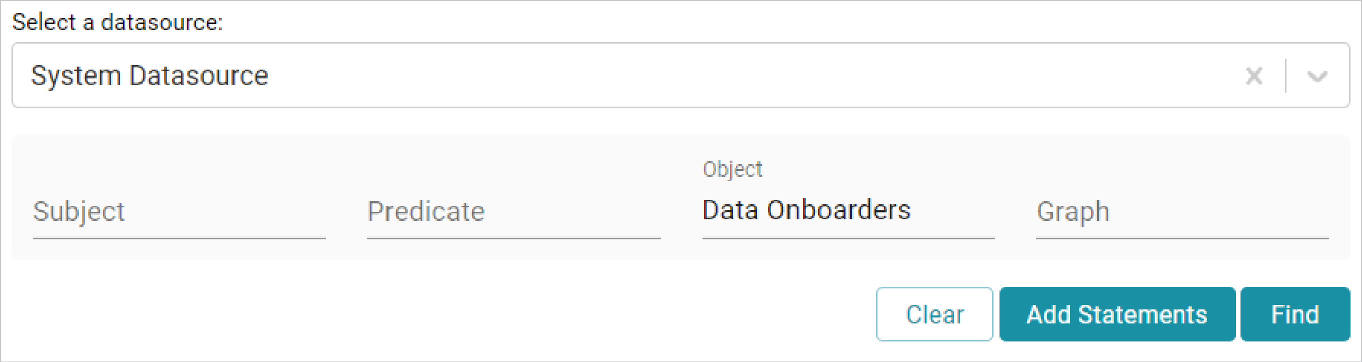
Rows per page: 25 1-11 of 11

2. Note the names of the Groups whose URIs you want to add to the template.
3. Next, find the URIs for the Group names. In the Anzo application on the target server, expand the **Access** menu and click **Query Builder**. Anzo displays the Query tab. Click the **Find** tab.



Select a datasource: System Datasource

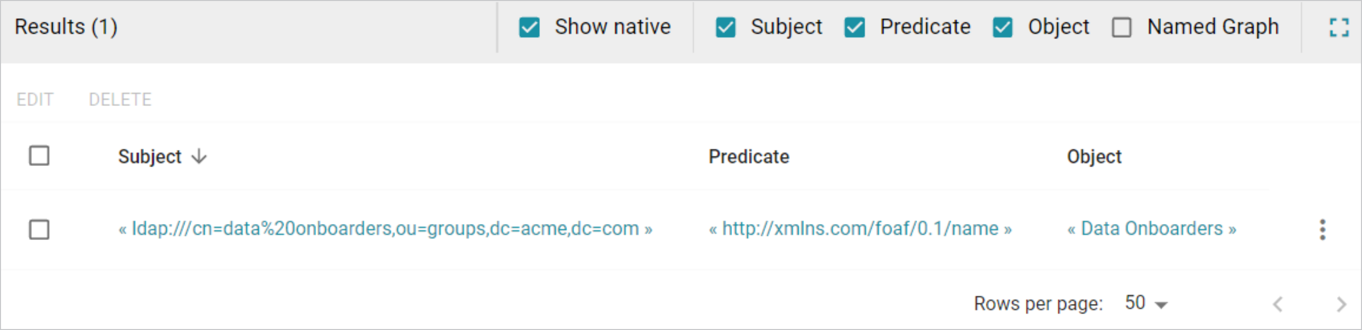
4. On the Find tab, leave the datasource set to **System Datasource** and then type a Group Name in the **Object** field. For example:



Select a datasource: System Datasource

Object: Data Onboarders

5. Next, click the **Find** button. The result is a statement that defines that Group. The value in the **Subject** position is the URI for the Group. For example:



□	Subject ↓	Predicate	Object
□	« ldap:///cn=data%20onboarders,ou=groups,dc=acme,dc=com »	« http://xmlns.com/foaf/0.1/name »	« Data Onboarders »

6. Click the URI to add it to the **Subject** field at the top of the screen, and then copy the URI from that field. For example, the URI copied from the image above is `<ldap:///cn=data%20onboarders,ou=groups,dc=acme,dc=com>`.

Repeat the steps above to find all of the URIs that you want to add to the template. To add URIs to the file, replace each of the placeholder URIs. You can add multiple URIs to a property in a comma-separated list. For example:

```
<http://openanzo.org/namedGraphs/AclTemplate> {
  <http://openanzo.org/namedGraphs/reserved/graphs/defaultGraphTemplate>
<http://openanzo.org/ontologies/2008/07/Anzo#canBeAddedToBy>
        <ldap:///cn=data%20onboarders,ou=groups,dc=acme,dc=com>,
<ldap:///cn=graphmart%20creator,ou=groups,dc=acme,dc=com>,
<ldap:///cn=graphmart%20user,ou=groups,dc=acme,dc=com>  ;
    <http://openanzo.org/ontologies/2008/07/Anzo#canBeReadBy>
<ldap:///cn=graphmart%20user,ou=groups,dc=acme,dc=com> ;
    <http://openanzo.org/ontologies/2008/07/Anzo#canBeRemovedFromBy>
<ldap:///cn=administrator,ou=groups,dc=acme,dc=com> .

  <http://openanzo.org/namedGraphs/reserved/graphs/defaultMetadataGraphTemplate>
<http://openanzo.org/ontologies/2008/07/Anzo#canBeAddedToBy>
<ldap:///cn=data%20onboarders,ou=groups,dc=acme,dc=com> ;
    <http://openanzo.org/ontologies/2008/07/Anzo#canBeReadBy>
<ldap:///cn=graphmart%20user,ou=groups,dc=acme,dc=com> ;
    <http://openanzo.org/ontologies/2008/07/Anzo#canBeRemovedFromBy>
<ldap:///cn=administrator,ou=groups,dc=acme,dc=com> .
}
```

Related Topics

Creating a Migration Package

Exporting a Migration Package

Export Configuration Settings Reference

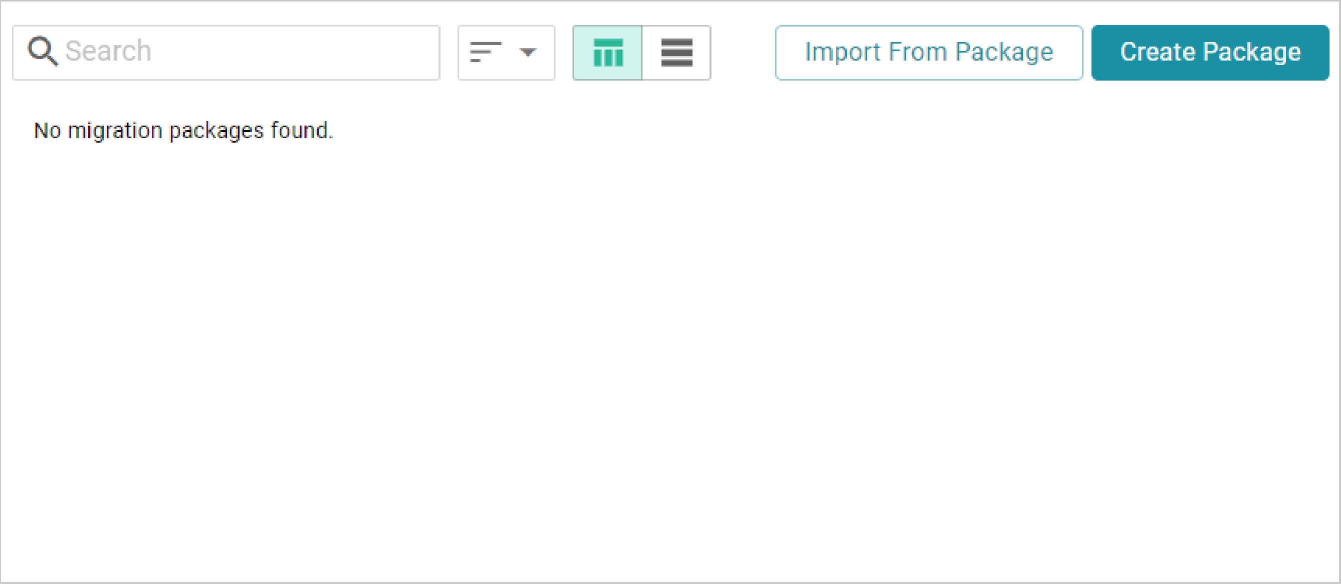
Importing a Migration Package

Importing a Migration Package

Follow the instructions below to import a migration package.

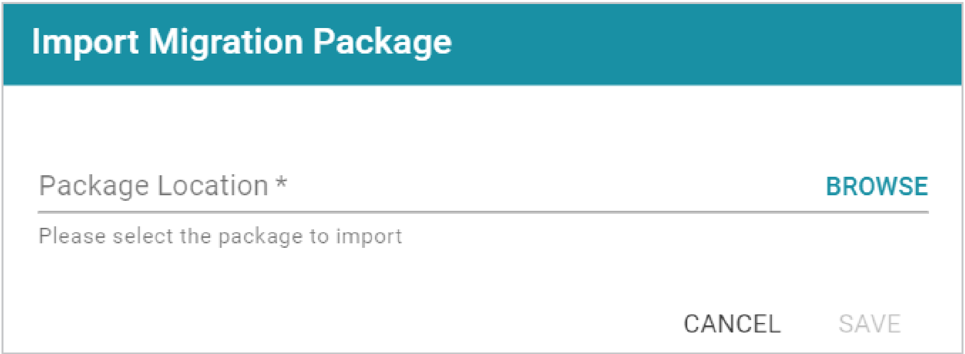
> **Note**
> There are two permissions that control access to export, modify, and import migration packages: **Manage Migration Packages** and **Perform Migration Package Operations As Sysadmin**. If a user has only the Manage Migration Packages permission, they cannot modify, export, or import artifacts in packages unless they have the appropriate permissions on the artifacts. If a user has Perform Migration Package Operations As Sysadmin, that means they can modify, export, and import migration packages that include artifacts they may not otherwise have permission to operate on.

1. First, if the package contains ACL and/or Variable Template files that were exported from the source Anzo server, make sure the files have been completed; all of the placeholder values are replaced with the desired values for the target server. For information about the templates, see Editing Migration Package Template Files.
2. If the package is unpacked, compress the directory to a .zip file so that it can be imported. Any template files should be included inside the .zip file. The package can be imported from your computer or a location on the target server's File Store.
3. In the Administration application on the target server, expand the **Tools** menu and click **Migration Packages**. Anzo displays the Migration Packages screen, which lists any packages that were created on this server. For example, the image below shows a target system where packages have been imported but not created:



No migration packages found.

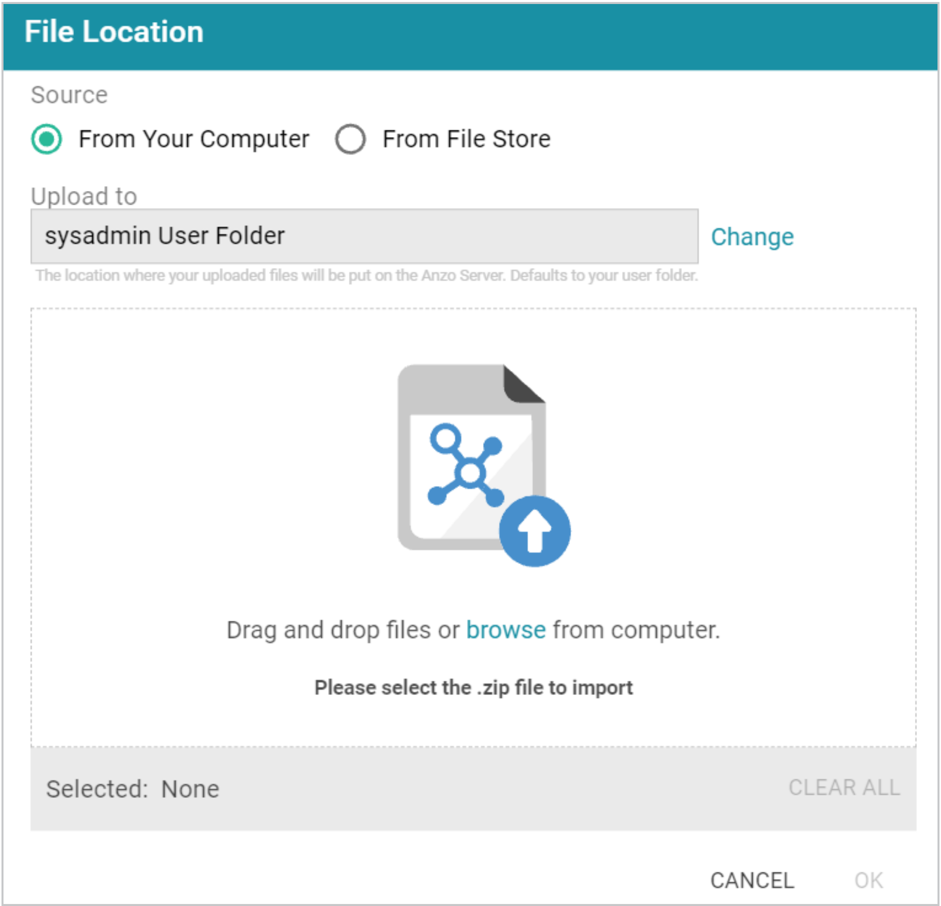
4. Click the **Import From Package** button at the top of the screen. The Import Migration Package dialog box is displayed:


Import Migration Package

Package Location *

Please select the package to import

5. Click **Browse** or the **Package Location** field to open the File Location dialog box.



File Location

Source

● From Your Computer ○ From File Store

Upload to

sysadmin User Folder

Change

The location where your uploaded files will be put on the Anzo Server. Defaults to your user folder.

Drag and drop files or browse from computer.

Please select the .zip file to import

Selected: None

CLEAR ALL

CANCEL OK

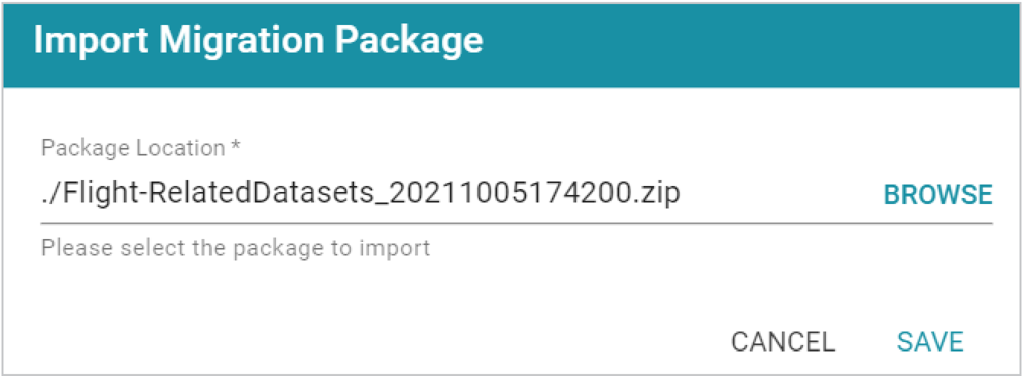
6. Depending on the location of the package to import, follow the appropriate instructions below:

   - If the package is on your computer, leave **From Your Computer** selected and drag and drop the file to the dialog box or click **browse** and select the file.

     > **Tip**
     >
     > As a best practice when uploading files from your computer, check the upload location that is listed in the **Upload To** field by hovering your pointer over the value to view the full path as a tooltip. Make sure the upload location is set to the desired directory. If necessary, you can click **Change** and select a different upload path.

   - If the package is on the File Store, select **From File Store**. Navigate to the location of the .zip file on the store and select it.

7. Click **OK** to add the file location to the Package Location field. For example:



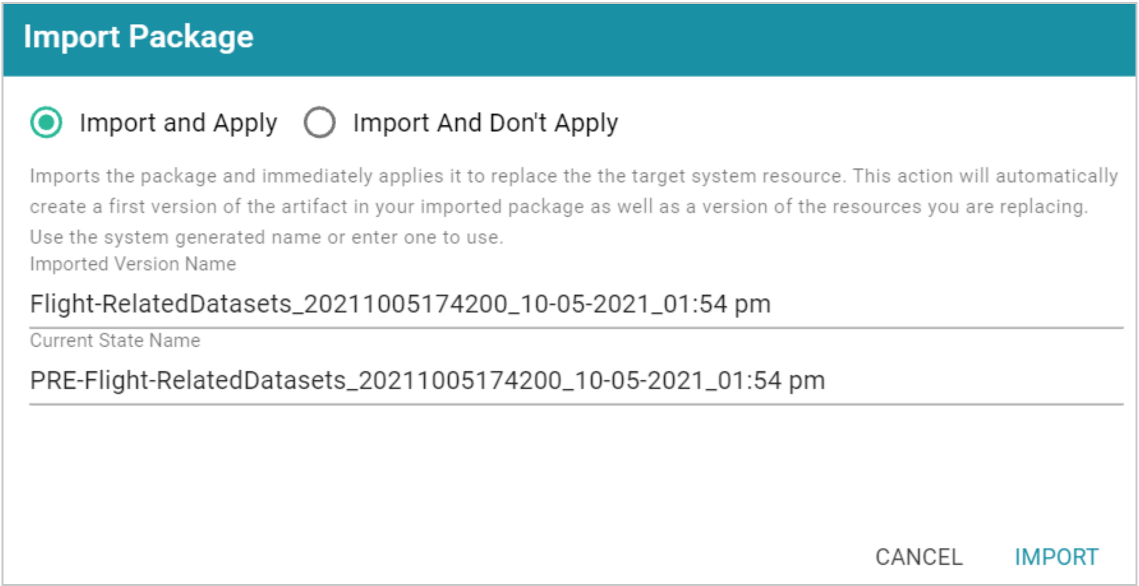
Import Migration Package

Package Location *

./Flight-RelatedDatasets_20211005174200.zip

Please select the package to import

8. Click **Save** to save the import configuration. Anzo validates the import by checking whether any included template files are completed. If the import is valid, the Import Package screen is displayed. For example:



Import Package

● Import and Apply ○ Import And Don't Apply

Imports the package and immediately applies it to replace the the target system resource. This action will automatically create a first version of the artifact in your imported package as well as a version of the resources you are replacing.

Use the system generated name or enter one to use.

Imported Version Name

Flight-RelatedDatasets_20211005174200_10-05-2021_01:54 pm

Current State Name

PRE-Flight-RelatedDatasets_20211005174200_10-05-2021_01:54 pm

9. On the Import Package screen, specify how you want the artifacts to be applied to the target server, either **Import And Apply** or **Import And Don't Apply**:

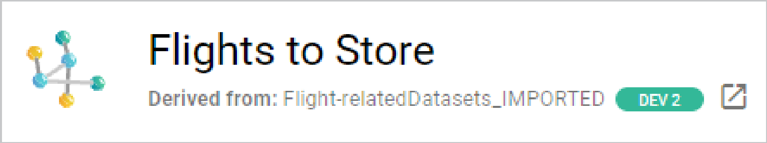
Import And Apply

Selecting this option means the artifacts included in the package should be applied to the target system as the current, working versions of the artifacts. When **Import And Apply** is selected, Anzo follows the procedure below:

a. If the artifacts to be imported also exist on the target system, Anzo compares the existing version with the import version. If the artifacts differ, Anzo creates a backup version of the existing artifacts. If the artifacts match, Anzo does not create backup versions of the existing artifacts.

b. Next, Anzo imports the artifacts from the package as versions. This ensures that the target server includes a copy of the artifacts exactly as they were originally imported.

c. The imported version of the artifacts are applied as the current, working version. In other words, the current version is now derived from the imported version and is given a **Derived from: <imported_version_name>** label. For example, the image below shows the label for a Dataset that was derived from an imported version.



Flights to Store

Derived from: Flight-relatedDatasets_IMPORTED DEV 2

Import And Don't Apply

Selecting this option means the artifacts included in the package should not be applied to the target system as the current, working versions. When **Import And Don't Apply** is selected, Anzo imports the artifacts as backup versions and does not replace the current versions of any existing artifacts.

10. Next, you have the option to modify the auto-generated names for the versions that are created during the import:
    - If you selected **Import And Apply**, you can edit the following values:
        - **Imported Version Name**: This is the name of the new version that is created by the import.
        - **Current State Name**: If the existing version of an artifact differs from the imported version, this is the name to give the backup version of the current state before the imported version is applied.
    - If you selected **Import And Don't Apply**, you can edit the **Imported Version Name** value to specify the name to give the imported version.
11. When you are ready to import the package, click **Import**. Anzo imports all of the artifacts according to your import configuration.

> **Note**
> When the import is complete, the imported package is not displayed in the list of Migration Packages. The Migration Packages screen displays only the packages that are created on this server.

Related Topics

Exporting a Migration Package

Editing Migration Package Template Files

Export Configuration Settings Reference

User Management

Anzo offers granular artifact and data access control as well as role-based security for controlling access to the Anzo applications and features. This section provides setup and administration information for role-based access control. The topics include instructions for connecting to your central directory server, connecting to an identity provider for SSO access, and configuring users, groups, roles, and permissions in Anzo.

> **Tip**
> When planning the user and access management solution for your system, Cambridge Semantics recommends that you refer to User Management and Access Control Concepts to learn about the fundamental concepts behind Anzo's access control implementation.

User Management and Access Control Concepts

The topics in this section provide an overview of user management and access control in Anzo and introduce the key concepts to consider when planning and implementing user and data access management for your system.

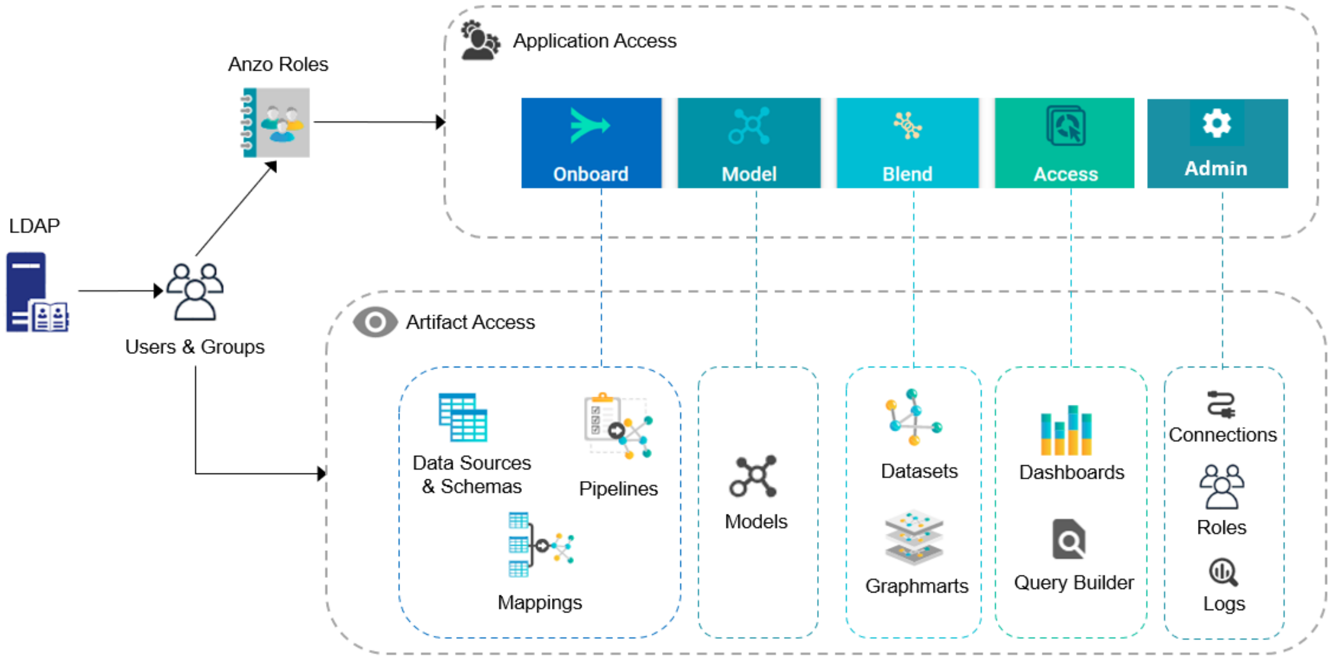
User Management Concepts

Typically organizations connect Anzo to their central directory server and then add users and groups from the server to Anzo. Once the accounts are added to Anzo, access control is managed in two ways:

1. Groups (or users) are added to **Roles** and the roles are configured to grant access to *functionality* in Anzo. Role permissions grant access to menus and screens in the Anzo and Administration applications. Access to functionality cannot be assigned to groups or users, only to roles.
2. Groups and users are used to control access to individual artifacts—Project's Data Sources, Models, Mappings, Pipelines, Graphmarts, etc.—and your data that is stored in Anzo.

> **Note**
> Though Anzo is flexible and allows you to assign artifact access to roles, the recommendation is to control access to artifacts with users and groups and reserve roles for granting access to functions in the applications.

The following diagram illustrates the concepts of roles and groups in Anzo:

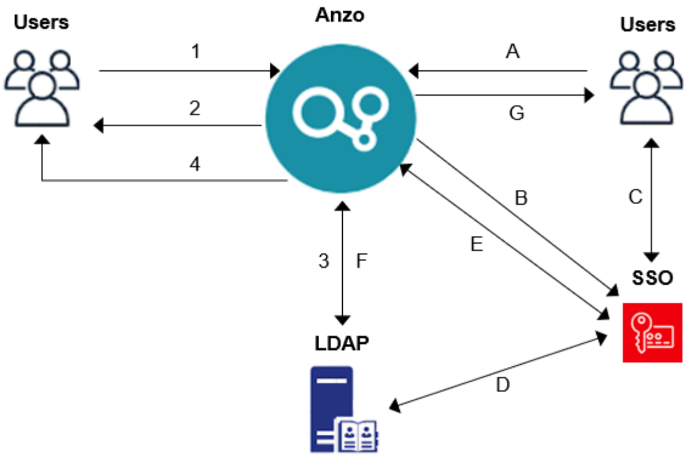


A user's role determines whether they can access the **Onboard** menu and create a new Data Source or see the **Blend** menu and create a new Graphmart. But their group assignment determines whether they can view, modify, or delete Data Source and Graphmart artifacts that are created by other users.

For more information about leveraging a directory server and details about users, groups, and roles see the sections below.

Leveraging a Directory Server (LDAP)

Anzo can be configured to access your directory server via Direct Authorization or Single Sign-On (SSO). The diagram below shows the procedures that are followed for both methods. The left side of the diagram (the numbered steps) shows the direct authorization method. The right side of the diagram (the lettered steps) shows the SSO method. The table below the diagram describes the processes for each method.



Direct Authorization	Single Sign-On
1. A new (unknown) user navigates to the Anzo application.	A. A new (unknown) user navigates to the Anzo application.
2. Anzo redirects the user to a login form. The user supplies credentials and submits the form.	B. Anzo redirects the user to the SSO provider. The SSO provider controls authentication validation.
3. Anzo queries the LDAP for the user and group membership.	C. Depending on the policy, the SSO provider presents a login screen for the user to complete and submit.
4. Anzo redirects the user to the application with the appropriate roles applied.	D. As needed, the SSO provider validates the credentials with the LDAP server.
	E. The SSO provider authenticates the Anzo session with a callback.

Direct Authorization	Single Sign-On
	F. Anzo fetches group information from the LDAP server. **Note** For SSO-configured systems, Anzo currently requires direct access to the LDAP directory (and a bind user) to look up groups. G. Anzo redirects the user to the application with the appropriate roles applied.

For more information on connecting to a directory server, see the following topics:

- Connecting to a Directory Server
- Connecting to an SSO Provider

Users and Groups

Groups typically originate in a directory server and are synced to Anzo. However, you can also create custom groups that are internal to Anzo. Typically users also originate from the directory server, but you can create user accounts in Anzo. Any users and groups that are created in Anzo are stored in Anzo's internal LDAP server.

For information about retrieving user and groups from the directory server or creating internal Anzo users, see the following topics:

- Adding Directory Users and Groups to Anzo
- Creating an Internal Anzo User

Roles

Anzo is configured with predefined roles. You can create new roles and disregard the predefined roles, remove the predefined roles, or add your groups to the predefined roles and modify the assigned permissions as needed.

For details about the default roles and instructions on creating new roles, see the following topics:

- Predefined Anzo Roles and Permissions
- Creating and Managing Roles

Permissions

The way you give a role access to the Anzo applications and particular functions in those applications is to assign permissions to the role. All permissions are predefined in Anzo. Custom permissions cannot be created, and the predefined permissions cannot be deleted.

For details about all of the permissions, see the following topic:

- Role Permissions Reference

For an overview of the data access management concepts, see Artifact Access Control Concepts.

Related Topics

Artifact Access Control Concepts

Connecting to a Directory Server

Adding Directory Users and Groups to Anzo

Connecting to an SSO Provider

Creating and Managing Roles

Creating an Internal Anzo User

Predefined Anzo Roles and Permissions

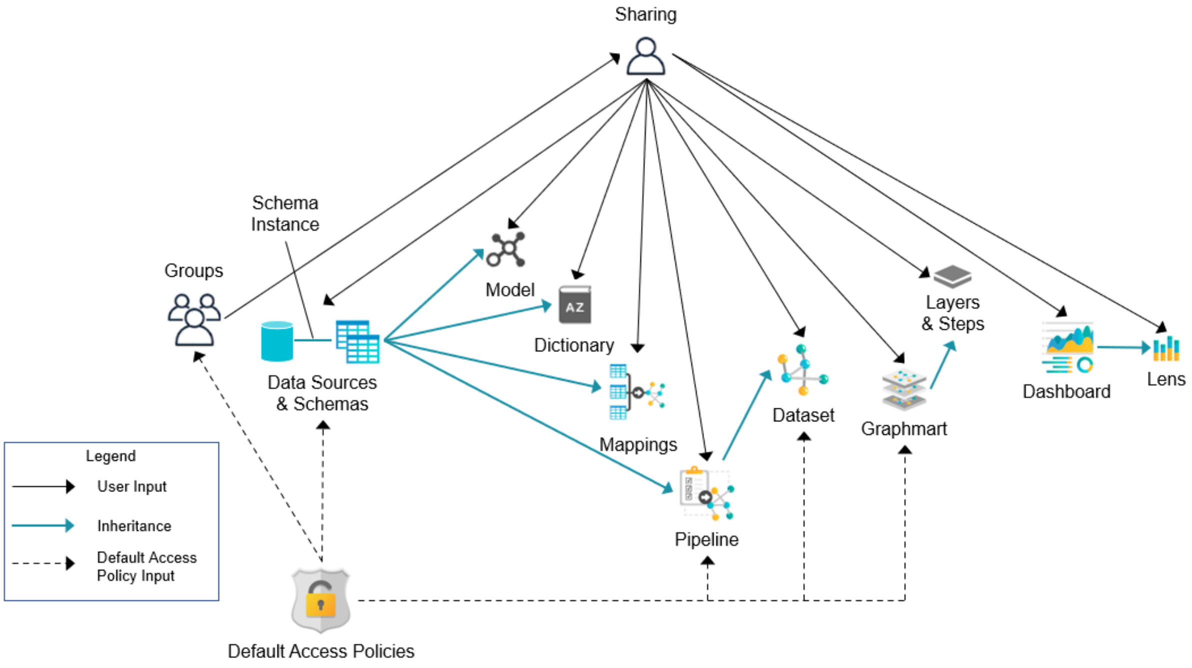
Role Permissions Reference

Artifact Access Control Concepts

The implementation of artifact and data access control in Anzo is an aggregation of three mechanisms:

1. **Default Access Policies**: These are the base permissions that are applied to artifacts by default when they are created. For most types of artifacts, the access control that is supplied by a Default Access Policy is augmented by the other two access control mechanisms.
2. **Permission Inheritance**: To facilitate common workflows, the Anzo application applies logic so that artifacts in the same workflow inherit the same permissions. For example, when a user creates a Data Source and uses the **Ingest** workflow to onboard the data, the generated Model, Pipeline, and Mapping artifacts inherit their permissions from the Data Source. Once the pipeline is published, the resulting Dataset inherits the permissions from the Pipeline. This permission inheritance is applied in addition to the applicable Default Access Policy.
3. **Sharing**: An artifact's creator can also share access to their artifact with other users or groups. When an artifact is shared, those user-configured permissions are applied in addition to any permissions that were inherited.

The following diagram illustrates the above concepts. Details about the processes and components depicted in the diagram are provided in the sections below.



Default Access Policies

Default Access Policies are the security policies that are applied by default to the artifacts that belong to a particular system **registry** (see Registries below). Default Access Policies are the base permissions that get assigned when an artifact is created—before any other access control logic (e.g., Permission Inheritance) is applied. Any artifact-level logic that is applied by Anzo or configured from the **Sharing** tab in the Anzo application augments the permissions that were supplied by the Default Access Policy.

For more information about Default Access Policies, see the following topic:

- Managing Default Access Policies

Registries

A registry is a system-level graph that stores metadata about artifacts of the same type. For example, a Data Sources Registry stores metadata about all of the Data Source and Schema artifacts, and an Ontology Registry stores metadata about all of the Data Model artifacts. Like onboarded data, registries are stored and managed as RDF named graphs according to system ontologies.

> **Important**
> Aside from changing the Default Access Policy for a registry, do not make additional modifications to registries. Changing or removing a registry can irreparably damage your Anzo server.

Permission Inheritance

The concept of inheritance is fundamental to the implementation of access control in Anzo. Inheritance allows related entities to share permissions with each other, making access easier to manage collectively, and ensuring that users have the appropriate access to each of the dependent artifacts that are crucial to their workflow. The following subsections describe the relationships and inheritance rules for each type of artifact.

- Data Sources & Schemas
- Ingest Workflow
- Graphmarts
- Structured Pipelines
- Unstructured Pipelines
- Metadata Dictionaries
- Users and Roles
- Role Permissions and Registries

Data Sources & Schemas

Data Sources and Schemas have a fundamental relationship since Schemas are imported from Data Sources and, in a sense, belong to them. Because a Data Source can have more than one Schema and the Schemas can be managed independently, Data Sources and Schemas exist as separate artifacts in Anzo. However, because of their implicit relationship, Anzo uses inheritance to facilitate users' interaction with Data Sources and the Schemas created from them.

If Anzo did not apply inheritance, a user who shares a Data Source would have to remember to add the new user to the data source *and* navigate to each related schema and add the new user there as well. Keeping permissions in sync manually presents a big challenge that is curtailed by applying inheritance.

To summarize the inheritance rules for Data sources and Schemas:

- Schemas inherit from the Data Source from which they were imported.
- Schema instances, which link Schemas to their Data Source, inherit from both the Schema and the Data Source.

Ingest Workflow

A primary workflow in Anzo is to create a new data source and then use the **Ingest** workflow (sometimes referred to as "auto-ingest") to generate all of the artifacts that are needed onboard the data and create the corresponding graph Dataset in Anzo. Artifacts created from the Ingest workflow inherit their permissions from the original Data Source.

If Anzo did not apply this inheritance, a user who wanted to share the Dataset that was derived from a Data Source would need to manually edit permissions for every artifact in the workflow: Model, Mappings, and Pipeline.

To summarize the inheritance rules for the Ingest workflow:

- Models generated by the Ingest workflow inherit permissions from the Data Source.
- Mappings generated by the Ingest workflow inherit permissions from the Data Source.
- Pipelines generated by the Ingest workflow inherit permissions from the Data Source.

> **Note**
> In rare cases when inheritance rules do not apply to artifacts, such as if a user manually creates a Mapping outside of the Ingest workflow, the SDI Registry Default Access Policy would supply the permissions for that Mapping until permissions are configured from the Mapping's **Sharing** tab.

Graphmarts

When a user creates a Graphmart, the Graphmart is assigned permissions according to the Graphmarts Registry Default Access Policy. Graphmarts contain Data Layers that describe and group the transformations that take place as the knowledge graph is generated. Since Data Layers are created in the context of a Graphmart, they inherit their permissions from the Graphmart by default.

If Anzo did not apply this inheritance, a user who wanted to share a Graphmart would have to remember to configure each newly created Data Layer to assign permissions that match the Graphmart's permissions. Otherwise someone who had access to the Graphmart would not be able to view or edit its layers and steps.

To summarize the inheritance rules for Graphmarts:

- Graphmarts inherit permissions from the Graphmarts Registry Default Access Policy.
- Data Layers and Steps created in a Graphmart inherit from the Graphmart.

For more information about graphmart permissions, see Sharing Access to Graphmarts in the User Guide.

Structured Pipelines

When a Structured Pipeline is published, it creates a Dataset. Since the most common data ingestion workflow is for a user to introduce a Data Source and then ingest the data into a Dataset by running a Pipeline, Datasets created from a Pipeline inherit their permissions from the Pipeline. If Anzo did not apply this inheritance, a user who has access to a Pipeline might lose the ability to see its output if the Pipeline happened to have been run by someone else first, for example.

To summarize the inheritance rules for Structured Pipelines:

- Datasets created from Structured Pipeline runs inherit from the Pipeline.
- Datasets created from auto-generated structured pipelines inherit from the original Data Source that was used to generate the Structured Pipeline.

Unstructured Pipelines

As with structured pipelines, running an Unstructured Pipeline produces a Dataset. For similar reasons, the output unstructured Dataset inherits from the Unstructured Pipeline. Additionally, each Unstructured Pipeline run produces a status dataset that is specific to the pipeline's execution. Since these status datasets are implicitly related to the Unstructured Pipeline, they inherit permissions from the pipeline.

To summarize the inheritance rules for Unstructured Pipelines:

- Datasets created from Unstructured Pipeline runs inherit from the corresponding Unstructured Pipeline.
- Pipeline status datasets inherit from the related unstructured pipeline. From an end user's perspective, this relates to the status information that is displayed in the Unstructured Pipeline user interface.

Metadata Dictionaries

Users can create Metadata Dictionaries from specific Data Sources. Because the dictionary is directly related to the origin Data Source, Metadata Dictionaries inherit their permissions from the corresponding Data Source. If one dictionary is used for multiple sources, the dictionary inherits the superset of permissions from the origin Data Sources.

To summarize the inheritance rules for Metadata Dictionaries:

- Dictionaries generated from Data Sources inherit permissions from the Data Source.
- Dictionaries that link concepts from multiple Data Sources inherit from all corresponding Data Sources.

Users and Roles

Users and roles are typically managed by administrators as a collective group. There are not clear use cases for a given user to manage some user and role accounts but not others. The expectation is that users who have the **Manage Users, Groups, and Roles** permission should be able to manage all users and roles, not just a subset of them.

To accomplish the above expectation, all users inherit permissions from one system registry, the **Role and Permissions Registry**. If user and role permissions were not centralized, there could be circumstances where one user creates a new user or role in Anzo and other users cannot see or edit that account even if they belong to a role that has the Manage Users, Groups, and Roles permission. Also if the original user or role creator had the Manage Users, Groups, and Roles permission revoked, they may retain control over the accounts they created when they had the ability to do so.

To summarize the inheritance rules for users and roles:

- Anyone who has the **Manage Users, Groups, and Roles** permission has the **Admin** level of access to all users, groups, and roles.
- The **Everyone** role has **View** access to all users, groups, and roles so that they can share artifacts with other users and groups.

Role Permissions and Registries

Access to certain registries is mapped to specific Anzo permissions. This is helpful when artifacts that are added to a registry inherit their permissions from the registry itself rather than another artifact, such as with Users and Roles. When users have a permission that grants them access to a registry, that means they can see all artifacts that belong to that registry.

The list below describes the registry access that is controlled by a permission.

- Access to the Role and Permissions Registry is granted by the **Manage Users, Groups, and Role** permission.

For more information about the Anzo permissions, see Role Permissions Reference.

Sharing

Artifacts can be shared with other users and groups from the artifact's **Sharing** tab in the Anzo application. When an artifact is shared, those user-defined permissions are added to the set of permissions that came from the Default Access Policy for the related registry as well as the permission inheritance that is applied by Anzo.

For details about artifact sharing, see Sharing Access to Artifacts in the User Guide.

Related Topics

User Management Concepts

Managing Default Access Policies

Role Permissions Reference

Connecting to a Directory Server

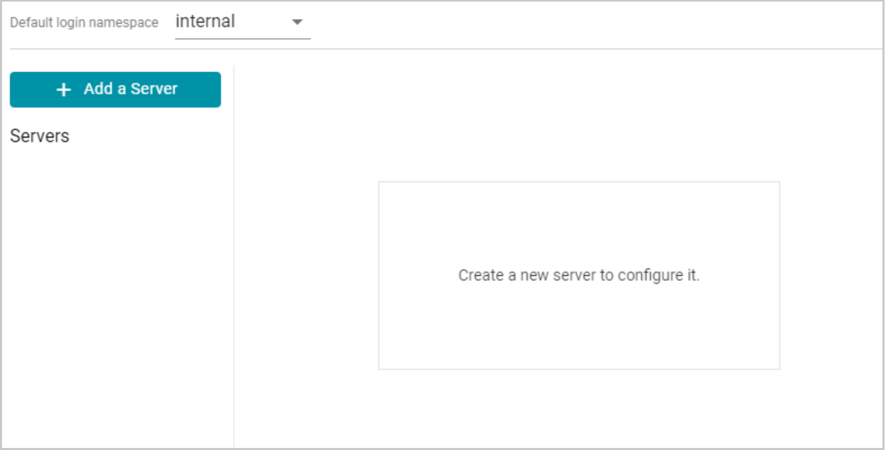
This section provides instructions for connecting to a directory server and mapping the user and group configuration to Anzo so that Anzo can leverage the users and groups from the server.

- Connect to the Directory Server
- Map Users to Anzo
- Map Groups to Anzo

Connect to the Directory Server

Follow the steps below to create a connection between Anzo and your directory server.

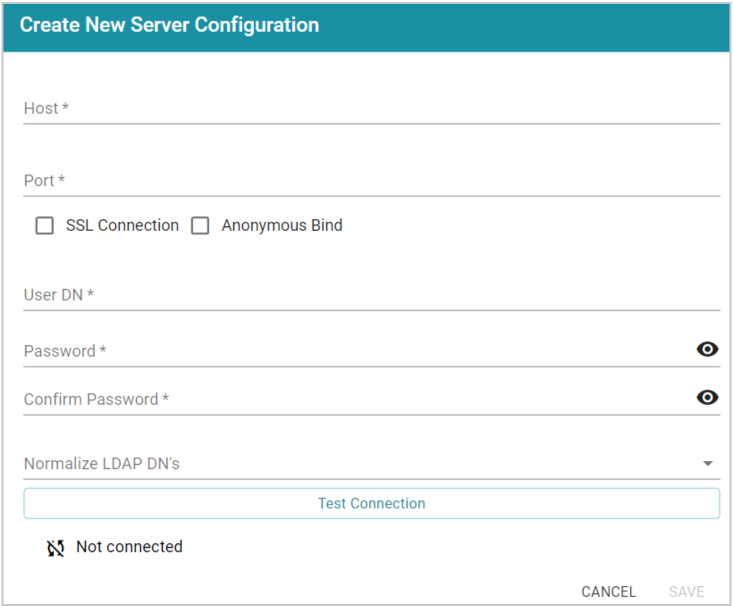
1. In the Administration application, expand the **User Management** menu and click **Directory**. Anzo displays the Directory screen. For example:



Default login namespace internal

Create a new server to configure it.

2. On the Directory screen, click the **Add a Server** button. Anzo displays the Create New Server Configuration screen.

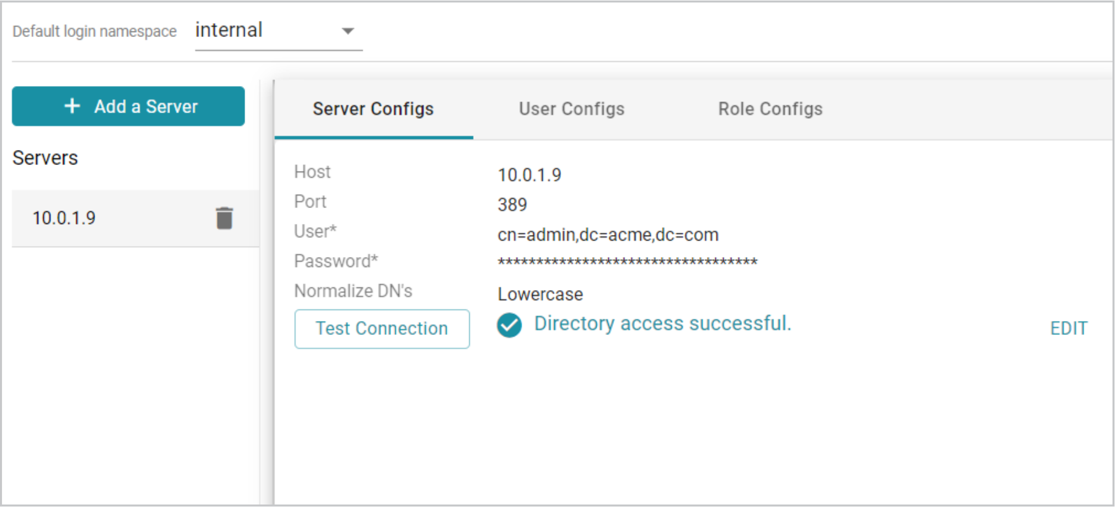


Not connected

3. Enter the connection details for the server:

   - **Host**: The host name or IP address for the directory server.
   - **Port**: The port to use to connect to the directory server.
   - **SSL Connection**: Indicates whether the directory server uses an SSL connection. Select the **SSL Connection** checkbox to enable the SSL connection. If you use SSL, make sure that you load the directory server's certificate to the Anzo trust store. See Adding a Certificate to the Anzo Trust Store for instructions.
   - **Anonymous Bind**: This option indicates whether you want Anzo to connect to the directory server anonymously. To avoid Anzo login problems when enabling this option, make sure the directory server allows anonymous binding and searches when bound anonymously. Select the **Anonymous Bind** checkbox to enable anonymous binding.
   - **User DN**: The full distinguished name of the account that Anzo will bind against to perform searches on the directory server.
   - **Password** and **Confirm Password**: The password for the User DN.
   - **Normalize LDAP DNs**: To ensure that duplicate user accounts are not created in Anzo if an LDAP distinguished name has both a lowercase and uppercase version, you can configure the system to normalize distinguished name strings so that values that differ only in capitalization are treated as the same value. If you do not want distinguished names to be normalized, leave the field blank or select **None**. To normalize distinguished names to lowercase, select **Lowercase**, or select **Uppercase** if you want names to be normalized to uppercase.

4. Anzo attempts to connect to the server automatically. If the connection fails, make sure that you entered the correct connection details. You can also click **Test Connection** to check if Anzo can connect to the server.
5. Click **Save** to save the server configuration and return to the Directory screen. The new server configuration is selected on the screen. For example:



Default login namespace internal

Servers

10.0.1.9

Host	10.0.1.9
Port	389
User*	cn=admin,dc=acme,dc=com
Password*	***********************************
Normalize DN's	Lowercase

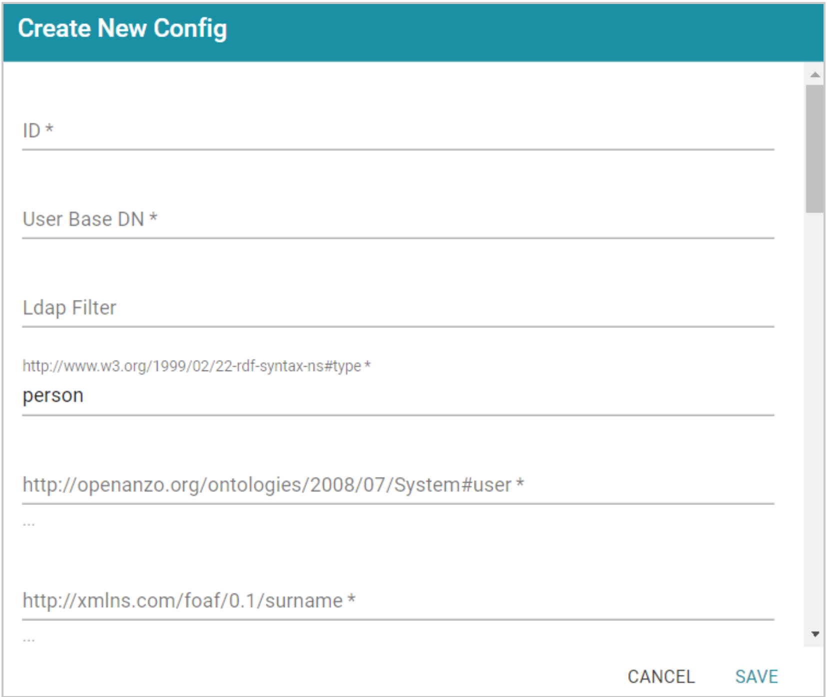
Directory access successful.

Once the connection to the server is established, create a user configuration for mapping directory users to Anzo. See Map Users to Anzo below for instructions.

Map Users to Anzo

Follow the steps below to create a user configuration by supplying the mapping the attributes to use to sync users with Anzo.

1. On the Directory screen, click the **User Configs** tab. Then click the **Create New User Config** button. Anzo displays the Create New Config dialog box.



Create New Config

ID *

User Base DN *

Ldap Filter

http://www.w3.org/1999/02/22-rdf-syntax-ns#type *
person

http://openanzo.org/ontologies/2008/07/System#user *
...

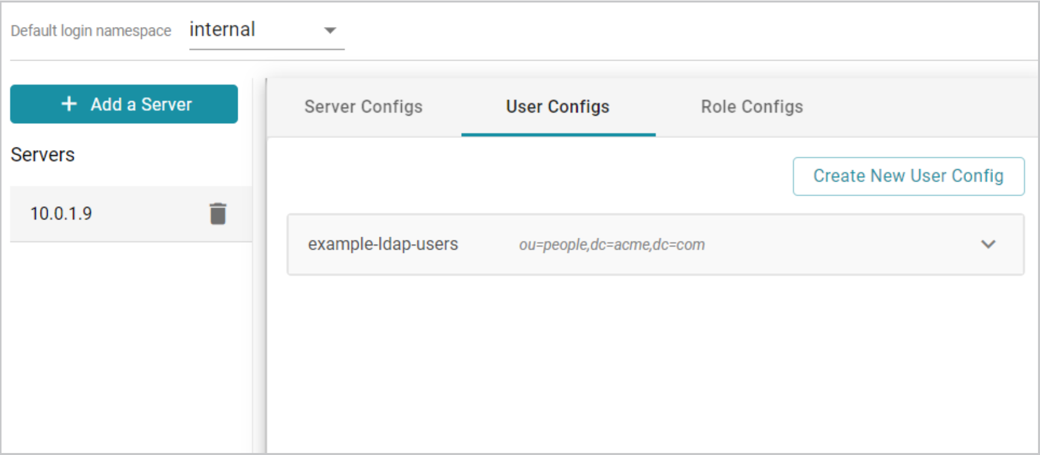
http://xmlns.com/foaf/0.1/surname *
...

2. Complete the following required fields and specify the optional values as desired. Each time you map an attribute, Anzo displays some samples of the values it retrieves for that attribute. If the specified attribute does not match an attribute in the system, Anzo displays an "LDAP Attribute unavailable" message.
   - **ID**: **Required** setting that defines the unique name for this user configuration. Anzo uses this value as a namespace for usernames in case you connect to multiple directories with conflicting names.
   - **User Base DN**: **Required** setting that specifies the LDAP distinguished name.
   - **LDAP Filter**: The optional LDAP filter to apply when searching for users (usually left blank).
   - **http://www.w3.org/1999/02/22-rdf-syntax-ns#type**: **Required** setting that specifies the LDAP class of the type of accounts that should be logged on. Typically **person**.
   - **http://openanzo.org/ontologies/2008/07/System#user**: **Required** setting that specifies the LDAP attribute that contains user login information. Typically **uid**.
   - **http://xmlns.com/foaf/0.1/surname**: **Required** setting that specifies the LDAP attribute that contains users' surnames. Typically **sn**.
   - **http://xmlns.com/foaf/0.1/name**: **Required** setting that specifies the LDAP attribute that contains users' full names. Typically **cn**.
   - **http://xmlns.com/foaf/0.1/givenname**: **Required** setting that specifies the LDAP attribute that contains users' first names. Typically **givenName**.

- **http://xmlns.com/foaf/0.1/title**: Optional value that specifies the LDAP attribute that contains users' job titles. Typically **title**.
- **http://www.w3.org/2003/06/sw-vocab-status/ns#term-status**: Optional value that specifies the status at the level of terms.
- **http://xmlns.com/wot/0.1/src_assurance**: Optional value that specifies the source for Assured Replication.
- **http://xmlns.com/foaf/0.1/phone**: Optional value that specifies the LDAP attribute that contains user phone numbers. Typically **telephoneNumber**.
- **http://xmlns.com/foaf/0.1/mbox**: Optional value that specifies the LDAP attribute that contains users' email addresses. Typically **mail**.
- **http://openanzo.org/ontologies/2008/07/Anzo#location**: Optional value that specifies the LDAP attribute that contains user location information.
- **http://openanzo.org/ontologies/2008/07/Anzo#isInternalUser**: Optional boolean value that indicates whether users are Anzo internally managed users.
- **http://xmlns.com/foaf/0.1/img**: Optional value that specifies the LDAP attribute that contains images for users.
- **http://purl.org/dc/elements/1.1/description**: Optional value that specifies the LDAP attribute that contains user descriptions. Typically **description**.
- **http://openanzo.org/ontologies/2008/07/Anzo#defaultGroup**: Optional value that specifies the LDAP attribute that contains the value of users' Anzo Default Group assignment.
- **http://openanzo.org/ontologies/2008/07/Anzo#companyDepartment**: Optional value that specifies the LDAP attribute that contains user department information. Typically **department**.
- **http://xmlns.com/wot/0.1/assurance**: Optional boolean value that indicates whether Assured Replication is enabled.

3. When you have finished mapping attributes, click **Save** to save the user configuration.

The new user configuration is added to the system and Anzo returns to the Directory screen, which shows the newly created configuration. For example:

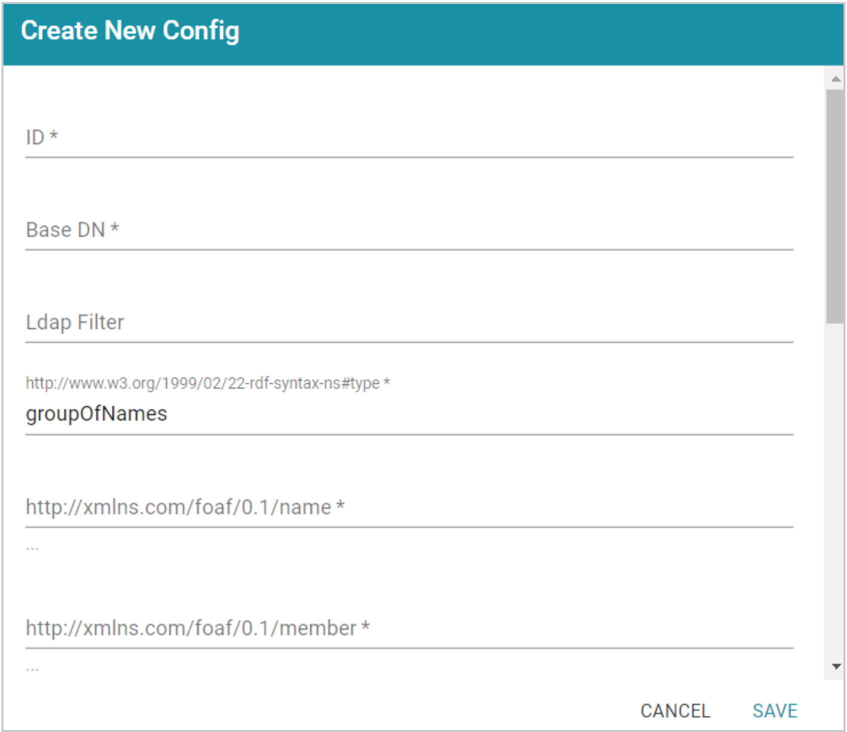


Once the user configuration is complete, create a role configuration for mapping directory groups to Anzo. See Map Groups to Anzo below for instructions.

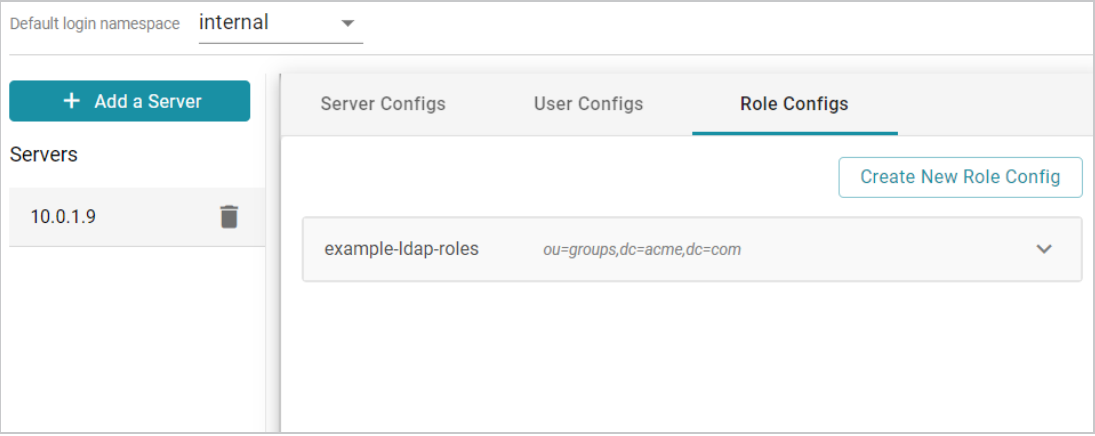
Map Groups to Anzo

Follow the steps below to create a role configuration by supplying the mapping the attributes to use to sync groups with Anzo.

1. On the Directory screen, click the **Role Configs** tab. Then click the **Create New Role Config** button. Anzo displays the Create New Config dialog box.

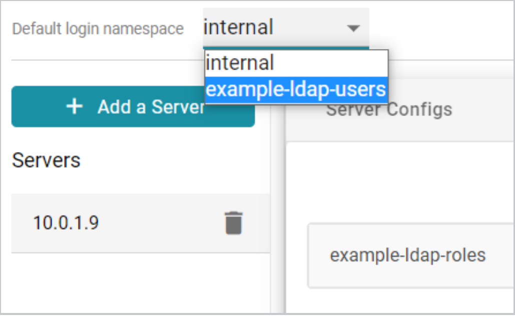

Create New Config
ID *
Base DN *
Ldap Filter
http://www.w3.org/1999/02/22-rdf-syntax-ns#type *
groupOfNames
http://xmlns.com/foaf/0.1/name *
...
http://xmlns.com/foaf/0.1/member *
...
CANCEL
SAVE

2. Complete the following required fields and specify the optional values as desired. Each time you map an attribute, Anzo displays some samples of the values it retrieves for that attribute. If the specified attribute does not match an attribute in the system, Anzo displays an "LDAP Attribute unavailable" message.
    - **ID**: **Required** setting that defines the unique name for this role configuration.
    - **Base DN**: **Required** setting that specifies the LDAP distinguished name that contains all of the system roles.
    - **LDAP Filter**: The optional LDAP filter to apply when searching for roles (usually left blank).
    - **http://www.w3.org/1999/02/22-rdf-syntax-ns#type**: **Required** setting that specifies the group object class of the type of roles. Typically **groupOfNames**.
    - **http://xmlns.com/foaf/0.1/name**: **Required** setting that specifies the LDAP attribute that contains the names of the roles.
    - **http://xmlns.com/foaf/0.1/member**: **Required** setting that specifies the LDAP attribute that contains common member attributes. Typically **member** or **uniqueMember**.
    - **http://openanzo.org/ontologies/2008/07/Anzo#usedBy**: Optional value that specifies how the role is used by Anzo.
    - **http://www.w3.org/2003/06/sw-vocab-status/ns#term-status**: Optional value that specifies the status at the level of terms.
    - **http://xmlns.com/wot/0.1/src_assurance**: Optional value that specifies the source for Assured Replication.
    - **http://openanzo.org/ontologies/2008/07/Anzo#permission**: Optional value that specifies the LDAP attribute that contains the Anzo permissions to assign to the roles.
    - **http://purl.org/dc/elements/1.1/description**: Optional value that specifies the LDAP attribute that contains role descriptions.
    - **http://purl.org/dc/elements/1.1/date**: Optional value that specifies the LDAP attribute that contains role dates.
    - **http://xmlns.com/wot/0.1/assurance**: Optional boolean value that indicates whether Assured Replication is enabled.
3. Click **Save** to save the role configuration. The new role configuration is added to the system and Anzo returns to the Directory screen, which shows the newly created configuration. For example:


Default login namespace: internal

example-ldap-roles ou=groups,dc=acme,dc=com

4. The last step in configuring the server is to designate the default login namespace to use if users do not fully qualify their username with the @suffix when they log in to Anzo. To set the namespace, click the **Default login namespace** drop-down list at the top of the screen and select the namespace for the directory server. It will be displayed as the ID that was specified when you set up the user configuration. The "Internal" namespace that is also listed is the internal Anzo LDAP server for local users. For example:


Default login namespace: internal

Once you have connected the directory server to Anzo and created user and role configurations, the next step is to add the directory users and groups to Anzo. See Adding Directory Users and Groups to Anzo for instructions.

> **Tip**
> You can also set up single-sign on access to Anzo. See Connecting to an SSO Provider for instructions.

Related Topics

User Management and Access Control Concepts

Adding Directory Users and Groups to Anzo

Connecting to an SSO Provider

Adding Directory Users and Groups to Anzo

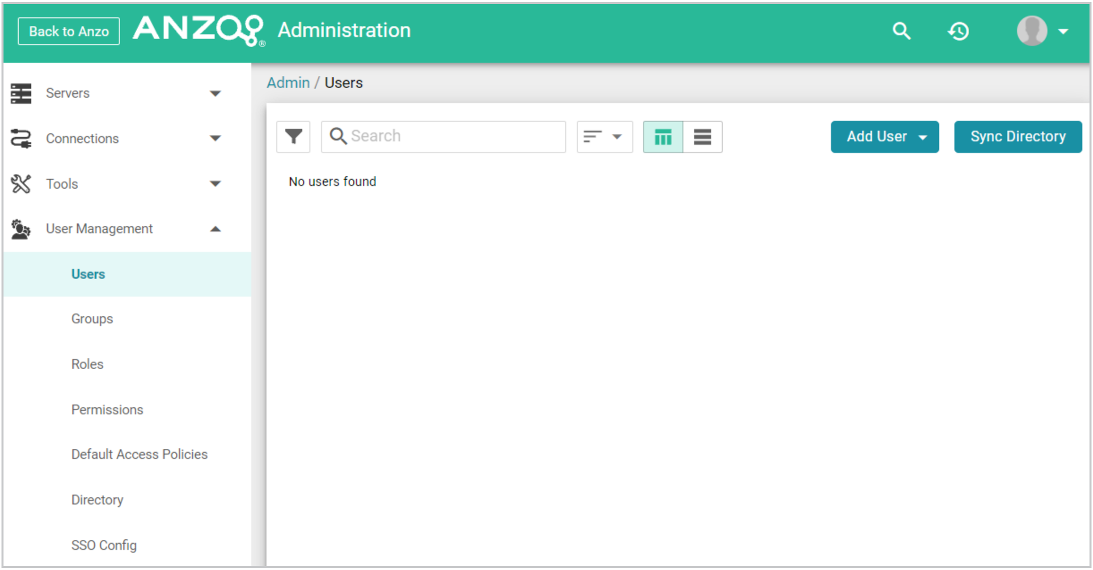
After you connect to a central directory server, you have multiple options for how LDAP users gain access to Anzo. Some organizations retrieve the LDAP users and groups from the server and add them to Anzo. An Anzo administrator then manages role and license assignment in Anzo. Other organizations pre-define LDAP-to-Anzo role configurations and mappings so that users are automatically assigned an Anzo license and can log in to Anzo as soon as the LDAP administrator adds them to the appropriate LDAP role. With this option, no action needs to be taken in Anzo once the directory server is connected and user and role mappings are configured.

This topic provides instructions for adding directory users and groups to Anzo. For instructions on setting up self-authorization for directory users so that they can log into Anzo and automatically become licensed after being added to the appropriate LDAP group, see Enabling Self-Authorization for Directory Users.

- Adding Directory Users to Anzo
- Adding Directory Groups to Anzo

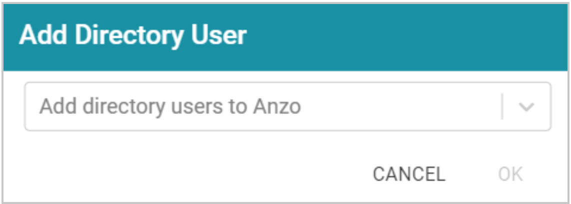
Adding Directory Users to Anzo

1. To add directory users to Anzo, select **Users** from the **User Management** menu in the Administration application. The Users screen is displayed. For example:

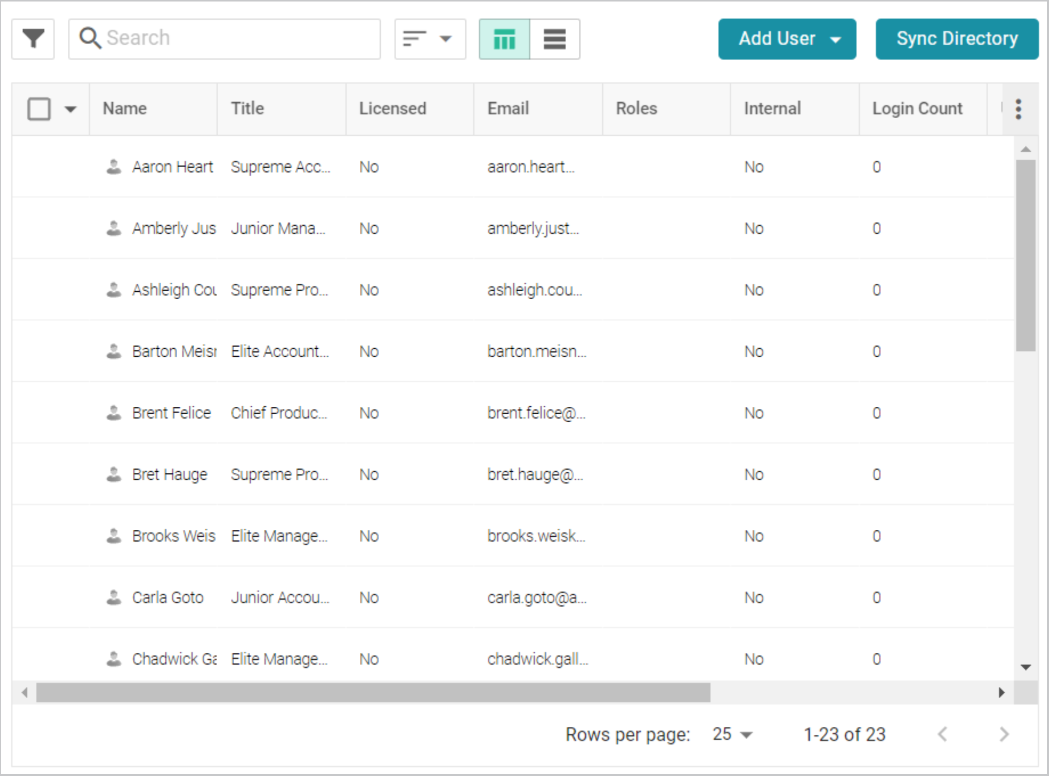


No users found

2. Click the **Add User** button and select **Add Directory Users**. The Add Directory User dialog box is displayed:



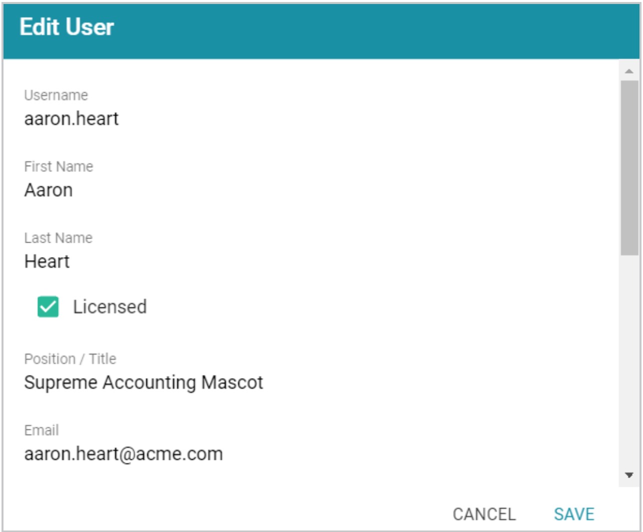
3. Click the **Add directory users to Anzo** drop-down list, and select each user to add to Anzo. Repeat this step for all of the users that you want to add.
4. When you have finished adding users, click **OK** to return to the Users screen. For example:



Name	Title	Licensed	Email	Roles	Internal	Login Count
Aaron Heart	Supreme Acc...	No	aaron.heart...		No	0
Amberly Jus	Junior Mana...	No	amberly.just...		No	0
Ashleigh Cou	Supreme Pro...	No	ashleigh.cou...		No	0
Barton Meisı	Elite Account...	No	barton.meisn...		No	0
Brent Felice	Chief Produc...	No	brent.felice@...		No	0
Bret Hauge	Supreme Pro...	No	bret.hauge@...		No	0
Brooks Weis	Elite Manage...	No	brooks.weisk...		No	0
Carla Goto	Junior Accou...	No	carla.goto@a...		No	0
Chadwick Gε	Elite Manage...	No	chadwick.gall...		No	0

> **Note**
>
> In order for the new users to be able to log in to Anzo, they must be **Licensed** users. Complete the next step to designate licensed users.

5. The last step in the process is to configure the **Licensed** users. If you want a user to be able to log in to Anzo, they must be specified as a licensed user. To designate a user as licensed, open the Edit User dialog box by clicking a user's name in the Users list. In the dialog box, select the **Licensed** checkbox and click **Save**. For example:



Edit User

Username
aaron.heart

First Name
Aaron

Last Name
Heart

☑ Licensed

Position / Title
Supreme Accounting Mascot

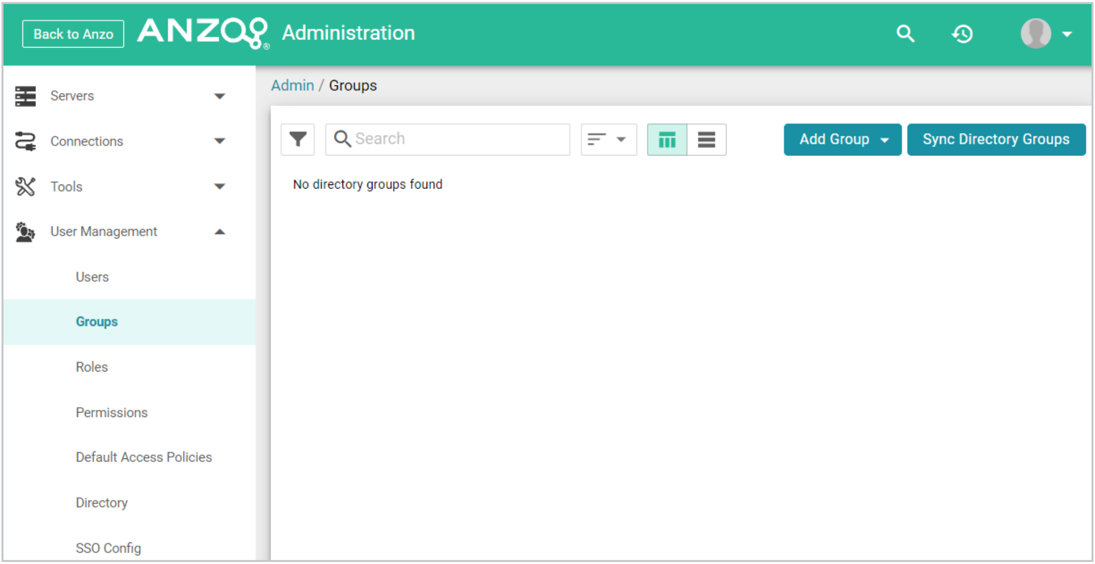
Email
aaron.heart@acme.com

Repeat this step for all of the users who should be licensed.

For instructions on adding groups to Anzo, proceed to Adding Directory Groups to Anzo below.

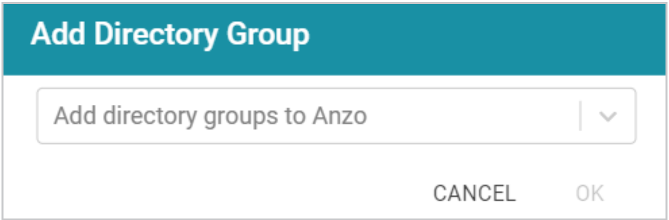
Adding Directory Groups to Anzo

1. To add directory groups to Anzo, select **Groups** from the **User Management** menu in the Administration application. The Groups screen is displayed. For example:

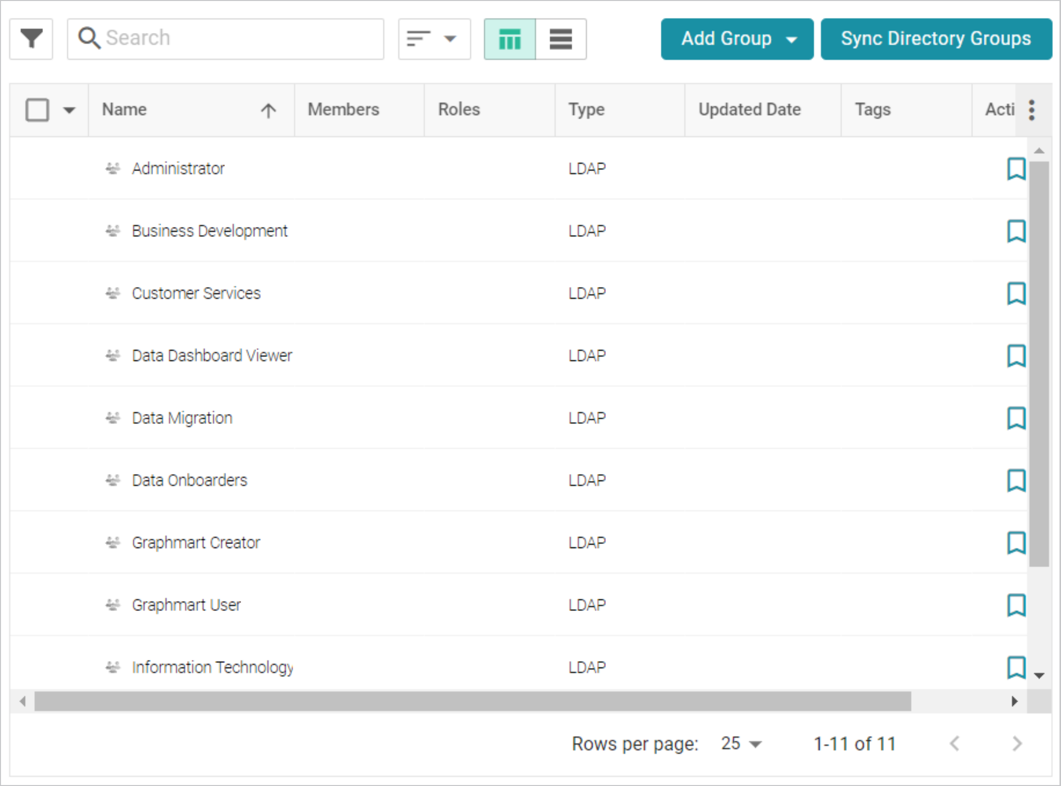


No directory groups found

2. Click the **Add Group** button and select **LDAP Directory Group**. The Add Directory Group dialog box is displayed:



3. Click the **Add directory groups to Anzo** drop-down list, and select each group to add to Anzo. Repeat this step for all of the groups that you want to add.
4. When you have finished adding groups, click **OK** to return to the Groups screen. For example:



Name	Members	Roles	Type	Updated Date	Tags	Acti
Administrator			LDAP			
Business Development			LDAP			
Customer Services			LDAP			
Data Dashboard Viewer			LDAP			
Data Migration			LDAP			
Data Onboarders			LDAP			
Graphmart Creator			LDAP			
Graphmart User			LDAP			
Information Technology			LDAP			

Rows per page: 25 1-11 of 11

Now that the users and groups from the directory server are available in Anzo, the next step is to associate the groups with Anzo roles. Roles are used to grant access to the Anzo applications and the functionality in those applications. See Creating and Managing Roles for instructions.

Related Topics

Enabling Self-Authorization for Directory Users

User Management and Access Control Concepts

Connecting to a Directory Server

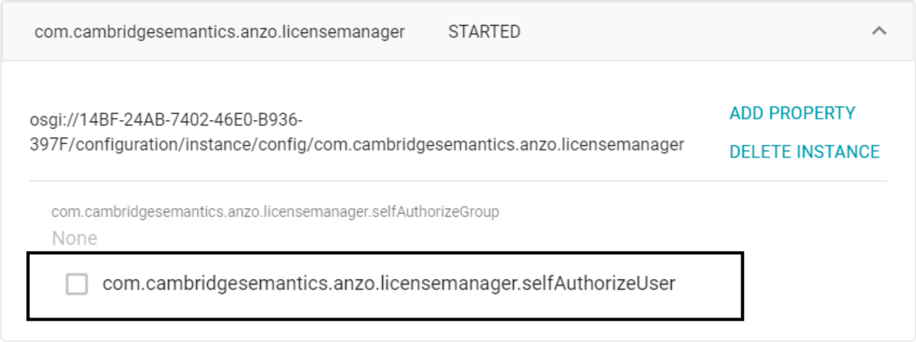
Creating and Managing Roles

Enabling Self-Authorization for Directory Users

In order to log in to Anzo, a user must be a **Licensed** user. If you defined LDAP-to-Anzo role configurations and mappings so that you can manage all permissions in the directory server without retrieving the user and group accounts and adding them to Anzo, you can configure Anzo to automatically license those users as they log in. Follow the instructions below to enable self-authorization.

> **Note**
> When deciding whether to enable self-authorization at all or whether to limit it to certain LDAP groups, consider the number of users who will access Anzo and the number of users allowed by your Anzo license. Your Cambridge Semantics Customer Success manager can help determine whether to enable the feature if you have questions.

1. In the Administration application, expand the **Servers** menu and click **Advanced Configuration**. Click **I understand and accept the risk**.
2. Search for the **Anzo License and Entitlement Manager** bundle and view its details.
3. Click the **Services** tab and expand **com.cambridgesemantics.anzo.licensemanager**.
4. Find the **com.cambridgesemantics.anzo.licensemanager.selfAuthorizeUser** property (shown below).



com.cambridgesemantics.anzo.licensemanager STARTED

osgi://14BF-24AB-7402-46E0-B936-397F/configuration/instance/config/com.cambridgesemantics.anzo.licensemanager

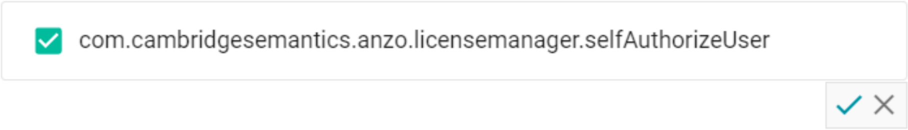
ADD PROPERTY

DELETE INSTANCE

com.cambridgesemantics.anzo.licensemanager.selfAuthorizeGroup
None

□ com.cambridgesemantics.anzo.licensemanager.selfAuthorizeUser

5. Click the property to make it editable, and then select the checkbox to enable it.



☑ com.cambridgesemantics.anzo.licensemanager.selfAuthorizeUser

✓ ✕

6. Click the checkmark icon (✓) to save the change.

7. If you want to limit the ability to self-authorize to a certain LDAP group, click the **com.cambridgesemantics.anzo.licensemanager.selfAuthorizeGroup** property to make it editable, and then specify the group name to include.
8. Click the checkmark icon (✓) to save the change.

Changes to the Anzo License and Entitlement Manager service take effect immediately. You do not need to restart Anzo or the service to apply the change.

Related Topics

User Management and Access Control Concepts

Connecting to a Directory Server

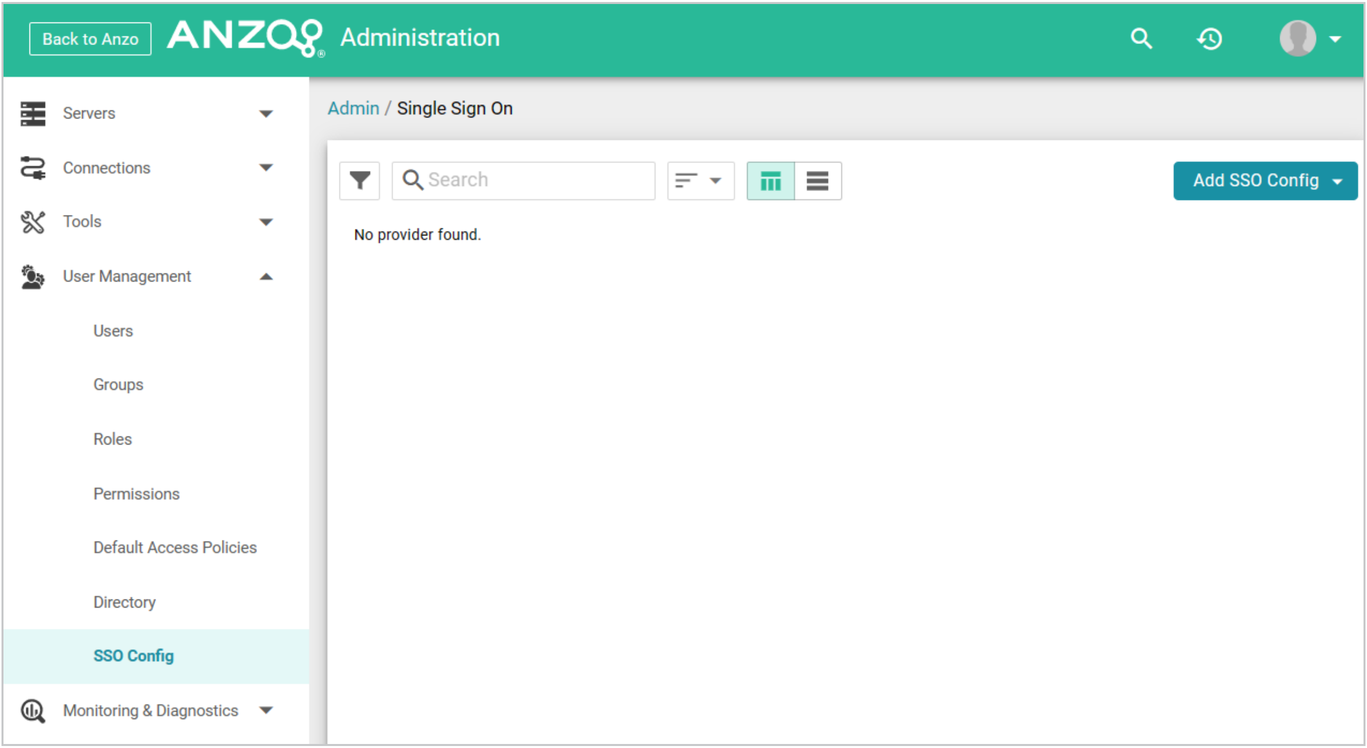
Connecting to an SSO Provider

This topic provides instructions for configuring Anzo to enable single sign-on (SSO) access using one of the following SSO providers:

- Direct and Indirect Basic
- Direct and Indirect Kerberos
- Facebook
- JSON Web Tokens (JWT) Header and Parameter
- OpenID Connect (OIDC)
- Security Assertion Markup Language (SAML)
- Google OpenID Connect (OIDC)

Follow the instructions below to add a provider.

1. In the Administration application, expand the **User Management** menu and click **SSO Config**. Anzo displays the Single Sign On screen, which lists any existing SSO providers. For example:



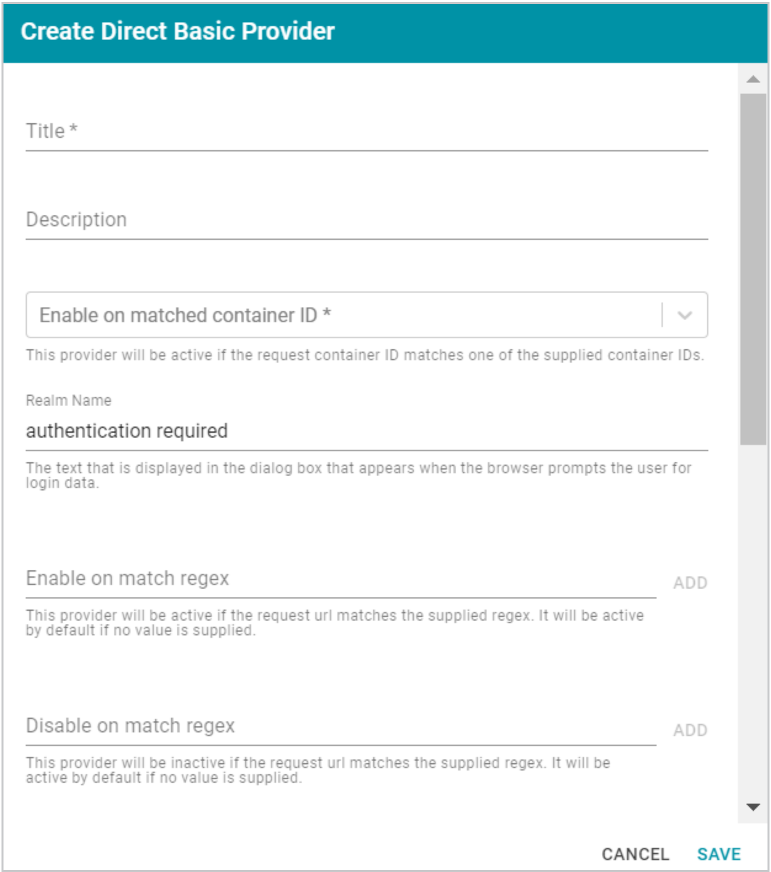
No provider found.

2. Click the **Add SSO Config** button and select the type of provider to configure. Anzo opens the Create dialog box for that provider. Complete the required fields and supply any of the relevant optional

information. The list below provides details about the properties for each provider.

Direct Basic Provider

This section describes the settings that are available on the Create Direct Basic Provider screen:



Create Direct Basic Provider

Title *

Description

Enable on matched container ID *

This provider will be active if the request container ID matches one of the supplied container IDs.

Realm Name

authentication required

The text that is displayed in the dialog box that appears when the browser prompts the user for login data.

Enable on match regex

This provider will be active if the request url matches the supplied regex. It will be active by default if no value is supplied.

Disable on match regex

This provider will be inactive if the request url matches the supplied regex. It will be active by default if no value is supplied.

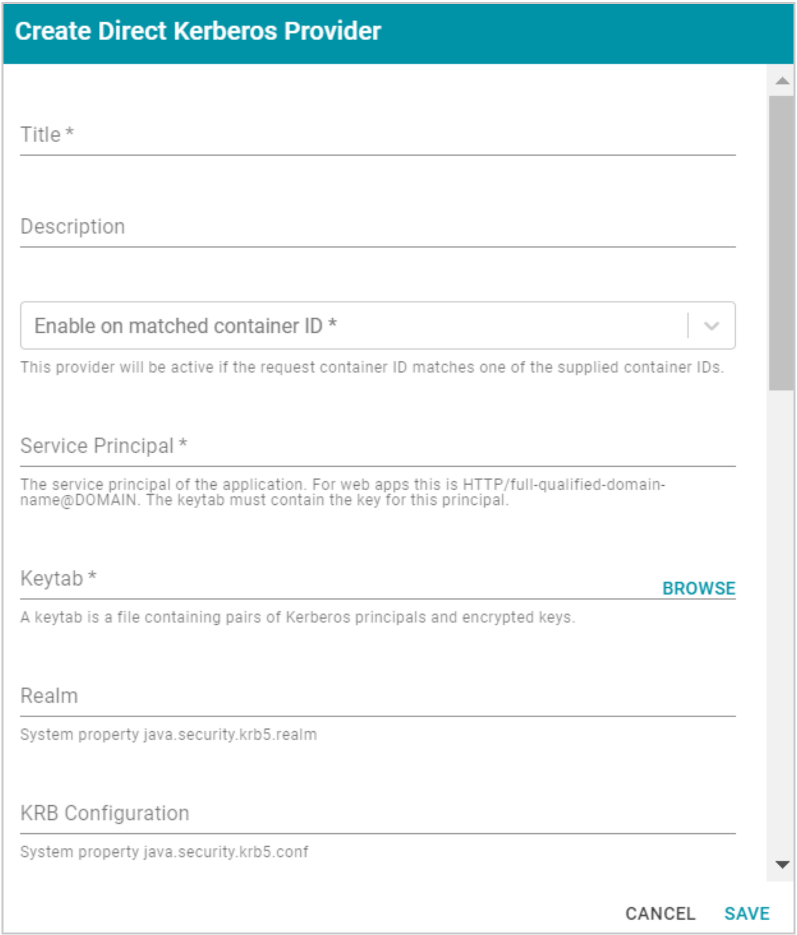
- **Title**: **Required** field that specifies the name for this provider configuration.
- **Description**: Optional field that provides a description for this provider configuration.
- **Enable on matched container ID**: **Required** field that lists the container ID(s) to match. This provider will be active if the request container ID matches one of the container IDs specified in this property. Click the field and select a container ID from the drop-down list. To specify multiple IDs, click the field again and select another value. To remove a container from the list, click the X on the right of the container name.
- **Realm Name**: Optional field that specifies the name of the security realm.
- **Enable on match regex**: Optional field that defines regular expression rules for matching request URLs to enable. To add a rule, type an expression in the field and

click **Add**. This provider will be active if the request URL matches any of the supplied expressions. If Enable on match regex is blank, the provider will be active by default.

- **Disable on match regex**: Optional field that defines regular expression rules for matching request URLs to disable. To add a rule, type an expression in the field and click **Add**. This provider will be inactive if the request URL matches any of the supplied expressions. If Disable on match regex is blank, the provider will be active by default.
- **Email Template regex**: If email was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between email addresses stored by the SSO provider and email addresses returned by the directory server.
- **Email Template Replacement**: Optional field that specifies a replacement email template to use if there are variations found by Email Template regex.
- **User Template regex**: If user was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between user names stored by the SSO provider and user names returned by the directory server.
- **User Template Replacement**: Optional field that specifies a replacement user template to use if there are variations found by User Template regex.
- **Use username directly**:
- **Skip CSRF check**: Optional property that specifies whether to perform a cross-site request forgery (CSRF) check.
- **LDAP domain**: Optional field that specifies the LDAP domain to use for user lookup.
- **LDAP email property**: Optional field that specifies the LDAP email property to use to find the associated user's dn. For example, `http://openanzo.org/ontologies/2008/07/Anzo#ldapEmailInfo`.

Direct Kerberos Provider

This section describes the settings that are available on the Create Direct Kerberos Provider screen:



Create Direct Kerberos Provider

Title *

Description

Enable on matched container ID *

This provider will be active if the request container ID matches one of the supplied container IDs.

Service Principal *

The service principal of the application. For web apps this is HTTP/full-qualified-domain-name@DOMAIN. The keytab must contain the key for this principal.

Keytab *

A keytab is a file containing pairs of Kerberos principals and encrypted keys.

Realm

System property java.security.krb5.realm

KRB Configuration

System property java.security.krb5.conf

- **Title**: **Required** field that specifies the name for this provider configuration.
- **Description**: Optional field that provides a description for this provider configuration.
- **Enable on matched container ID**: **Required** field that lists the container ID(s) to match. This provider will be active if the request container ID matches one of the container IDs specified in this property. Click the field and select a container ID from the drop-down list. To specify multiple IDs, click the field again and select another value. To remove a container from the list, click the X on the right of the container name.
- **Service Principal**: **Required** field that specifies the service and DNS name for the application. For authentication through the web browser, specify the service principal value in the following format:

  ```
  HTTP/fully_qualified_domain_name@domain
  ```

  For example, `HTTP/server.example.com@example.com`.

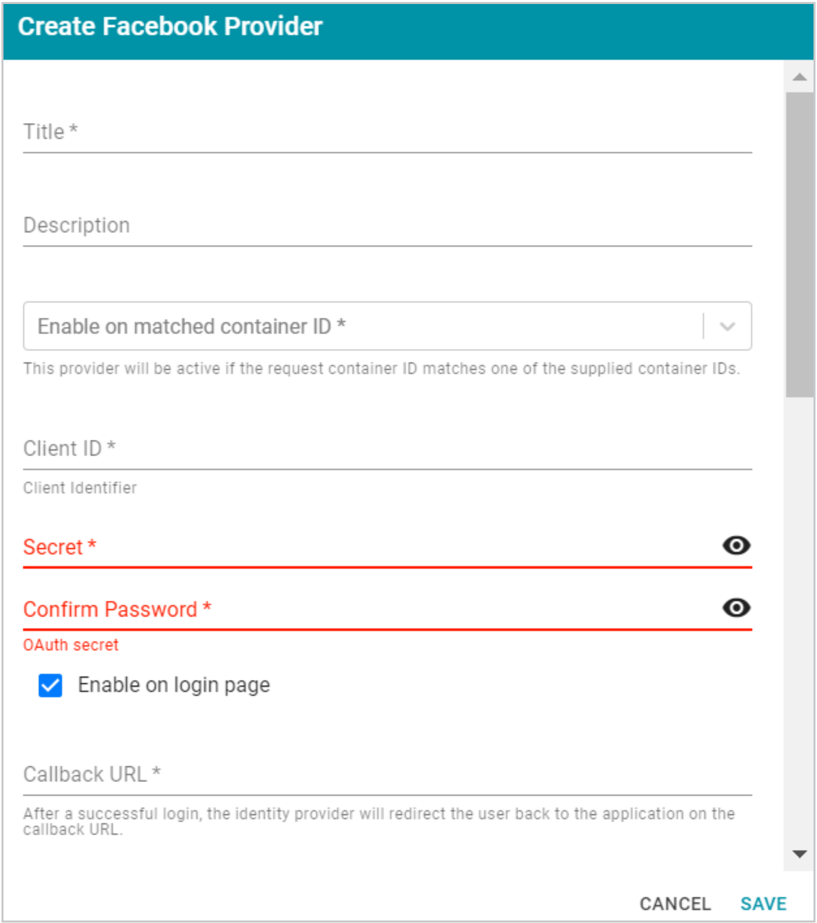
> **Note** The keytab file must contain the key for this principal.

- **Keytab**: **Required** field that specifies the .keytab file that lists the Kerberos principals and encrypted keys. Click the **Keytab** field to open the File Location dialog box and select the keytab file.
- **Realm**: Optional field that specifies the Kerberos realm that the service principal maps to.
- **KRB Configuration**: Optional field that specifies the path and file name for the krb5.conf file on the Kerberos instance. The default location is `/etc/krb5.conf`.
- **KDC**: Optional field that specifies the domain name for the Key Distribution Center.
- **Debug mode**: Optional field that specifies whether Kerberos debug logging is enabled for your provider.
- **Enable on match regex**: Optional field that defines regular expression rules for matching request URLs to enable. To add a rule, type an expression in the field and click **Add**. This provider will be active if the request URL matches any of the supplied expressions. If Enable on match regex is blank, the provider will be active by default.
- **Disable on match regex**: Optional field that defines regular expression rules for matching request URLs to disable. To add a rule, type an expression in the field and click **Add**. This provider will be inactive if the request URL matches any of the supplied expressions. If Disable on match regex is blank, the provider will be active by default.
- **Email Template regex**: If email was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between email addresses stored by the SSO provider and email addresses returned by the directory server.
- **Email Template Replacement**: Optional field that specifies a replacement email template to use if there are variations found by Email Template regex.
- **User Template regex**: If user was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between user names stored by the SSO provider and user names returned by the directory server.
- **User Template Replacement**: Optional field that specifies a replacement user template to use if there are variations found by User Template regex.

- **Use username directly**:
- **Skip CSRF check**: Optional property that specifies whether to perform a cross-site request forgery (CSRF) check.
- **LDAP domain**: Optional field that specifies the LDAP domain to use for user lookup.
- **LDAP email property**: Optional field that specifies the LDAP email property to use to find the associated user's dn. For example, `http://openanzo.org/ontologies/2008/07/Anzo#ldapEmailInfo`.

Facebook Provider

This section describes the settings that are available on the Create Facebook Provider screen:



Create Facebook Provider

Title *

Description

Enable on matched container ID *

This provider will be active if the request container ID matches one of the supplied container IDs.

Client ID *

Client Identifier

Secret *

Confirm Password *

OAuth secret

☑ Enable on login page

Callback URL *

After a successful login, the identity provider will redirect the user back to the application on the callback URL.

CANCEL SAVE

- **Title**: **Required** field that specifies the name for this provider configuration.
- **Description**: Optional field that provides a description for this provider configuration.

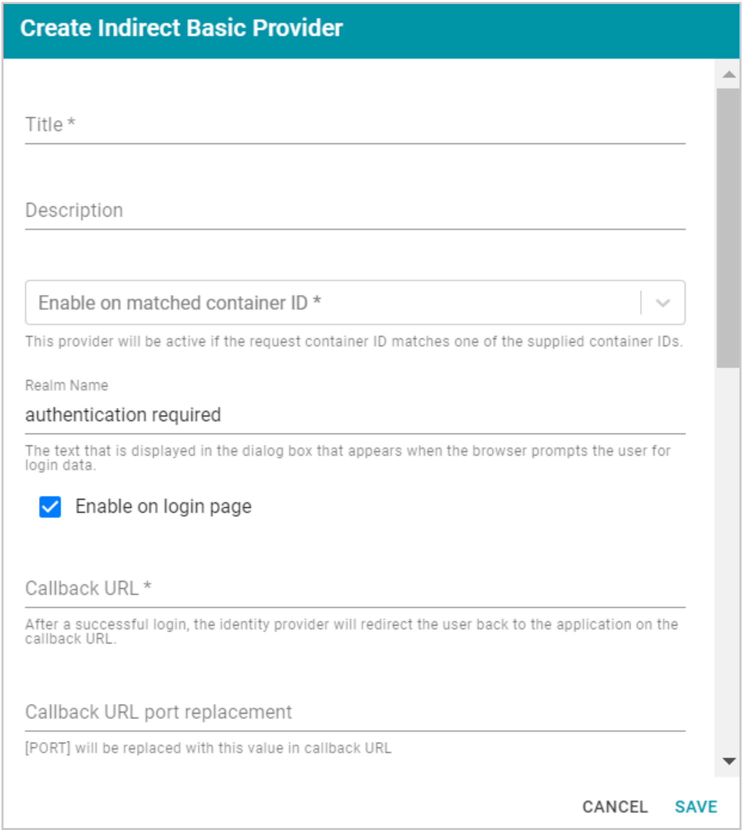
- **Enable on matched container ID**: **Required** field that lists the container ID(s) to match. This provider will be active if the request container ID matches one of the container IDs specified in this property. Click the field and select a container ID from the drop-down list. To specify multiple IDs, click the field again and select another value. To remove a container from the list, click the X on the right of the container name.
- **Client ID**: **Required** field that specifies the unique App ID for the client application.
- **Secret**: **Required** field that specifies the App Secret for the specified Client ID.
- **Confirm Password**: **Required** field that confirms the specified Secret.
- **Enable on login page**: Optional field that specifies whether to enable a link for this provider on the Anzo login screen.
- **Callback URL**: **Required** field that specifies the URL for the provider to use to redirect users back to the Anzo application after a successful login. Include the full URL to the Anzo instance, through the proxy if one exists. Specify the URL in quotes and append the value with `/anzo_authenticate`, i.e., `"<IP:port>/anzo_authenticate"`.
- **Callback URL port replacement**: Optional field that lists the port to use if the one specified in the Callback URL field is unavailable.
- **User Identifier**: Optional field that specifies the SSO provider attribute, such as email or username, to use for looking up users in the directory server.
- **Logout of IDP**: Optional field that specifies whether logging out of Anzo should also prompt the user to log out of the identity provider session. When this option is enabled, logging out of the Anzo application presents a "Perform central logout" dialog box. Selecting the **Perform central logout** checkbox logs the user out of the SSO session.
- **Default to IDP Logout**: When **Logout of IDP** is enabled, users are presented with an option to perform a central log out when they log out of Anzo. When the **Default to IDP Logout** option is enabled, users are not given a choice about logging out of the IDP. The central logout is performed by default.
- **Logout URL Suffix**: When Logout of IDP is enabled, the Logout URL Suffix is used to access the logout URL for the SSO provider. The [urlAfterLogout] placeholder is replaced with the SSO provider server URL.
- **Email Template regex**: If email was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between

email addresses stored by the SSO provider and email addresses returned by the directory server.

- **Email Template Replacement**: Optional field that specifies a replacement email template to use if there are variations found by Email Template regex.
- **User Template regex**: If user was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between user names stored by the SSO provider and user names returned by the directory server.
- **User Template Replacement**: Optional field that specifies a replacement user template to use if there are variations found by User Template regex.
- **Use username directly**:
- **Skip CSRF check**: Optional property that specifies whether to perform a cross-site request forgery (CSRF) check.
- **LDAP domain**: Optional field that specifies the LDAP domain to use for user lookup.
- **LDAP email property**: Optional field that specifies the LDAP email property to use to find the associated user's dn. For example, `http://openanzo.org/ontologies/2008/07/Anzo#ldapEmailInfo`.
- **Icon**: Optional property that specifies an SSO icon to use on the Anzo login screen. To select an image file, click the **Icon** field and select **Add File**.

Indirect Basic Provider

This section describes the settings that are available on the Create Indirect Basic Provider screen:



Create Indirect Basic Provider

Title *

Description

Enable on matched container ID *

This provider will be active if the request container ID matches one of the supplied container IDs.

Realm Name
authentication required

The text that is displayed in the dialog box that appears when the browser prompts the user for login data.

☑ Enable on login page

Callback URL *

After a successful login, the identity provider will redirect the user back to the application on the callback URL.

Callback URL port replacement

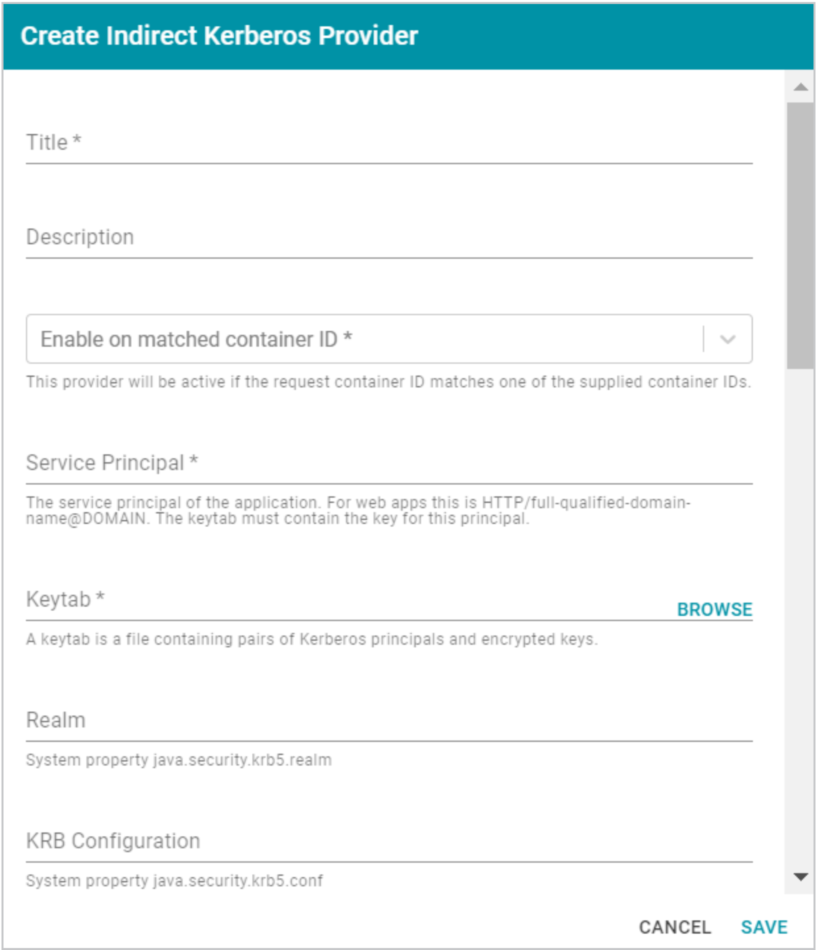
[PORT] will be replaced with this value in callback URL

- **Title**: **Required** field that specifies the name for this provider configuration.
- **Description**: Optional field that provides a description for this provider configuration.
- **Enable on matched container ID**: **Required** field that lists the container ID(s) to match. This provider will be active if the request container ID matches one of the container IDs specified in this property. Click the field and select a container ID from the drop-down list. To specify multiple IDs, click the field again and select another value. To remove a container from the list, click the X on the right of the container name.
- **Realm Name**: Optional field that specifies the name of the security realm.
- **Enable on login page**: Optional field that specifies whether to enable a link for this provider on the Anzo login screen.
- **Callback URL**: **Required** field that specifies the URL for the provider to use to redirect users back to the Anzo application after a successful login. Include the full URL to the Anzo instance, through the proxy if one exists. Specify the URL in quotes and append the value with `/anzo_authenticate`, i.e., `"<IP:port>/anzo_authenticate"`.

- **Callback URL port replacement**: Optional field that lists the port to use if the one specified in the Callback URL field is unavailable.
- **User Identifier**: Optional field that specifies the SSO provider attribute, such as email or username, to use for looking up users in the directory server.
- **Email Template regex**: If email was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between email addresses stored by the SSO provider and email addresses returned by the directory server.
- **Email Template Replacement**: Optional field that specifies a replacement email template to use if there are variations found by Email Template regex.
- **User Template regex**: If user was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between user names stored by the SSO provider and user names returned by the directory server.
- **User Template Replacement**: Optional field that specifies a replacement user template to use if there are variations found by User Template regex.
- **Use username directly**:
- **Skip CSRF check**: Optional property that specifies whether to perform a cross-site request forgery (CSRF) check.
- **LDAP domain**: Optional field that specifies the LDAP domain to use for user lookup.
- **LDAP email property**: Optional field that specifies the LDAP email property to use to find the associated user's dn. For example, `http://openanzo.org/ontologies/2008/07/Anzo#ldapEmailInfo`.
- **Icon**: Optional property that specifies an SSO icon to use on the Anzo login screen. To select an image file, click the **Icon** field and select **Add File**.

Indirect Kerberos Provider

This section describes the settings that are available on the Create Indirect Kerberos Provider screen:



Create Indirect Kerberos Provider

Title *

Description

Enable on matched container ID *

This provider will be active if the request container ID matches one of the supplied container IDs.

Service Principal *

The service principal of the application. For web apps this is HTTP/full-qualified-domain-name@DOMAIN. The keytab must contain the key for this principal.

Keytab *

A keytab is a file containing pairs of Kerberos principals and encrypted keys.

Realm

System property java.security.krb5.realm

KRB Configuration

System property java.security.krb5.conf

- **Title**: **Required** field that specifies the name for this provider configuration.
- **Description**: Optional field that provides a description for this provider configuration.
- **Enable on matched container ID**: **Required** field that lists the container ID(s) to match. This provider will be active if the request container ID matches one of the container IDs specified in this property. Click the field and select a container ID from the drop-down list. To specify multiple IDs, click the field again and select another value. To remove a container from the list, click the X on the right of the container name.
- **Service Principal**: **Required** field that specifies the service and DNS name for the application. For authentication through the web browser, specify the service principal value in the following format:

  ```
  HTTP/fully_qualified_domain_name@domain
  ```

  For example, `HTTP/server.example.com@example.com`.

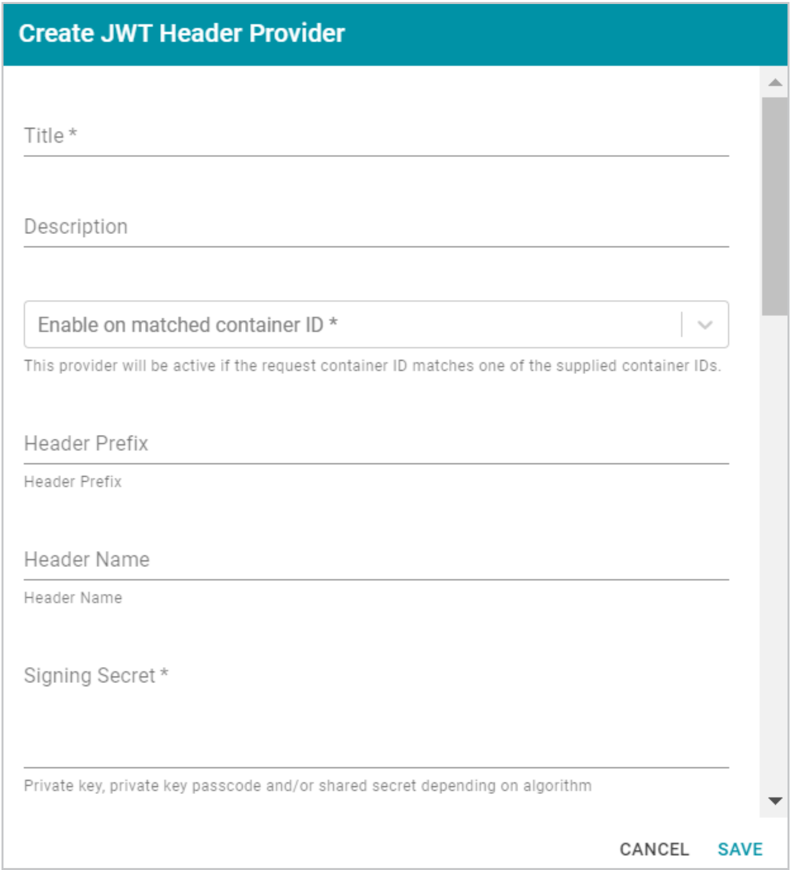
> **Note** The keytab file must contain the key for this principal.

- **Keytab**: **Required** field that specifies the .keytab file that lists the Kerberos principals and encrypted keys. Click the **Keytab** field to open the File Location dialog box and select the keytab file.
- **Realm**: Optional field that specifies the Kerberos realm that the service principal maps to.
- **KRB Configuration**: Optional field that specifies the path and file name for the krb5.conf file on the Kerberos instance. The default location is `/etc/krb5.conf`.
- **KDC**: Optional field that specifies the domain name for the Key Distribution Center.
- **Debug mode**: Optional field that specifies whether Kerberos debug logging is enabled for your provider.
- **Enable on login page**: Optional field that specifies whether to enable a link for this provider on the Anzo login screen.
- **Callback URL**: **Required** field that specifies the URL for the provider to use to redirect users back to the Anzo application after a successful login. Include the full URL to the Anzo instance, through the proxy if one exists. Specify the URL in quotes and append the value with `/anzo_authenticate`, i.e., `"<IP:port>/anzo_authenticate"`.
- **Callback URL port replacement**: Optional field that lists the port to use if the one specified in the Callback URL field is unavailable.
- **User Identifier**: Optional field that specifies the SSO provider attribute, such as email or username, to use for looking up users in the directory server.
- **Logout of IDP**: Optional field that specifies whether logging out of Anzo should also prompt the user to log out of the identity provider session. When this option is enabled, logging out of the Anzo application presents a "Perform central logout" dialog box. Selecting the **Perform central logout** checkbox logs the user out of the SSO session.
- **Default to IDP Logout**: When **Logout of IDP** is enabled, users are presented with an option to perform a central log out when they log out of Anzo. When the **Default to IDP Logout** option is enabled, users are not given a choice about logging out of the IDP. The central logout is performed by default.

- **Default to IDP Logout**: When **Logout of IDP** is enabled, users are presented with an option to perform a central log out when they log out of Anzo. When the **Default to IDP Logout** option is enabled, users are not given a choice about logging out of the IDP. The central logout is performed by default.
- **Logout URL Suffix**: When Logout of IDP is enabled, the Logout URL Suffix is used to access the logout URL for the SSO provider. The [urlAfterLogout] placeholder is replaced with the SSO provider server URL.
- **Email Template regex**: If email was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between email addresses stored by the SSO provider and email addresses returned by the directory server.
- **Email Template Replacement**: Optional field that specifies a replacement email template to use if there are variations found by Email Template regex.
- **User Template regex**: If user was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between user names stored by the SSO provider and user names returned by the directory server.
- **User Template Replacement**: Optional field that specifies a replacement user template to use if there are variations found by User Template regex.
- **Use username directly**:
- **Skip CSRF check**: Optional property that specifies whether to perform a cross-site request forgery (CSRF) check.
- **LDAP domain**: Optional field that specifies the LDAP domain to use for user lookup.
- **LDAP email property**: Optional field that specifies the LDAP email property to use to find the associated user's dn. For example, `http://openanzo.org/ontologies/2008/07/Anzo#ldapEmailInfo`.
- **Icon**: Optional property that specifies an SSO icon to use on the Anzo login screen. To select an image file, click the **Icon** field and select **Add File**.

JWT Header Provider

This section describes the settings that are available on the Create JWT Header Provider screen:



Create JWT Header Provider

Title *

Description

Enable on matched container ID *

This provider will be active if the request container ID matches one of the supplied container IDs.

Header Prefix

Header Prefix

Header Name

Header Name

Signing Secret *

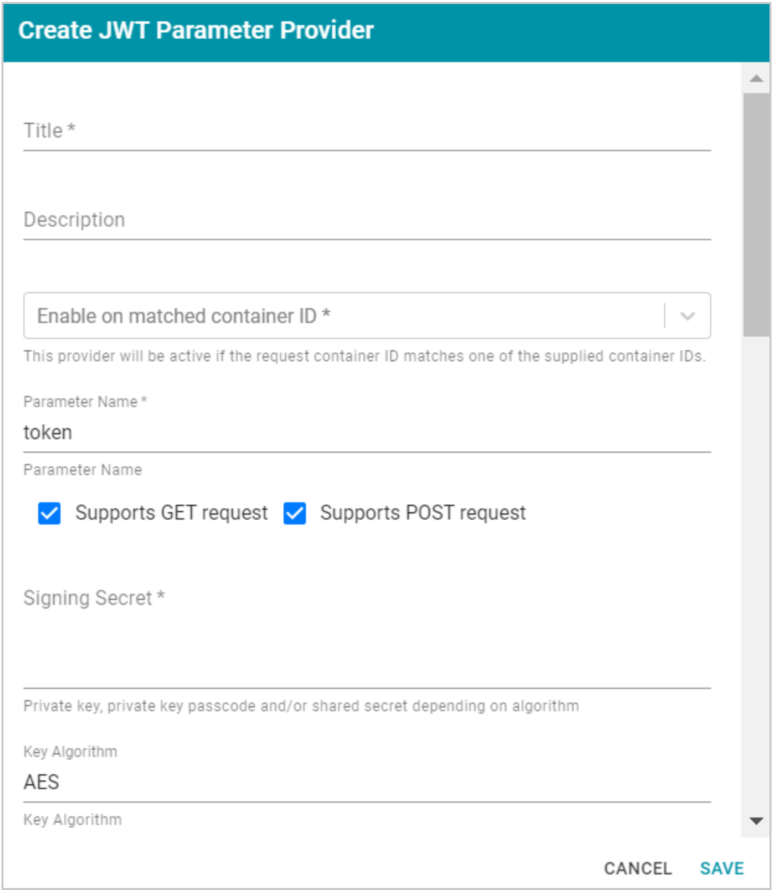
Private key, private key passcode and/or shared secret depending on algorithm

- **Title**: **Required** field that specifies the name for this provider configuration.
- **Description**: Optional field that provides a description for this provider configuration.
- **Enable on matched container ID**: **Required** field that lists the container ID(s) to match. This provider will be active if the request container ID matches one of the container IDs specified in this property. Click the field and select a container ID from the drop-down list. To specify multiple IDs, click the field again and select another value. To remove a container from the list, click the X on the right of the container name.
- **Header Prefix**: Optional field that specifies the header prefix if one is used.
- **Header Name**: Optional field that specifies the header name.
- **Signing Secret**: **Required** field that specifies the secret the token is signed with.
- **Key Algorithm**: Optional field that specifies the signing algorithm that is used.

- **Encryption Method**: Optional field that specifies the encryption method used for encrypted tokens.
- **Encryption Secret**: Optional field that specifies the secret used for encrypted tokens.
- **Enable on match regex**: Optional field that defines regular expression rules for matching request URLs to enable. To add a rule, type an expression in the field and click **Add**. This provider will be active if the request URL matches any of the supplied expressions. If Enable on match regex is blank, the provider will be active by default.
- **Disable on match regex**: Optional field that defines regular expression rules for matching request URLs to disable. To add a rule, type an expression in the field and click **Add**. This provider will be inactive if the request URL matches any of the supplied expressions. If Disable on match regex is blank, the provider will be active by default.
- **Email Template regex**: If email was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between email addresses stored by the SSO provider and email addresses returned by the directory server.
- **Email Template Replacement**: Optional field that specifies a replacement email template to use if there are variations found by Email Template regex.
- **User Template regex**: If user was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between user names stored by the SSO provider and user names returned by the directory server.
- **User Template Replacement**: Optional field that specifies a replacement user template to use if there are variations found by User Template regex.
- **Use username directly**:
- **Skip CSRF check**: Optional property that specifies whether to perform a cross-site request forgery (CSRF) check.
- **LDAP domain**: Optional field that specifies the LDAP domain to use for user lookup.
- **LDAP email property**: Optional field that specifies the LDAP email property to use to find the associated user's dn. For example, `http://openanzo.org/ontologies/2008/07/Anzo#ldapEmailInfo`.

JWT Parameter Provider

This section describes the settings that are available on the Create JWT Parameter Provider screen:



Create JWT Parameter Provider

Title *

Description

Enable on matched container ID *

This provider will be active if the request container ID matches one of the supplied container IDs.

Parameter Name *
token
Parameter Name

☑ Supports GET request ☑ Supports POST request

Signing Secret *

Private key, private key passcode and/or shared secret depending on algorithm

Key Algorithm
AES
Key Algorithm

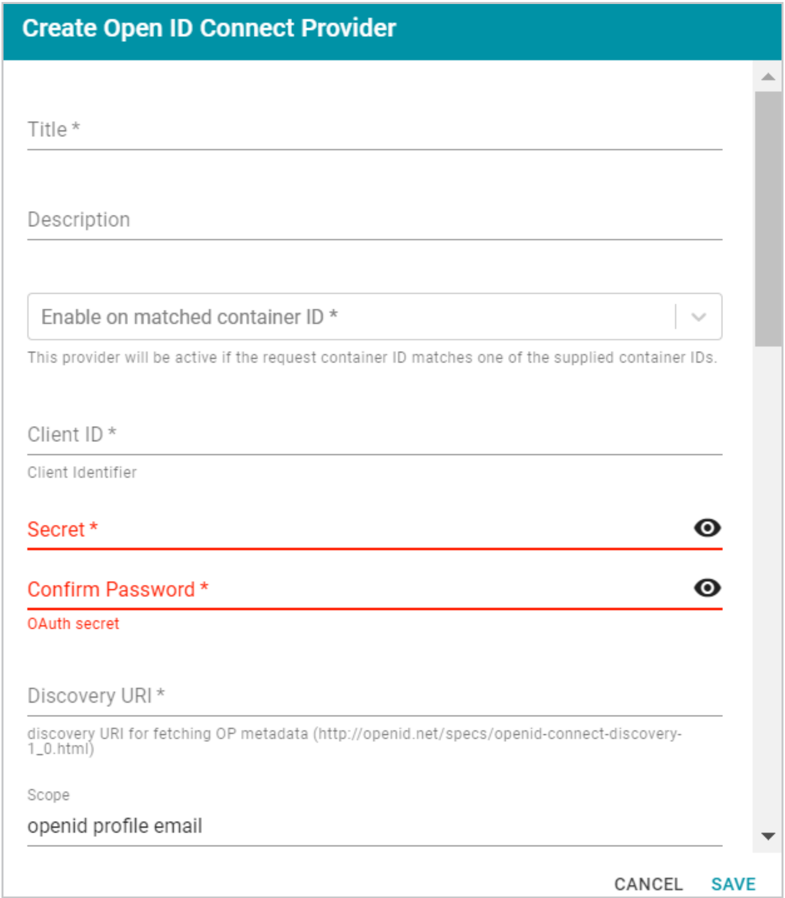
- **Title**: **Required** field that specifies the name for this provider configuration.
- **Description**: Optional field that provides a description for this provider configuration.
- **Enable on matched container ID**: **Required** field that lists the container ID(s) to match. This provider will be active if the request container ID matches one of the container IDs specified in this property. Click the field and select a container ID from the drop-down list. To specify multiple IDs, click the field again and select another value. To remove a container from the list, click the X on the right of the container name.
- **Parameter Name**: **Required** field that specifies the header parameter name.
- **Supports GET request**: Optional field that indicates whether GET requests are supported using the token.
- **Supports POST request**: Optional field that indicates whether POST requests are supported using the token.

- **Signing Secret**: **Required** field that specifies the secret the token is signed with.
- **Key Algorithm**: Optional field that specifies the signing algorithm that is used.
- **Encryption Algorithm**:
- **Encryption Method**: Optional field that specifies the encryption method used for encrypted tokens.
- **Encryption Secret**: Optional field that specifies the secret used for encrypted tokens.
- **Enable on match regex**: Optional field that defines regular expression rules for matching request URLs to enable. To add a rule, type an expression in the field and click **Add**. This provider will be active if the request URL matches any of the supplied expressions. If Enable on match regex is blank, the provider will be active by default.
- **Disable on match regex**: Optional field that defines regular expression rules for matching request URLs to disable. To add a rule, type an expression in the field and click **Add**. This provider will be inactive if the request URL matches any of the supplied expressions. If Disable on match regex is blank, the provider will be active by default.
- **Email Template regex**: If email was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between email addresses stored by the SSO provider and email addresses returned by the directory server.
- **Email Template Replacement**: Optional field that specifies a replacement email template to use if there are variations found by Email Template regex.
- **User Template regex**: If user was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between user names stored by the SSO provider and user names returned by the directory server.
- **User Template Replacement**: Optional field that specifies a replacement user template to use if there are variations found by User Template regex.
- **Use username directly**:
- **Skip CSRF check**: Optional property that specifies whether to perform a cross-site request forgery (CSRF) check.
- **LDAP domain**: Optional field that specifies the LDAP domain to use for user lookup.

- **LDAP email property**: Optional field that specifies the LDAP email property to use to find the associated user's dn. For example, `http://openanzo.org/ontologies/2008/07/Anzo#ldapEmailInfo`.

Open ID Connect Provider

This section describes the settings that are available on the Create Open ID Connect Provider screen:



Create Open ID Connect Provider

Title *

Description

Enable on matched container ID *

This provider will be active if the request container ID matches one of the supplied container IDs.

Client ID *

Client Identifier

Secret *

Confirm Password *

OAuth secret

Discovery URI *

discovery URI for fetching OP metadata (http://openid.net/specs/openid-connect-discovery-1_0.html)

Scope

openid profile email

- **Title**: **Required** field that specifies the name for this provider configuration.
- **Description**: Optional field that provides a description for this provider configuration.
- **Enable on matched container ID**: **Required** field that lists the container ID(s) to match. This provider will be active if the request container ID matches one of the container IDs specified in this property. Click the field and select a container ID from the

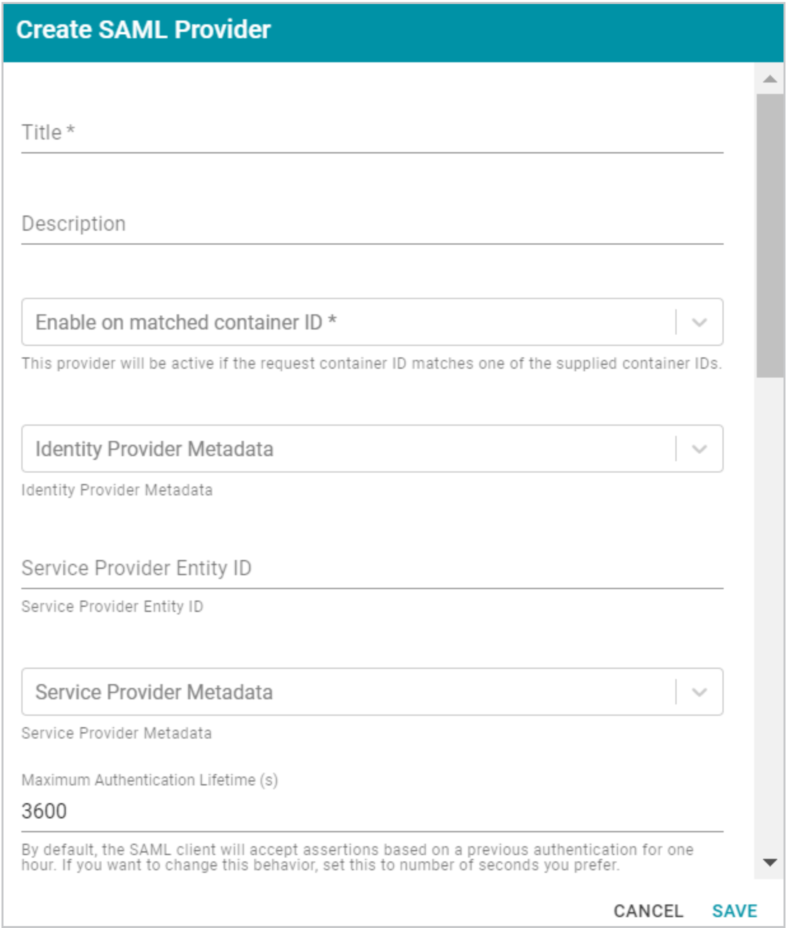
drop-down list. To specify multiple IDs, click the field again and select another value. To remove a container from the list, click the X on the right of the container name.

- **Client ID**: **Required** field that specifies client ID or consumer key value from the provider application.
- **Secret**: **Required** field that specifies the client secret from the provider application.
- **Confirm Secret**: **Required** field to confirm the specified Secret.
- **Discovery URI**: **Required** field that specifies the discovery URI to use for fetching OP Metadata.
- **Scope**: Optional field that specifies the scope to send to the authorization endpoint with the request.
- **Preferred JWS Algorithm**: Optional field that lists the preferred signing algorithm.
- **Enable on login page**: Optional field that specifies whether to enable a link for this provider on the Anzo login screen.
- **Callback URL**: **Required** field that specifies the URL for the provider to use to redirect users back to the Anzo application after a successful login. Include the full URL to the Anzo instance, through the proxy if one exists. Specify the URL in quotes and append the value with `/anzo_authenticate`, i.e., `"<IP:port>/anzo_authenticate"`.
- **Callback URL port replacement**: Optional field that lists the port to use if the one specified in the Callback URL field is unavailable.
- **User Identifier**: Optional field that specifies the SSO provider attribute, such as email or username, to use for looking up users in the directory server.
- **Logout of IDP**: Optional field that specifies whether logging out of Anzo should also prompt the user to log out of the identity provider session. When this option is enabled, logging out of the Anzo application presents a "Perform central logout" dialog box. Selecting the **Perform central logout** checkbox logs the user out of the SSO session.
- **Default to IDP Logout**: When **Logout of IDP** is enabled, users are presented with an option to perform a central log out when they log out of Anzo. When the **Default to IDP Logout** option is enabled, users are not given a choice about logging out of the IDP. The central logout is performed by default.

- **Logout URL Suffix**: When Logout of IDP is enabled, the Logout URL Suffix is used to access the logout URL for the SSO provider. The [urlAfterLogout] placeholder is replaced with the SSO provider server URL.
- **Email Template regex**: If email was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between email addresses stored by the SSO provider and email addresses returned by the directory server.
- **Email Template Replacement**: Optional field that specifies a replacement email template to use if there are variations found by Email Template regex.
- **User Template regex**: If user was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between user names stored by the SSO provider and user names returned by the directory server.
- **User Template Replacement**: Optional field that specifies a replacement user template to use if there are variations found by User Template regex.
- **Use username directly**:
- **Skip CSRF check**: Optional property that specifies whether to perform a cross-site request forgery (CSRF) check.
- **LDAP domain**: Optional field that specifies the LDAP domain to use for user lookup.
- **LDAP email property**: Optional field that specifies the LDAP email property to use to find the associated user's dn. For example, `http://openanzo.org/ontologies/2008/07/Anzo#ldapEmailInfo`.
- **Icon**: Optional property that specifies an SSO icon to use on the Anzo login screen. To select an image file, click the **Icon** field and select **Add File**.

SAML Provider

This section describes the settings that are available on the Create SAML Provider screen:



Create SAML Provider

Title *

Description

Enable on matched container ID *

This provider will be active if the request container ID matches one of the supplied container IDs.

Identity Provider Metadata

Identity Provider Metadata

Service Provider Entity ID

Service Provider Entity ID

Service Provider Metadata

Service Provider Metadata

Maximum Authentication Lifetime (s)

3600

By default, the SAML client will accept assertions based on a previous authentication for one hour. If you want to change this behavior, set this to number of seconds you prefer.

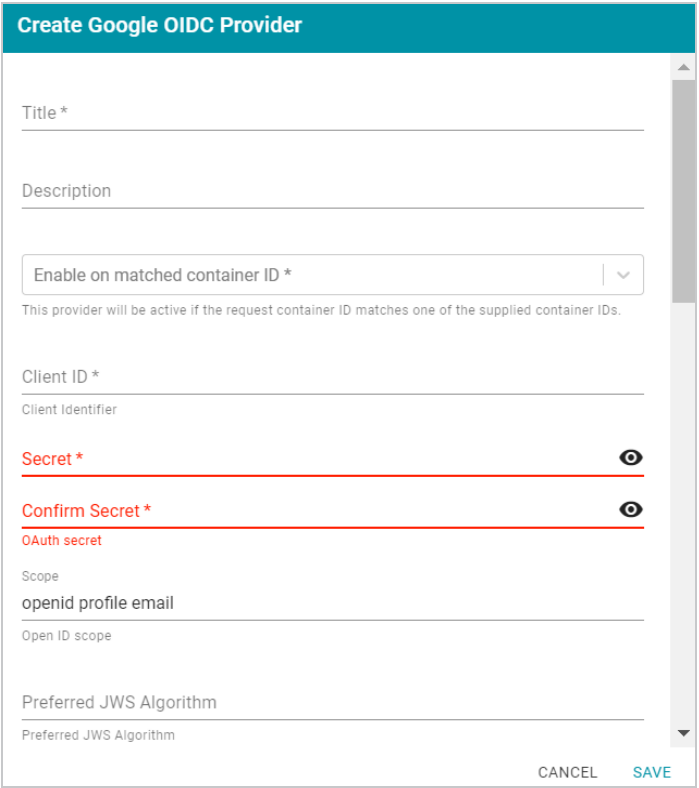
- **Title**: **Required** field that specifies the name for this provider configuration.
- **Description**: Optional field that provides a description for this provider configuration.
- **Enable on matched container ID**: **Required** field that lists the container ID(s) to match. This provider will be active if the request container ID matches one of the container IDs specified in this property. Click the field and select a container ID from the drop-down list. To specify multiple IDs, click the field again and select another value. To remove a container from the list, click the X on the right of the container name.
- **Identity Provider Metadata**: **Required** field that specifies the identity provider metadata .xml file. To add the file, click the **Identity Provider Metadata** field, click **Add File**, and select the file.
- **Service Provider Entity ID**:

- **Service Provider Metadata**: Optional field that specifies the server provider metadata .xml file. To add the file, click the **Server Provider Metadata** field, click **Add File**, and select the file.
- **Maximum Authentication Lifetime (s)**:
- **Enable on login page**: Optional field that specifies whether to enable a link for this provider on the Anzo login screen.
- **Callback URL**: **Required** field that specifies the URL for the provider to use to redirect users back to the Anzo application after a successful login. Include the full URL to the Anzo instance, through the proxy if one exists. Specify the URL in quotes and append the value with `/anzo_authenticate`, i.e., `"<IP:port>/anzo_authenticate"`.
- **Callback URL port replacement**: Optional field that lists the port to use if the one specified in the Callback URL field is unavailable.
- **User Identifier**: Optional field that specifies the SSO provider attribute, such as email or username, to use for looking up users in the directory server.
- **Logout of IDP**: Optional field that specifies whether logging out of Anzo should also prompt the user to log out of the identity provider session. When this option is enabled, logging out of the Anzo application presents a "Perform central logout" dialog box. Selecting the **Perform central logout** checkbox logs the user out of the SSO session.
- **Default to IDP Logout**: When **Logout of IDP** is enabled, users are presented with an option to perform a central log out when they log out of Anzo. When the **Default to IDP Logout** option is enabled, users are not given a choice about logging out of the IDP. The central logout is performed by default.
- **Logout URL Suffix**: When Logout of IDP is enabled, the Logout URL Suffix is used to access the logout URL for the SSO provider. The [urlAfterLogout] placeholder is replaced with the SSO provider server URL.
- **Email Template regex**: If email was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between email addresses stored by the SSO provider and email addresses returned by the directory server.
- **Email Template Replacement**: Optional field that specifies a replacement email template to use if there are variations found by Email Template regex.

- **User Template regex**: If user was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between user names stored by the SSO provider and user names returned by the directory server.
- **User Template Replacement**: Optional field that specifies a replacement user template to use if there are variations found by User Template regex.
- **Use username directly**:
- **Skip CSRF check**: Optional property that specifies whether to perform a cross-site request forgery (CSRF) check.
- **LDAP domain**: Optional field that specifies the LDAP domain to use for user lookup.
- **LDAP email property**: Optional field that specifies the LDAP email property to use to find the associated user's dn. For example, `http://openanzo.org/ontologies/2008/07/Anzo#ldapEmailInfo`.
- **Icon**: Optional property that specifies an SSO icon to use on the Anzo login screen. To select an image file, click the **Icon** field and select **Add File**.

Google OIDC Provider

This section describes the settings that are available on the Create Google OIDC Provider screen:



Create Google OIDC Provider

Title *

Description

Enable on matched container ID *

This provider will be active if the request container ID matches one of the supplied container IDs.

Client ID *

Client Identifier

Secret *

Confirm Secret *

OAuth secret

Scope

openid profile email

Open ID scope

Preferred JWS Algorithm

Preferred JWS Algorithm

- **Title**: **Required** field that specifies the name for this provider configuration.
- **Description**: Optional field that provides a description for this provider configuration.
- **Enable on matched container ID**: **Required** field that lists the container ID(s) to match. This provider will be active if the request container ID matches one of the container IDs specified in this property. Click the field and select a container ID from the drop-down list. To specify multiple IDs, click the field again and select another value. To remove a container from the list, click the X on the right of the container name.
- **Client ID**: **Required** field that specifies client ID or consumer key value from the provider application.
- **Secret**: **Required** field that specifies the client secret from the provider application.
- **Confirm Secret**: **Required** field to confirm the specified Secret.
- **Scope**: Optional field that specifies the scope to send to the authorization endpoint with the request.
- **Preferred JWS Algorithm**: Optional field that lists the preferred signing algorithm.
- **Enable on login page**: Optional field that specifies whether to enable a link for this provider on the Anzo login screen.

- **Callback URL**: **Required** field that specifies the URL for the provider to use to redirect users back to the Anzo application after a successful login. Include the full URL to the Anzo instance, through the proxy if one exists. Specify the URL in quotes and append the value with `/anzo_authenticate`, i.e., "`<IP:port>/anzo_authenticate`".
- **Callback URL port replacement**: Optional field that lists the port to use if the one specified in the Callback URL field is unavailable.
- **User Identifier**: Optional field that specifies the SSO provider attribute, such as email or username, to use for looking up users in the directory server.
- **Logout of IDP**: Optional field that specifies whether logging out of Anzo should also prompt the user to log out of the identity provider session. When this option is enabled, logging out of the Anzo application presents a "Perform central logout" dialog box. Selecting the **Perform central logout** checkbox logs the user out of the SSO session.
- **Default to IDP Logout**: When **Logout of IDP** is enabled, users are presented with an option to perform a central log out when they log out of Anzo. When the **Default to IDP Logout** option is enabled, users are not given a choice about logging out of the IDP. The central logout is performed by default.
- **Logout URL Suffix**: When Logout of IDP is enabled, the Logout URL Suffix is used to access the logout URL for the SSO provider. The [urlAfterLogout] placeholder is replaced with the SSO provider server URL.
- **Email Template regex**: If email was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between email addresses stored by the SSO provider and email addresses returned by the directory server.
- **Email Template Replacement**: Optional field that specifies a replacement email template to use if there are variations found by Email Template regex.
- **User Template regex**: If user was specified as the User Identifier, you can use this optional field to include a regular expression to use for identifying variations between user names stored by the SSO provider and user names returned by the directory server.
- **User Template Replacement**: Optional field that specifies a replacement user template to use if there are variations found by User Template regex.
- **Use username directly**:

- **Skip CSRF check**: Optional property that specifies whether to perform a cross-site request forgery (CSRF) check.
- **LDAP domain**: Optional field that specifies the LDAP domain to use for user lookup.
- **LDAP email property**: Optional field that specifies the LDAP email property to use to find the associated user's dn. For example, `http://openanzo.org/ontologies/2008/07/Anzo#ldapEmailInfo`.
- **Icon**: Optional property that specifies an SSO icon to use on the Anzo login screen. To select an image file, click the **Icon** field and select **Add File**.

3. Click **Save** to save the provider configuration.

Related Topics

User Management and Access Control Concepts

Connecting to a Directory Server

Creating and Managing Roles

In Anzo, Groups (or Users) are added to **Roles** and the Roles are configured to grant access to *functionality*. Role permissions control access to menus and screens in the Anzo and Administration applications. Access to functionality cannot be assigned to Groups or Users, only to Roles.

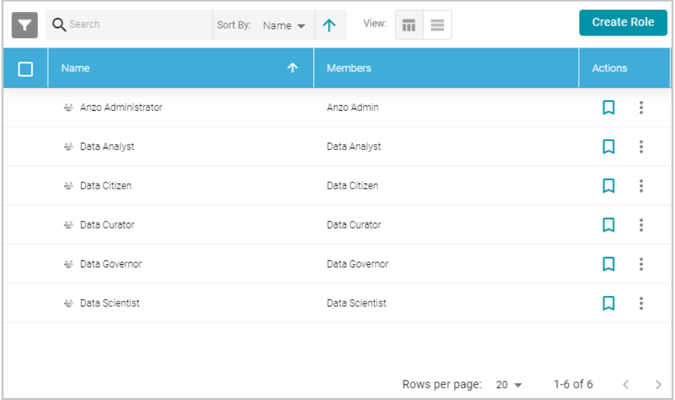
> **Tip**
> For more information about Role, User, and Group management, see User Management Concepts.

This topic provides instructions for creating or changing the Roles to use for controlling access to Anzo functionality. For information about the predefined Anzo roles, see Predefined Anzo Roles and Permissions.

- Creating a New Role
- Adding Users or Groups to a Role
- Configuring Role Permissions

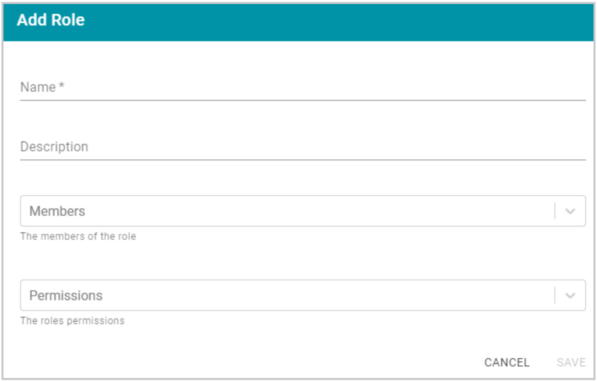
Creating a New Role

1. In the Administration application, expand the **User Management** menu and click **Roles**. Anzo displays the Roles screen, which lists the existing roles. For example:



□	Name	Members	Actions
	Anzo Administrator	Anzo Admin	
	Data Analyst	Data Analyst	
	Data Citizen	Data Citizen	
	Data Curator	Data Curator	
	Data Governor	Data Governor	
	Data Scientist	Data Scientist	

2. On the Roles screen, click the **Create Role** button. Anzo displays the Add Role dialog box.



Add Role

Name *

Description

Members

The members of the role

Permissions

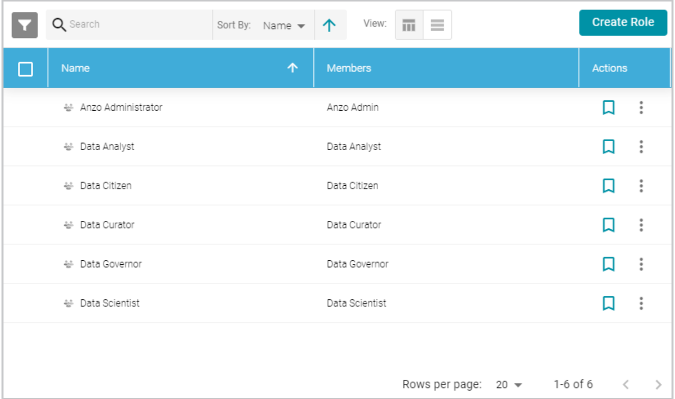
The roles permissions

3. Complete the required fields and enter any optional group details:
   - **Name**: The name for the new role.
   - **Description**: An optional description of the role.
   - **Members**: The users or groups who are members of the role. Click the **Members** field to select a member. Click the field again to select additional members.
   - **Permissions**: The list of Anzo features that this role has permission to access. Click the **Permissions** field and select a permission to add it to the list. Click the field again to select additional permissions. For details about each of the permissions, see the Role Permissions Reference.
4. Click **Save** to add the role to the system. Anzo adds the new role to the list of roles on the Roles screen.

Adding Users or Groups to a Role

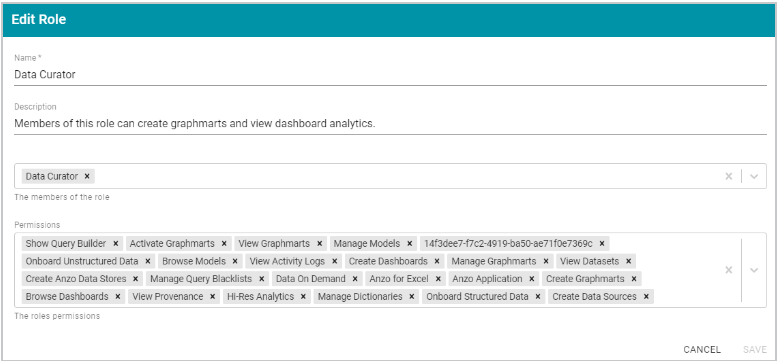
Follow the instructions below to add users and/or groups to a role.

1. In the Administration application, expand the **User Management** menu and click **Roles**. Anzo displays the Roles screen, which lists the existing roles. For example:



Name	Members	Actions
Anzo Administrator	Anzo Admin	
Data Analyst	Data Analyst	
Data Citizen	Data Citizen	
Data Curator	Data Curator	
Data Governor	Data Governor	
Data Scientist	Data Scientist	

2. Click the name of the role that you want to add users or groups to. Anzo opens the Edit Role dialog box. For example:



Name *
Data Curator

Description
Members of this role can create graphmarts and view dashboard analytics.

Data Curator ×
The members of the role

Permissions
Show Query Builder × Activate Graphmarts × View Graphmarts × Manage Models × 14f3dee7-f7c2-4919-ba50-ae71f0e7369c ×
Onboard Unstructured Data × Browse Models × View Activity Logs × Create Dashboards × Manage Graphmarts × View Datasets ×
Create Anzo Data Stores × Manage Query Blacklists × Data On Demand × Anzo for Excel × Anzo Application × Create Graphmarts ×
Browse Dashboards × View Provenance × Hi-Res Analytics × Manage Dictionaries × Onboard Structured Data × Create Data Sources ×
The roles permissions

3. Click the **Members** drop-down list to display the list of all available users and groups. You can also search for a user or group by typing a name in the Members field. Click a name to add that user or group to the role. Click the field again to select additional members. To remove a member from the role, click the X to the right of the name.

> **Note**
>
> If you do not see users or groups that you expect to see, it is possible that Anzo is out of sync with the directory server. If groups or users have been modified on the directory server, and a user has not logged in to Anzo for an extended time, the data may need to be refreshed in Anzo. The **Users** and **Groups** screens in the User Management menu have **Sync Directory** buttons that you can click to synchronize with the directory server and update the data in Anzo.

4. When you have finished adding members, click **Save** to save the changes to the role.

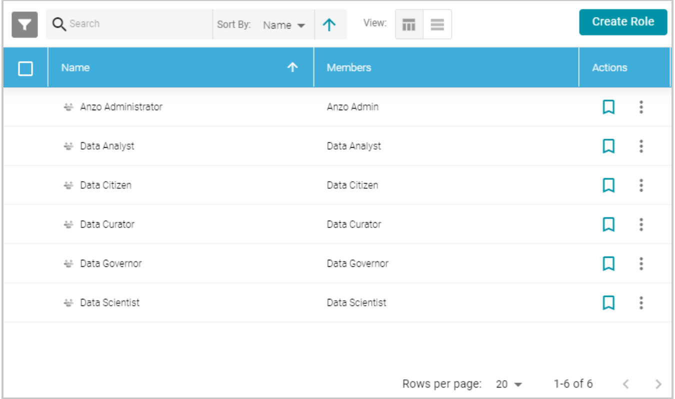
Note

When modifying an existing user's access by adding or removing roles from their account, Cambridge Semantics recommends that the user logs out of Anzo and clears their browser cache to ensure that the access changes are reflected in the user interface.

Configuring Role Permissions

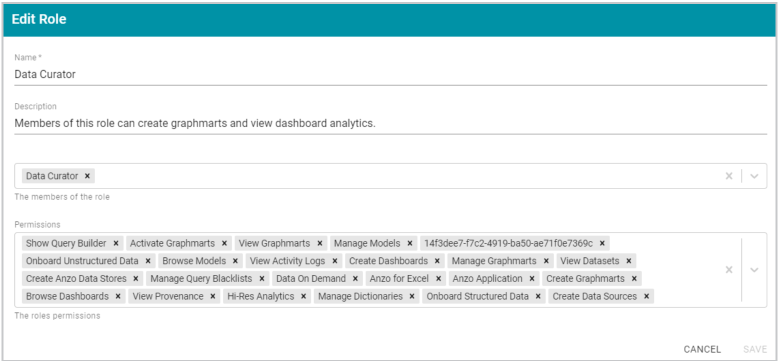
Follow the instructions below to add or remove permissions from a role. For details about each of the permissions, see the Role Permissions Reference.

1. In the Administration application, expand the **User Management** menu and click **Roles**. Anzo displays the Roles screen, which lists the existing roles. For example:



Name	Members	Actions
Anzo Administrator	Anzo Admin	
Data Analyst	Data Analyst	
Data Citizen	Data Citizen	
Data Curator	Data Curator	
Data Governor	Data Governor	
Data Scientist	Data Scientist	

2. Click the name of the role for which you want to configure permissions. Anzo opens the Edit Role dialog box. For example:



Edit Role

Name *
Data Curator

Description
Members of this role can create graphmarts and view dashboard analytics.

Data Curator ×
The members of the role

Permissions
Show Query Builder × Activate Graphmarts × View Graphmarts × Manage Models × 14f3dee7-f7c2-4919-ba50-ae71f0e7369c × Onboard Unstructured Data × Browse Models × View Activity Logs × Create Dashboards × Manage Graphmarts × View Datasets × Create Anzo Data Stores × Manage Query Blacklists × Data On Demand × Anzo for Excel × Anzo Application × Create Graphmarts × Browse Dashboards × View Provenance × Hi-Res Analytics × Manage Dictionaries × Onboard Structured Data × Create Data Sources ×
The roles permissions

3. The **Permissions** field lists all of the permissions that are applied to the role. To remove a permission, click the X to the right of the permission name. To add a permission click the field to open the Permissions drop-down list. Click a name to add that permission to the role. Click the field again to

select additional permissions.

4. When you have finished changing permissions, click **Save** to save the changes to the role.

Related Topics

User Management and Access Control Concepts

Predefined Anzo Roles and Permissions

Role Permissions Reference

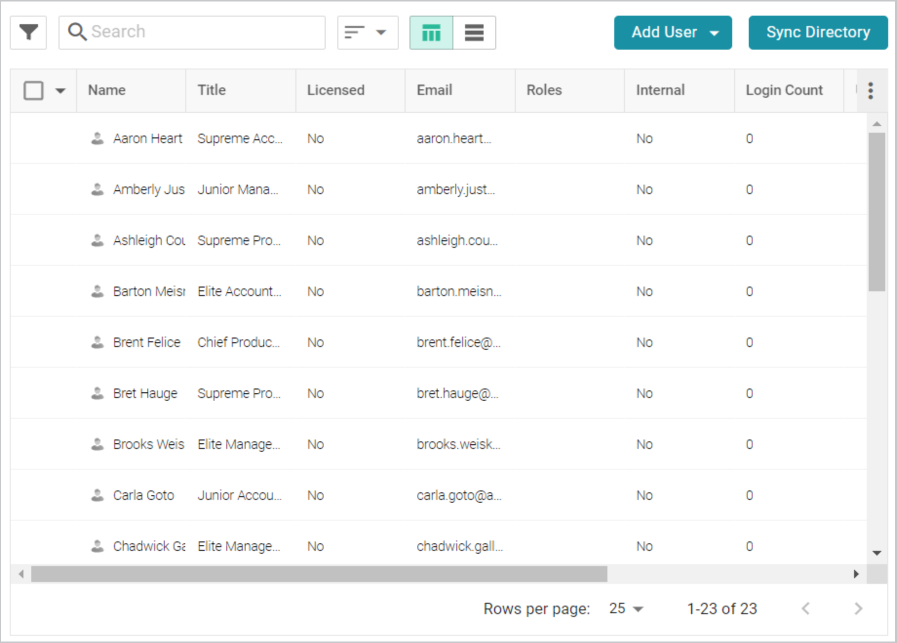
Creating an Internal Anzo User

Creating an Internal Anzo User

User and group accounts are typically managed in a central directory server that is connected to Anzo. The groups from the directory server are added to Anzo roles, and access to Anzo applications and features is configured for the roles. However, you can create a user account directly in Anzo. Accounts that are created in Anzo are stored in Anzo's internal LDAP server. Follow the instructions below to create a new internal Anzo user account.

> **Tip**
> For instructions on adding directory users to Anzo, see Adding Directory Users and Groups to Anzo.

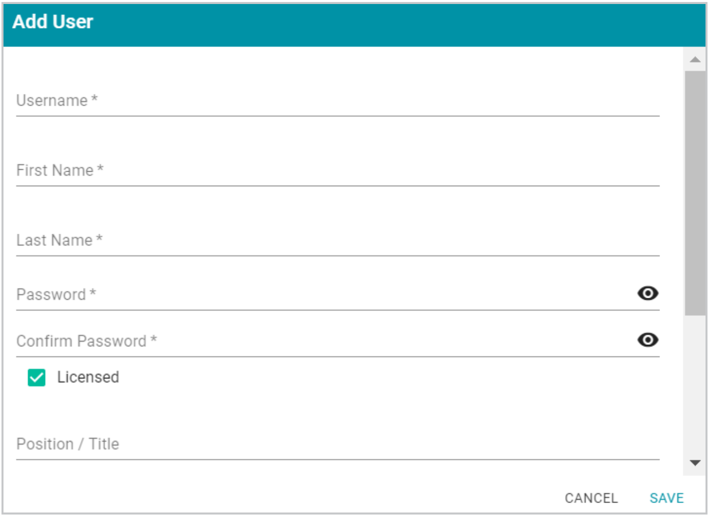
1. In the Administration application, expand the **User Management** menu and click **Users**. Anzo displays the Users screen, which lists the existing users. For example:



Name	Title	Licensed	Email	Roles	Internal	Login Count
Aaron Heart	Supreme Acc...	No	aaron.heart...		No	0
Amberly Jus	Junior Mana...	No	amberly.just...		No	0
Ashleigh Cou	Supreme Pro...	No	ashleigh.cou...		No	0
Barton Meisı	Elite Account...	No	barton.meisn...		No	0
Brent Felice	Chief Produc...	No	brent.felice@...		No	0
Bret Hauge	Supreme Pro...	No	bret.hauge@...		No	0
Brooks Weis	Elite Manage...	No	brooks.weisk...		No	0
Carla Goto	Junior Accou...	No	carla.goto@a...		No	0
Chadwick Ga	Elite Manage...	No	chadwick.gall...		No	0

Rows per page: 25 1-23 of 23

2. On the Users screen, click the **Add User** button and select **Add User**. Anzo opens the Add User dialog box.



Add User

Username *

First Name *

Last Name *

Password *

Confirm Password *

☑ Licensed

Position / Title

3. Complete the required fields and enter any optional user details:
   - **Username**: The user name that the user will use to log in to Anzo.
   - **First Name**: The user's first name.
   - **Last Name**: The user's last name.
   - **Password** and **Confirm Password**: Type a password for the user.
   - **Licensed**: Select the **Licensed** checkbox if you want this user to be able to log in to the Anzo applications. If you want to add this user to the system but do not want to give him or her access to Anzo applications at this time, clear the Licensed checkbox.
   - **Position/Title**: The user's job title or position.
   - **Email**: The user's email address.
   - **Phone**: The user's phone number.
   - **Roles**: The role or roles that the user is a member of. Roles define the user's level of access to Anzo applications and features. Click the **Roles** field and select a role from the drop-down list. Click the field again to select additional roles.
4. When you have finished configuring the user account, click **Save** to add the user to the system.

For more information about roles, see Creating and Managing Roles. For a description of the default Anzo roles, see Predefined Anzo Roles and Permissions.

Related Topics

User Management and Access Control Concepts

Creating and Managing Roles

Predefined Anzo Roles and Permissions

Adding Directory Users and Groups to Anzo

Predefined Anzo Roles and Permissions

This topic describes the roles that are predefined in Anzo and lists the permissions that are assigned to each role by default. The predefined roles can be removed or modified as desired. For instructions on changing roles, see Creating and Managing Roles.

- System Administrator
- Base Permissions (Everyone and Authenticated User Roles)
- Anzo Administrator
- Data Analyst
- Data Citizen
- Data Curator
- Data Governor
- Data Scientist

System Administrator

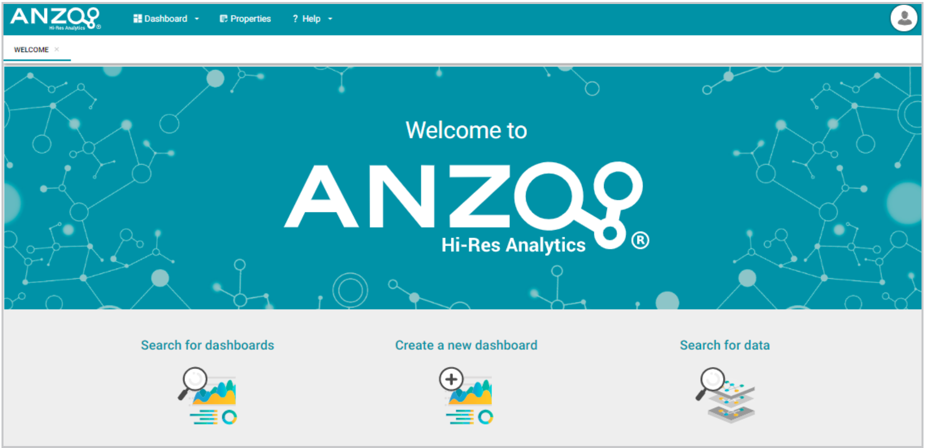
The System Administrator account, usually named **sysadmin**, is created during the Anzo installation. This account has permission to access all Anzo features in the main Anzo application as well as administrative features in the Administration application. In addition, the sysadmin user has read and write access to all of the artifacts (such as data sources, models, and pipelines) that are created by all Anzo users. The sysadmin user permissions cannot be changed, and the account cannot be deleted. In addition, artifacts cannot be configured to restrict sysadmin access. For information about changing the system administrator password, see Set the System Administrator Password.

Base Permissions (Everyone and Authenticated User Roles)

There is a set of base permissions that are applied to all user accounts by default. If a user account is created in Anzo but no roles are assigned, that user has the permissions of the **Authenticated User** role. By default, authenticated users cannot access the Anzo application but can access the Hi-Res Analytics application where they can browse for and create dashboards. They can also view data that is shared from Data on Demand endpoints.

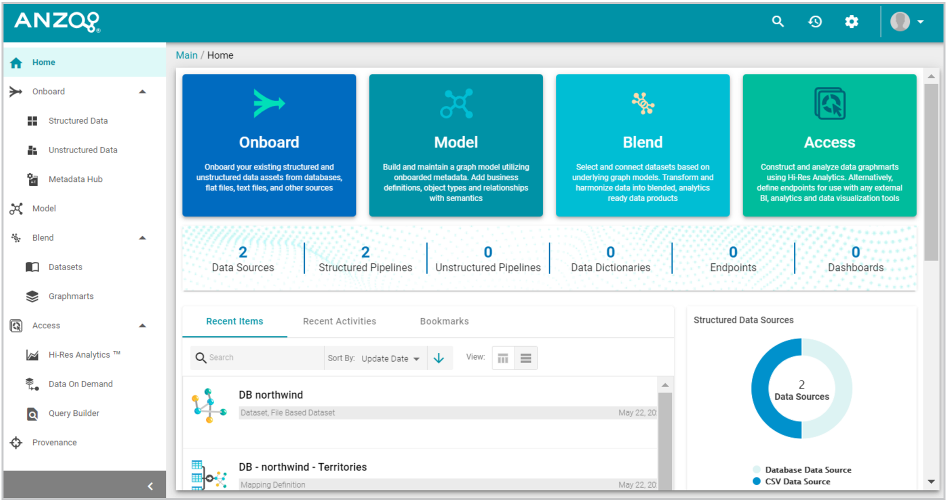
> **Note**
>
> If **Anonymous User Access** is enabled on the system, unauthenticated users (users who do not have a user account in Anzo) have the permissions that are included in the **Everyone** role. The Everyone role is only used to apply permissions for unauthenticated users when anonymous access is allowed. For information about anonymous access, see Configure Anonymous User Access.

The image below shows an example of the view an authenticated user has in the Hi-Res Analytics application.



Anzo Administrator

By default the Anzo Administrator role has access to all menus and features in the Anzo application as well as the Administration application. The image below shows an example of the view a user with the Anzo Administrator role has in the Anzo application.



Onboard

Onboard your existing structured and unstructured data assets from databases, flat files, text files, and other sources

Model

Build and maintain a graph model utilizing onboarded metadata. Add business definitions, object types and relationships with semantics

Blend

Select and connect datasets based on underlying graph models. Transform and harmonize data into blended, analytics ready data products

Access

Construct and analyze data graphmarts using Hi-Res Analytics. Alternatively, define endpoints for use with any external BI, analytics and data visualization tools

2 Data Sources | 2 Structured Pipelines | 0 Unstructured Pipelines | 0 Data Dictionaries | 0 Endpoints | 0 Dashboards

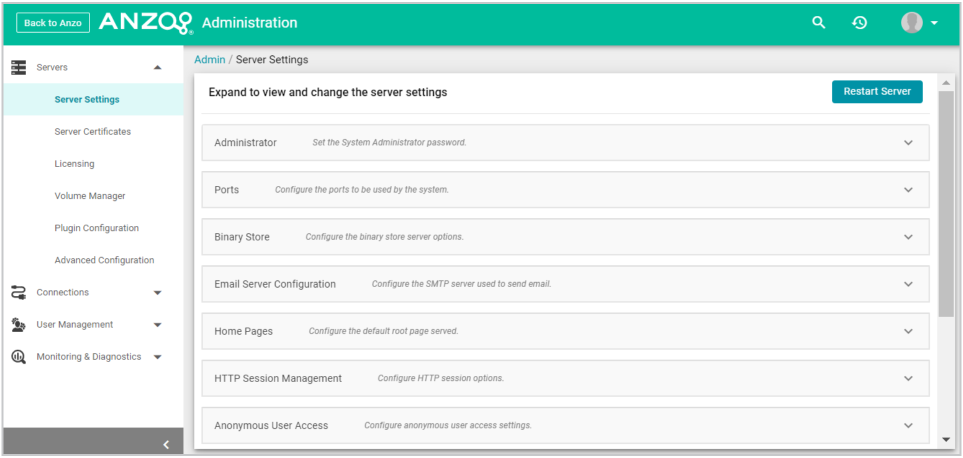
DB northwind
Dataset, File Based Dataset May 22, 20

DB - northwind - Territories
Mapping Definition May 22, 20

Structured Data Sources

> **Note**
> Having full access to all features does not mean the Anzo Administrator has full access to all of the data in the system. Unlike the System Administrator (the **sysadmin** user), Anzo Administrators must still be granted access to specific entities.

The following image shows an example of the Anzo Administrator view of the Administration application.



Expand to view and change the server settings

Administrator — Set the System Administrator password.

Ports — Configure the ports to be used by the system.

Binary Store — Configure the binary store server options.

Email Server Configuration — Configure the SMTP server used to send email.

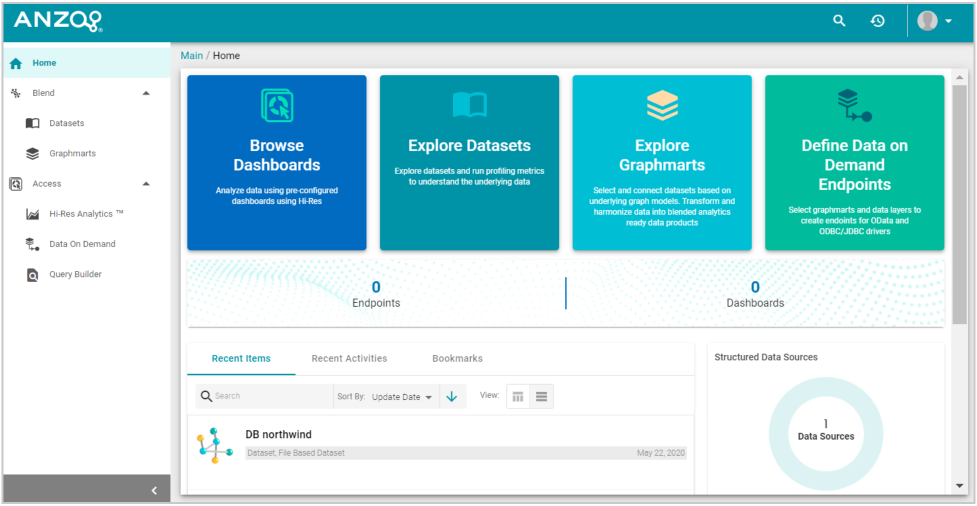
Home Pages — Configure the default root page served.

HTTP Session Management — Configure HTTP session options.

Anonymous User Access — Configure anonymous user access settings.

Data Analyst

By default the Data Analyst role has access to the Blend menu, Access menu, and Activity Log in the Anzo application. The image below shows an example of the view a user with the Data Analyst role has in the Anzo application.



Browse Dashboards
Analyze data using pre-configured dashboards using Hi-Res

Explore Datasets
Explore datasets and run profiling metrics to understand the underlying data

Explore Graphmarts
Select and connect datasets based on underlying graph models. Transform and harmonize data into blended analytics ready data products

Define Data on Demand Endpoints
Select graphmarts and data layers to create endoints for OData and ODBC/JDBC drivers

0 Endpoints

0 Dashboards

DB northwind
Dataset, File Based Dataset — May 22, 2020

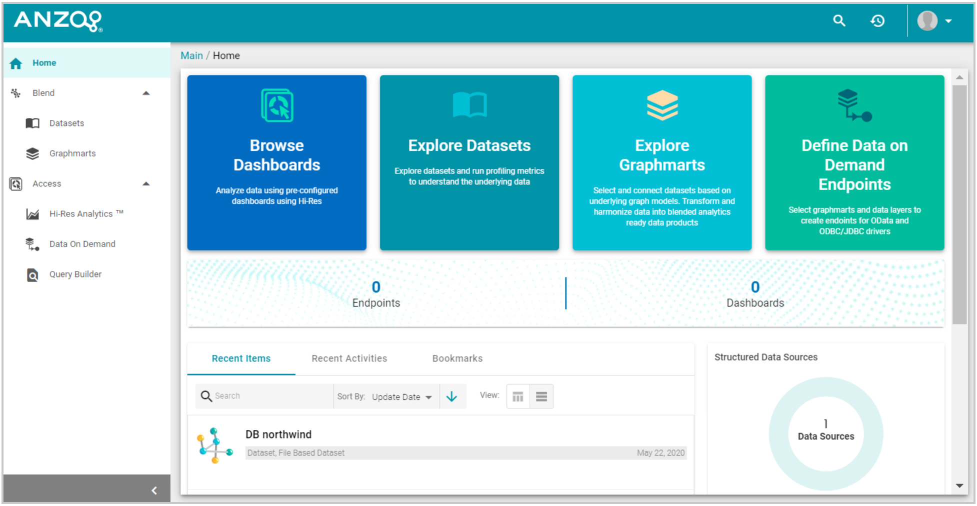
1 Data Sources

Members of the Data Analyst role can:

- View the Dataset catalog
- View and create graphmarts
- View and create Hi-Res Analytics dashboards
- View the Activity Log
- Access data with the Query Builder
- Create and access Data on Demand endpoints

Data Citizen

By default the Data Citizen role has access to the Blend menu, Access menu, and Activity Log in the Anzo application. The image below shows an example of the view a user with the Data Citizen role has in the Anzo application.



Analyze data using pre-configured dashboards using Hi-Res

Explore datasets and run profiling metrics to understand the underlying data

Select and connect datasets based on underlying graph models. Transform and harmonize data into blended analytics ready data products

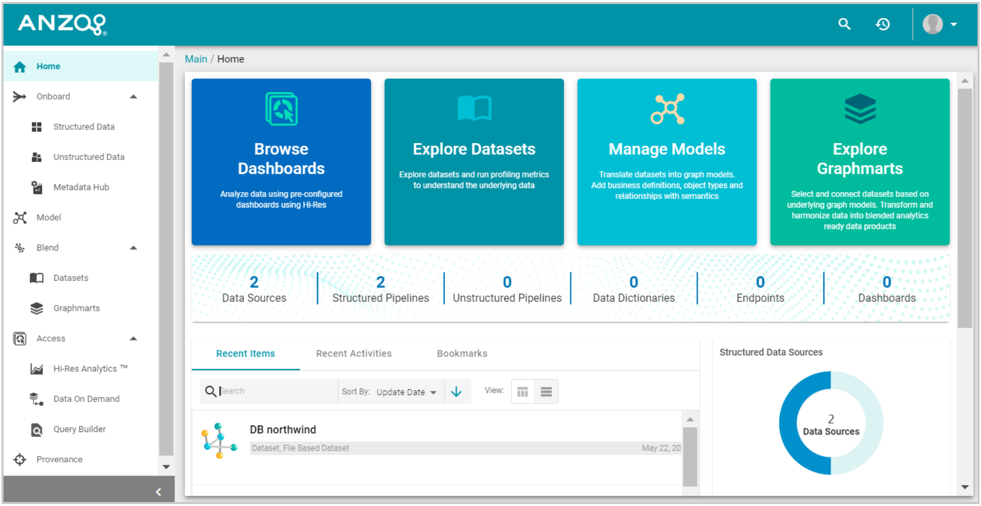
Select graphmarts and data layers to create endoints for OData and ODBC/JDBC drivers

Members of the Data Citizen role can:

- View the Dataset catalog
- View graphmarts
- View and create Hi-Res Analytics dashboards
- View the Activity Log
- Access data with the Query Builder
- Create and access Data on Demand endpoints

Data Curator

By default the Data Curator role has access to the Onboard menu, Model manager, Blend menu, Access menu, Provenance, and Activity Log in the Anzo application. The image below shows an example of the view a user with the Data Curator role has in the Anzo application.



Analyze data using pre-configured dashboards using Hi-Res

Explore datasets and run profiling metrics to understand the underlying data

Translate datasets into graph models. Add business definitions, object types and relationships with semantics

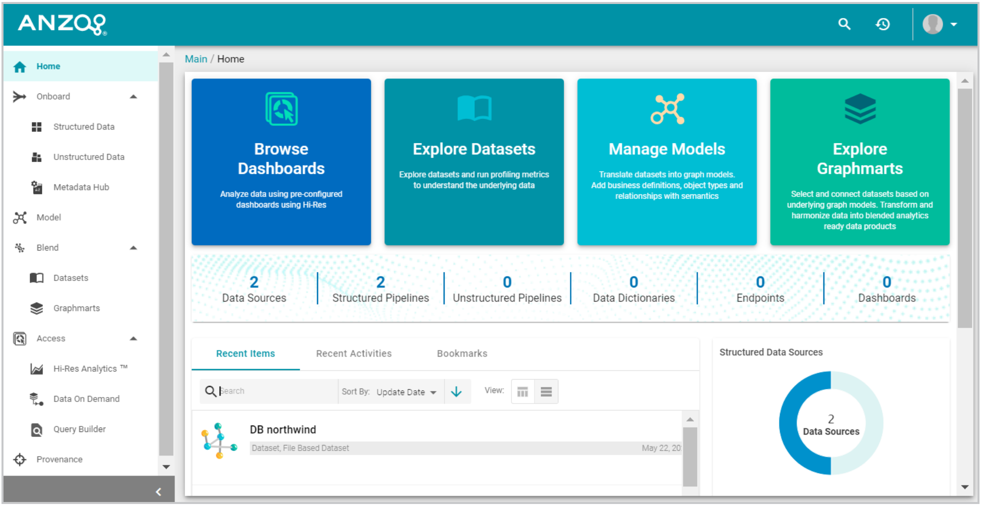
Select and connect datasets based on underlying graph models. Transform and harmonize data into blended analytics ready data products

Members of the Data Curator role can:

- Connect to data sources and onboard structured and unstructured data
- View and create data models, mappings, and pipelines
- View and create metadata dictionaries
- View the Dataset catalog
- View and create graphmarts
- View and create Hi-Res Analytics dashboards
- Manage the Query Blacklist Editor in the Hi-Res Analytics application
- View the Activity Log
- Access data with the Query Builder
- Create and access Data on Demand endpoints
- View data provenance

Data Governor

By default the Data Governor role has access to the Onboard menu, Model manager, Blend menu, Access menu, Provenance, and Activity Log in the Anzo application. The image below shows an example of the view a user with the Data Governor role has in the Anzo application.



Analyze data using pre-configured dashboards using Hi-Res

Explore datasets and run profiling metrics to understand the underlying data

Translate datasets into graph models. Add business definitions, object types and relationships with semantics

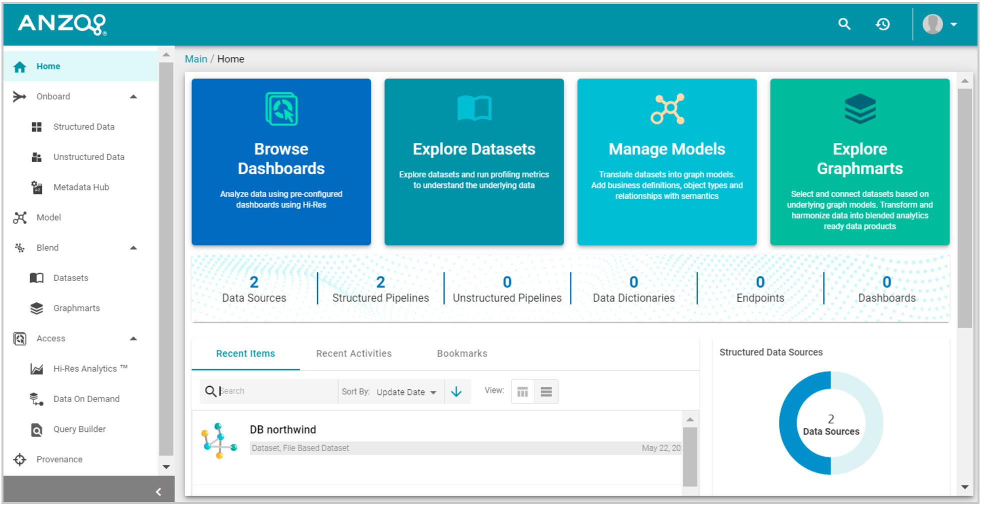
Select and connect datasets based on underlying graph models. Transform and harmonize data into blended analytics ready data products

Members of the Data Governor role can:

- Connect to data sources and onboard structured and unstructured data
- View and create data models, mappings, and pipelines
- View and create metadata dictionaries
- View the Dataset catalog
- View and create graphmarts
- View and create Hi-Res Analytics dashboards
- Manage the Query Blacklist Editor in the Hi-Res Analytics application
- View the Activity Log
- Access data with the Query Builder
- Create and access Data on Demand endpoints
- View data provenance

Data Scientist

By default the Data Scientist role has access to the Onboard menu, Model manager, Blend menu, Access menu, Provenance, and Activity Log in the Anzo application. The image below shows an example of the view a user with the Data Scientist role has in the Anzo application.



Analyze data using pre-configured dashboards using Hi-Res

Explore datasets and run profiling metrics to understand the underlying data

Translate datasets into graph models. Add business definitions, object types and relationships with semantics

Select and connect datasets based on underlying graph models. Transform and harmonize data into blended analytics ready data products

Members of the Data Scientist role can:

- Connect to data sources and onboard structured and unstructured data
- View and create data models, mappings, and pipelines
- View and create metadata dictionaries
- View the Dataset catalog
- View and create graphmarts
- View and create Hi-Res Analytics dashboards
- View the Activity Log
- Access data with the Query Builder
- Create and access Data on Demand endpoints
- View data provenance

To review the specific permissions for each role, select **Roles** in the **User Management** menu in the Admin application. Click a role to open the Edit dialog box and review the permissions. For more information about the permissions, see Role Permissions Reference.

Related Topics

User Management and Access Control Concepts

Creating and Managing Roles

Role Permissions Reference

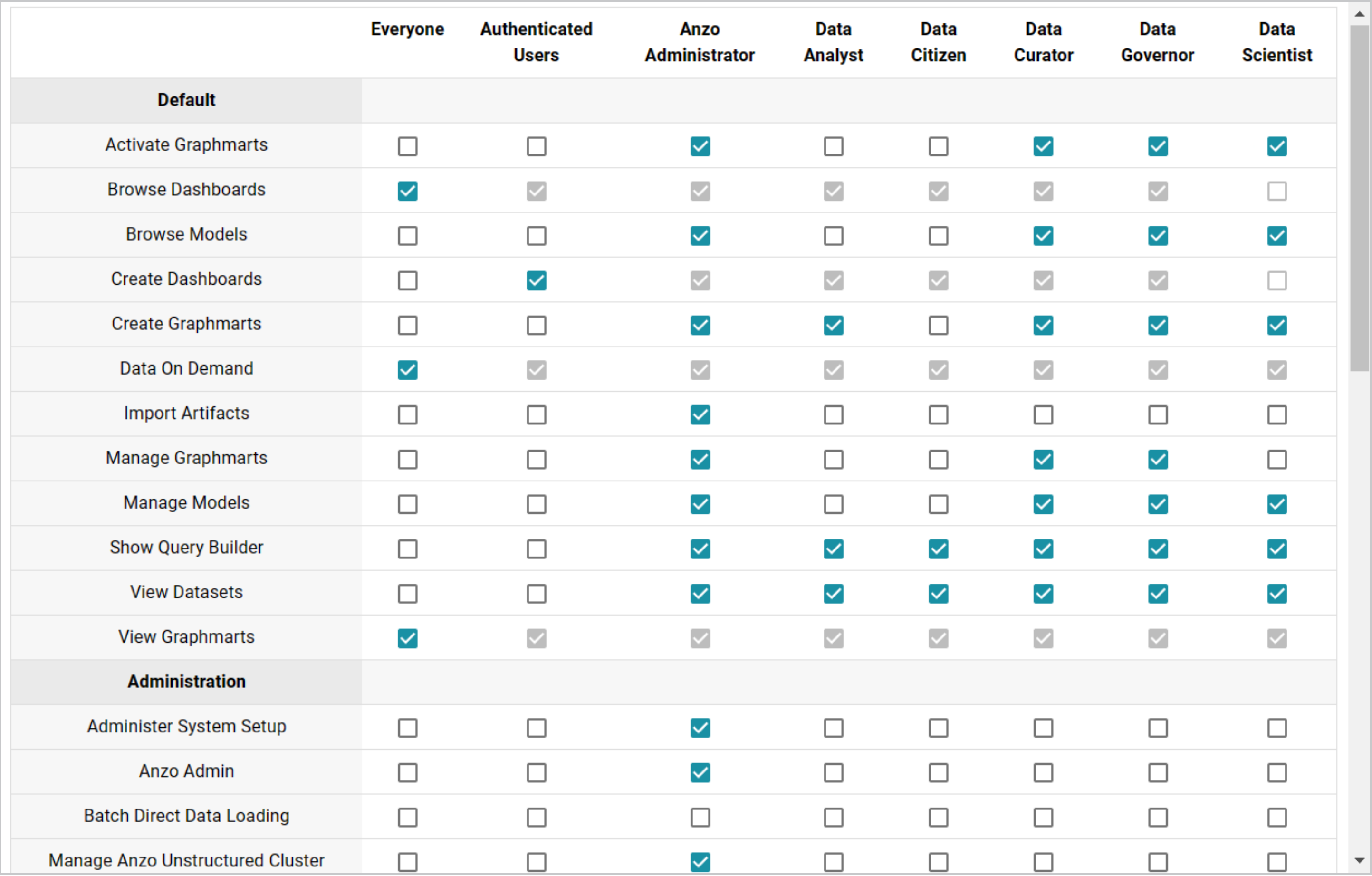
Role Permissions Reference

This topic provides details about each of the permissions that can be applied to roles. These permissions grant access to functionality, i.e., the menus and screens in the Anzo and Administration applications. For example, role permissions determine whether a member of a role can access the **Onboard** menu and create a new data source or see the **Blend** menu and create a new graphmart. Whether a member can view, modify, or delete a data source or graphmart artifact that is created by someone else, however, is controlled by the user or group permissions that are applied at the artifact level.

> **Tip**
> For more information about artifact-level permissions, see Artifact Access Control Concepts. And for more information about roles versus users and groups, see User Management Concepts.

Permissions Overview Screen

To view an overview of the configured permissions for all Anzo roles, you can view the **Permissions** page under the **User Management** menu in the Administration application. The screen displays a table; the heading row lists each role, and the first column lists each permission. The permissions are grouped into categories, such as Application or Data Onboarding. The rows for each role column include checkboxes that control permissions. You can select or clear checkboxes to enable or disable permissions for a role. For example:



	Everyone	Authenticated Users	Anzo Administrator	Data Analyst	Data Citizen	Data Curator	Data Governor	Data Scientist
Default								
Activate Graphmarts	□	□	☑	□	□	☑	☑	☑
Browse Dashboards	☑	☑	☑	☑	☑	☑	☑	□
Browse Models	□	□	☑	□	□	☑	☑	☑
Create Dashboards	□	☑	☑	☑	☑	☑	☑	□
Create Graphmarts	□	□	☑	☑	□	☑	☑	☑
Data On Demand	☑	☑	☑	☑	☑	☑	☑	☑
Import Artifacts	□	□	☑	□	□	□	□	□
Manage Graphmarts	□	□	☑	□	□	☑	☑	□
Manage Models	□	□	☑	□	□	☑	☑	☑
Show Query Builder	□	□	☑	☑	☑	☑	☑	☑
View Datasets	□	□	☑	☑	☑	☑	☑	☑
View Graphmarts	☑	☑	☑	☑	☑	☑	☑	☑
Administration								
Administer System Setup	□	□	☑	□	□	□	□	□
Anzo Admin	□	□	☑	□	□	□	□	□
Batch Direct Data Loading	□	□	□	□	□	□	□	□
Manage Anzo Unstructured Cluster	□	□	☑	□	□	□	□	□

Permission Descriptions

The tables below list the permissions in each category and describe the pages and menus that are enabled for members of a role where that permission is applied.

> **Note**
> The permissions described below give access to functionality in the Anzo and Administration applications. Whether members of the role have view or edit access to certain datasets, models, dashboards, graphmarts, etc. depends on the permissions that are granted at the artifact level.

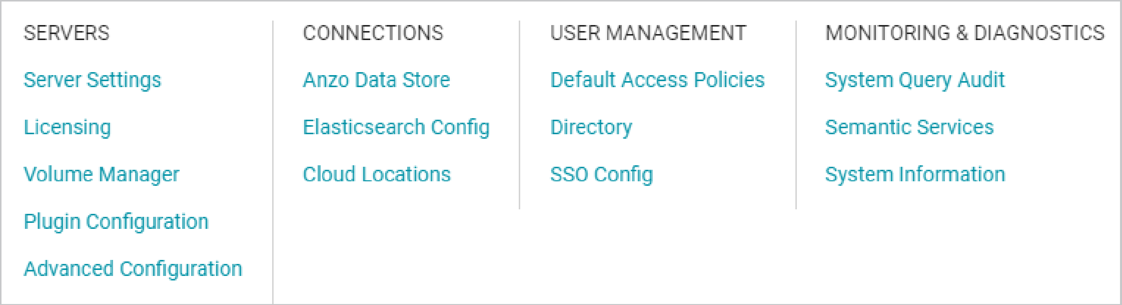
Default

Permission	Description
Activate Graphmarts	If the user has the appropriate permissions at the graphmart level, this permission allows them to activate and deactivate the graphmarts and import graphmarts into Anzo. It does not give permission to create new graphmarts or delete graphmarts.

Permission	Description
	To be able to access a graphmart screen in the Anzo application and move the **Inactive** → **Active** slider, the **Anzo Application** permission also needs to be applied.
Browse Dashboards	Gives permission to view existing dashboards in the Hi-Res Analytics application. Does not give permission to create new dashboards.
Browse Models	Gives permission to view existing data models. Applying this permission exposes the **Models** menu item in the Anzo application. Must also have the **Anzo Application** permission to access the Anzo application.
Create Dashboards	Gives permission to create dashboards in the Hi-Res Analytics application. Applying this permission also exposes the **Create Dashboard** button on the graphmart screens in the Anzo application when the user has the **Anzo Application** permission.
Create Graphmarts	Gives permission to create new graphmarts. Applying this permission exposes the **Add Graphmart** button on the graphmarts screen. Must also have the **Anzo Application** permission to create graphmarts in the application.
Data on Demand	If the user has the appropriate permissions at the graphmart level, this permission enables the user to create Data on Demand endpoints. Applying this permission enables the **Create New Endpoint** button on the Data on Demand tab for graphmarts. Must also have the **Anzo Application** permission to access the application.
Import Artifacts	Gives permission to perform Import operations from the Anzo application. If a user is a member of a role that has **Import Artifact** assigned, they will see the **Import** option in the menu when they click the **Add** button to add a data source, dataset, model, etc. Must also have the **Anzo Application** permission.
Manage Graphmarts	Gives permission to manage permissions for graphmarts. Must also have the **Anzo Application** permission to access the graphmart screens.
Manage Models	Gives permission to create and import models. Must also have the **Anzo Application** permission to access the Model screen.

Permission	Description
Show Query Builder	Gives permission to find data and run SPARQL queries using the Query Builder. Applying this permission exposes the **Query Builder** option in the **Access** menu. Must also have the **Anzo Application** permission.
View Datasets	Gives permission to view the Datasets catalog. Applying this permission exposes the **Datasets** option in the **Blend** menu in the Anzo application. Must also have the **Anzo Application** permission.
View Graphmarts	Gives permission to view the list of existing graphmarts. Must also have the **Anzo Application** permission to view the graphmarts screen in the Anzo application.

Administration

Permission	Description
Administer System Setup	Gives permission to access the options in the Administration application that are related to system setup, such as **Server Settings**, **Licensing**, **Anzo Data Store**, and **Directory** server configuration. The image below shows the view of the Administration menu that users have if **Administer System Setup** and **Anzo Application** are the only two applied permissions:  SERVERS Server Settings Licensing Volume Manager Plugin Configuration Advanced Configuration CONNECTIONS Anzo Data Store Elasticsearch Config Cloud Locations USER MANAGEMENT Default Access Policies Directory SSO Config MONITORING & DIAGNOSTICS System Query Audit Semantic Services System Information **Note** Some menu items in the above image, such as **Semantic Services**, **AnzoGraph**, and **Anzo Data Store**, are also controlled by more granular permissions: **Manage Semantic Services**, **Manage AnzoGraph**, and **Create Anzo Data Stores**. To give an administrator full create, modify, and

Permission	Description
	delete access to those functions, the granular permissions need to be enabled in addition to **Administer System Setup**.
Anzo Admin	The **Anzo Admin** permission is a legacy permission that granted access to the Admin application that existed in pre-5.1 versions of Anzo. This permission no longer controls access to administrative functions and will be removed in an upcoming release.
Batch Direct Data Loading	Gives permission to create a graphmart from multiple data sources at once when ingesting sources via graphmarts. For more information, see Directly Loading Data Sources via Graphmarts in the User Guide.
Manage Anzo Unstructured Cluster	Gives permission to view and create connections to Anzo Distributed Unstructured clusters.
Manage AnzoGraph	Gives permission to view and create AnzoGraph connections. Does not give permission to delete connections or change the configuration of an existing connection. **Administer System Setup** is required to grant permission to delete and change existing AnzoGraph connections.
Manage Certificates	Gives permission to upload and delete server certificates.
Manage ETL Engines	Gives permission to add new ETL engine connections and delete or change the configuration of existing connections.
Manage File Stores	Gives permission to create new File Store connections and view existing connections. Does not grant permission to delete or change existing file store connections. The **Administer System Setup** permission is required in conjunction with **Manage File Stores** to be able to delete or edit existing file stores.
Manage Query Blocklists	Gives permission to create and remove queries from the **Query Blocklist** tab in the **System Query Audit** Log.

Permission	Description
	Note If a user only has the **Manage Query Blocklist** permission, the Administration menu is not available. Use this permission in conjunction with **Administer System Setup** to grant access to System Query Audit and the Query Blocklist.
Manage Semantic Services	Gives permission to stop and start Semantic Services from the Semantic Services screen as well as view details about the services and use the Service Builder to generate and run semantic service requests. Note If a user only has the **Manage Semantic Services** permission, the Administration menu is not available. Use this permission in conjunction with **Administer System Setup** to grant access to the Semantic Services screen.
Manage Users, Groups, and Roles	Gives permission to create, change, and delete users, groups, and roles. A user who has this permission has Admin level access to all users, groups, and roles.
Profile Data	Gives permission to Profile data sources, datasets, and graphmarts. Applying this permission exposes the **Profile Data** button on the data source, datasets, and graphmart screens.
Use Experimental Anzo Features	Grants permission use experimental Anzo features. Experimental features are recently implemented and may not be reliable for production use.
View Activity Logs	Gives permission to view the Activity Log. Applying this permission exposes the Activity Log icon (🕘) in the top menu bar of the Anzo and Administration applications. The **Anzo Application** permission is needed to give access to the Anzo application.
View Log Files	Gives permission to view and download log files from the Log Files tab. Does not grant

Permission	Description
	permission to change logging levels or add new log packages. Use this permission in conjunction with **Administer System Setup** to grant access to configure log levels and packages.

Application

Permission	Description
Anzo Application	Grants access to the main Anzo application.
Anzo CLI	Gives permission to use the administration command line interface.
Anzo for Excel	Gives permission to open, edit, and create mappings using the Anzo for Office Excel plugin.
Hi-Res Analytics	Grants access to the Hi-Res Analytics application.

Data Onboarding

Permission	Description
Create Anzo Data Stores	Gives permission to create Anzo Data Stores. Must also have the **Administer System Setup** permission to make the **Anzo Data Store** option available in the Administration application.
Create Data Sources	Gives permission to add new data sources. Does not give permission to delete existing sources. Must also have the **Anzo Application** and **Onboard Structured Data** permissions to access the Data Sources screen and add new sources.
Manage Dictionaries	Gives permission to view, edit, and create metadata dictionaries. Applying this permission exposes the **Metadata Hub** option in the Onboard menu. Must also have the **Anzo Application** permission to access the application.

Permission	Description
Onboard Structured Data	Gives permission to access the **Onboard** > **Structured Data** menu. Must also have the **Anzo Application** permission.
Onboard Unstructured Data	Gives permission to create unstructured pipelines. Applying this permission exposes the **Onboard** > **Unstructured Data** menu. Must also have the **Anzo Application** permission.

Migration

Permission	Description
Manage Migration Packages	Gives permission to create, export, and import migration packages that include artifacts the user has access to.
Perform Migration Package Operations As Sysadmin	Gives permission to create, export, and import migration packages with sysadmin privileges. That means the package can include artifacts the user may not otherwise have permission to access.

Related Topics

User Management and Access Control Concepts

Creating and Managing Roles

Predefined Anzo Roles and Permissions

Managing Default Access Policies

Default Access Policies are the security policies that are applied by default to the artifacts that are stored in a particular **registry**. A registry is a system-level graph that stores metadata about artifacts of the same type. For example, metadata about all of your data source artifacts is stored in a Data Sources Registry, and metadata about all of your data model artifacts is stored in an Ontology Registry. A Default Access Policy defines the base permissions to assign to a type of artifact when it is created—before permission inheritance and user-configured sharing is applied.

> **Note**
> Any **Permission Inheritance** that is applied by Anzo and artifact-level **Sharing** that is configured by users is applied to artifacts in addition to the permissions supplied by the Default Access Policy. For more information about permission inheritance and artifact sharing, see Artifact Access Control Concepts.

This topic provides information about the permission sets that can be assigned to users and groups and describes the default access policies for each registry. This topic also includes instructions for changing access policies.

- Default Access Policy Permissions Reference
- Default Access Policy Reference
- Configuring Default Access Policies

Default Access Policy Permissions Reference

Default access policies use the same predefined permission sets and mechanism for assigning permissions as other artifacts in the Anzo application (see Sharing Access to Artifacts in the User Guide for more information).

There are three predefined permission sets that include a combination of six permissions that can be assigned to the creator of an artifact and other users and groups.

The table below lists the predefined permission sets and describes the privileges that are granted for each permission that is part of the set:

Set	Permission	Allows a user to:
View	View	• See an artifact in the Anzo application. • Create versions of the artifact.
	Meta View	• Relates only to an artifact's permissions. A user with Meta View can see the permissions on the artifact's **Sharing** tab but they cannot change permissions.
Modify	In addition to the **View** and **Meta View** permissions described above, the **Modify** set includes the **Add/Edit** and **Delete** permissions described below.	
	Add/Edit	• Change an artifact, such as to rename it or edit its description. • Add a related entity to an artifact. For example, add a schema to a data source or a layer to a graphmart.
	Delete	• Remove a related entity from the artifact. For example, delete a layer from a graphmart or a schema from a data source. • Does not give permission to remove the

Set	Permission	Allows a user to:
		parent artifact. For example, a user can remove a schema from a source but cannot delete the data source.
Admin	In addition to the **View**, **Meta View**, **Add/Edit**, and **Delete** permissions described above, the **Admin** set includes the **Meta Add/Edit** and **Meta Delete** permissions described below.	
	Meta Add/Edit	• Relates only to an artifact's permissions. A user with Meta Add/Edit can add permissions to a user or group. They cannot remove permissions from any user or group.
	Meta Delete	• Remove permissions from a user or group. • Delete the parent artifact and its related entities.

Default Access Policy Reference

There is a configurable Default Access Policy for several of the Anzo registries. To see and manage the Default Access Policies, go to the Administration application, expand the **User Management** menu, and click **Default Access Policies**.

> **Important**
> Never modify any of the Anzo registries. Changing or removing a registry can irreparably damage your Anzo server.

The sections below provide details about each of the registries for which you can configure Default Access Policies:

- Data Sources Registry
- Elastic Search Configuration Registry
- Global Linked Data Configuration
- Graphmarts Registry
- Linked Data Set Registry
- Ontology Registry
- Orchestration Configuration Registry
- Query Builder Registry
- Role and Permissions Registry
- SDI Registry

Data Sources Registry

The **Data Sources Registry** is the system graph that stores metadata about all of the **File Store**, **Anzo Data Store**, **Data Source**, and **Schema** artifacts that have been created in Anzo. Since data sources and schemas have a fundamental relationship in that schemas are derived or imported from data sources, one registry stores metadata about both types of artifacts. The Data Sources Registry access policy is applied by default when a user creates a data source or an Anzo Data Store.

Default Permissions Configuration

- The **Creator** of a source is assigned the Admin permission set for that source and the associated schemas. In addition, the Creator of an Anzo Data Store is also assigned the Admin permission set for that data store.

- The **Everyone** role is assigned the View permission set for a new source and its schemas. The Everyone role is also assigned the View permission set for any Anzo Data Stores.
- The **Creator Default Group** is assigned the Modify permission set for new source, schema, and Anzo Data Store artifacts.

Elastic Search Configuration Registry

The **Elastic Search Configuration Registry** is the system graph that stores metadata about all of the **Elasticsearch** connection artifacts in Anzo. This access policy is applied by default when an Elasticsearch connection is created.

Default Permissions Configuration

- The **Creator** of an Elasticsearch connection is assigned the Admin permission set for that artifact.
- The **Everyone** role is assigned the View permission set for that Elasticsearch connection artifact.
- The **Creator Default Group** is assigned the Modify permission set for that artifact.

Global Linked Data Configuration

The **Global Linked Data Configuration Registry** is a global policy that applies to all artifacts created in Anzo—unless another Default Access Policy (such as the Data Sources Registry, Graphmarts Registry, or Ontology Registry) applies.

> **Example**
> If a user created a model and the Ontology Registry Default Access Policy was removed or unset, the Global Linked Data Configuration access policy would be applied to that model artifact.

Default Permissions Configuration

- The **Creator** of an artifact that follows this policy is assigned the Admin permission set for that artifact.
- The **Creator Default Group** is assigned the Modify permission set for that artifact.

Graphmarts Registry

The **Graphmarts Registry** is a system graph that stores metadata about all of the **Graphmart** artifacts in Anzo. All graphmarts inherit permissions from the Graphmarts Registry Default Access Policy. In addition, since data layers and steps are created in the context of a graphmart, they inherit permissions from the graphmart by default.

Default Permissions Configuration

- The **Creator** of a graphmart is assigned the Admin permission set for that artifact.
- The **Everyone** role is assigned the View permission set for that graphmart.
- The **Creator Default Group** is assigned the Modify permission set for the graphmart.

Linked Data Set Registry

The **Linked Data Set Registry** is a system graph that stores metadata about all of the linked data sets, notably the File-Based Linked Data Sets (FLDS) that are listed in the Datasets catalog. This includes datasets that are generated from unstructured pipelines as well as datasets that are created by users, such as empty datasets, dataset from Export Steps, and Existing RDF imports directly to the Datasets catalog.

Default Permissions Configuration

FLDS artifacts inherit from the workflow that created it. If raw RDF files are imported to the catalog or an empty dataset is created, the Linked Data Set Registry Default Access policy is applied to the resulting FLDS artifact.

Ontology Registry

The **Ontology Registry** is the system graph that stores metadata about all of the model artifacts in Anzo. This access policy is applied by default if a model is imported or manually created by a user. When a model is generated from an unstructured pipeline or the automated Direct Data Load workflow, however, the model inherits the permissions from the related data source.

Default Permissions Configuration

- The **Creator** of a model is assigned the Admin permission set for that artifact.
- The **Everyone** role is assigned the View permission set for that model.
- The **Creator Default Group** is assigned the Modify permission set for that artifact.

Orchestration Configuration Registry

The **Orchestration Configuration Registry** is a system graph that stores metadata about workflows. This access policy is applied by default when a workflow is created.

Default Permissions Configuration

- The **Anzo Administrator** is assigned the Admin permission set for the artifact.
- The **Creator** of a workflow that follows this policy is assigned the Admin permission set for that artifact.
- The **Creator Default Group** is assigned the Modify permission set for that artifact.

Query Builder Registry

The **Query Builder Registry** is a system graph that stores metadata about saved Query Builder queries. This access policy is applied by default when a new query is saved.

Default Permissions Configuration

The user who saves a query is assigned the Admin permission set. By default, saved queries are unique to each creator, and other users do not see the creator's queries.

Role and Permissions Registry

The **Role and Permissions Registry** is a system graph that stores metadata about roles and permissions. Roles are not treated like other artifacts in Anzo. Unlike a data source, model, or graphmart artifact, for example, the permissions for a single role or subset of roles cannot be configured separately. Access to create and edit roles is controlled by the **Manage Users, Groups, and Roles** permission. For more information, see Role Permissions and Registries.

Default Permissions Configuration

- The **System Administrator** is assigned the Admin permission set for all role artifacts.
- The **Everyone** role is assigned the View permission set for all role artifacts.
- A member of a role that is assigned the **Manage Users, Groups, and Roles** permission has the Admin permission set for all role artifacts.

SDI Registry

The **SDI Registry** is a legacy system graph that stored metadata about the mapping, pipeline, and job artifacts that were manually created by a user.

Default Permissions Configuration

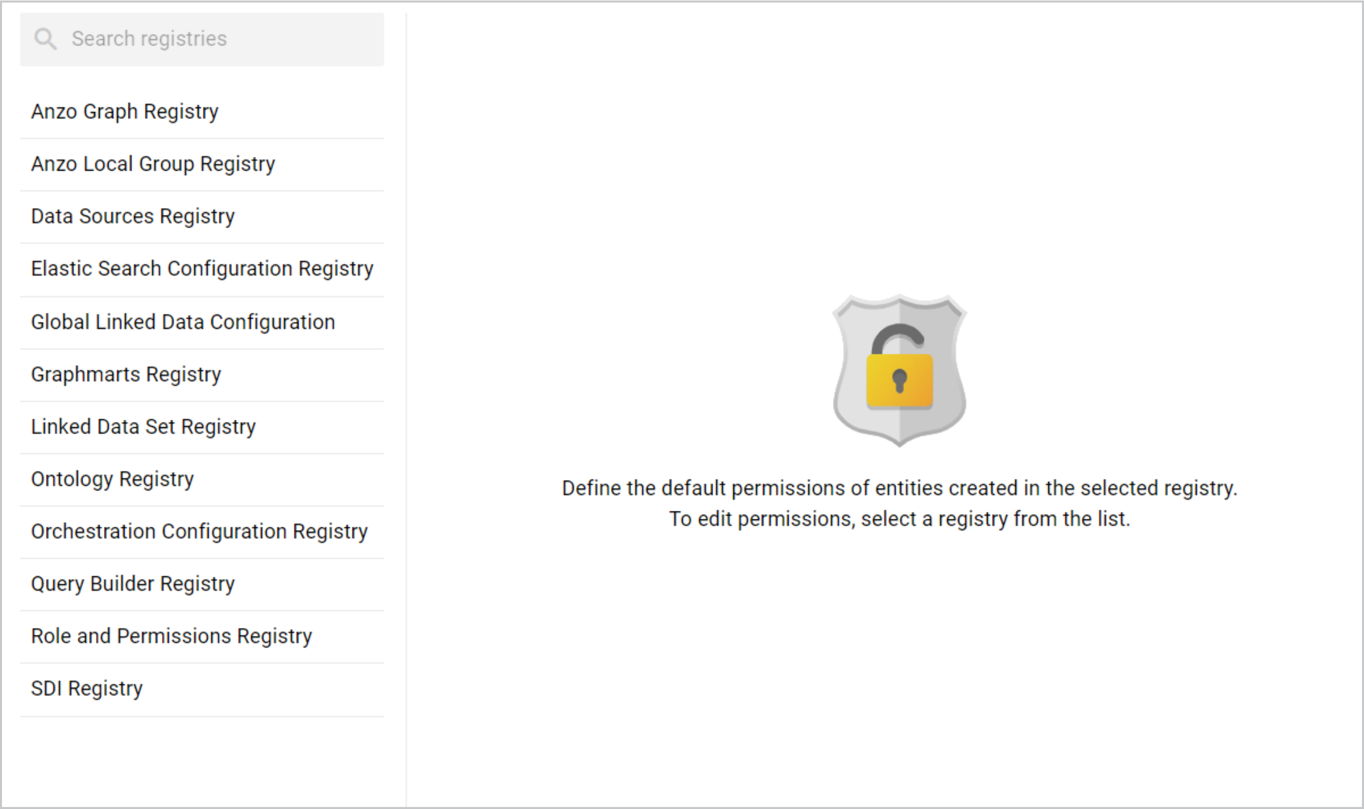
- The **Creator** of a mapping, pipeline, or job is assigned the Admin permission set for that artifact.
- The **Everyone** role is assigned the View permission set for the new artifact.
- The **Creator Default Group** is assigned the Modify permission set for that artifact.

Configuring Default Access Policies

Follow the instructions below to change the default access policy for a registry.

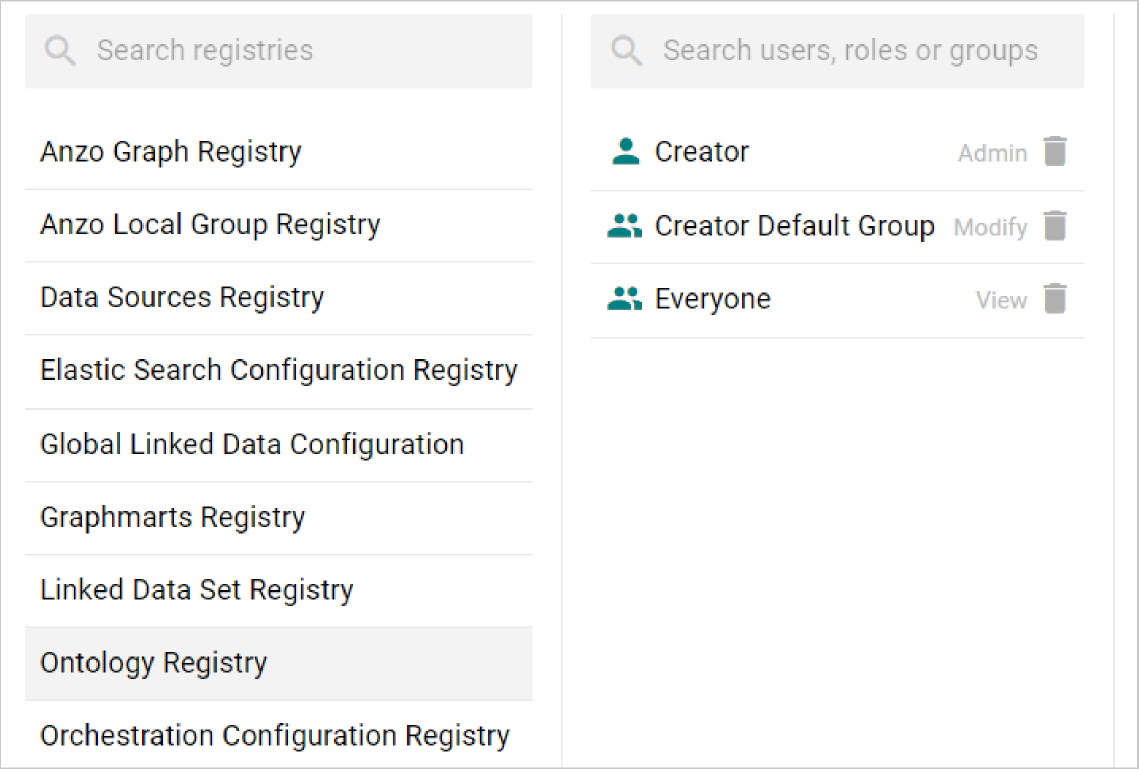
> **Important**
> Changing default access control policies does not change permissions on any existing artifacts. The changes affect only new artifacts that are created after the change.

1. In the Administration application, expand the **User Management** menu and click **Default Access Policies**. The Default Access Policies screen is displayed.


Search registries
Anzo Graph Registry
Anzo Local Group Registry
Data Sources Registry
Elastic Search Configuration Registry
Global Linked Data Configuration
Graphmarts Registry
Linked Data Set Registry
Ontology Registry
Orchestration Configuration Registry
Query Builder Registry
Role and Permissions Registry
SDI Registry

Define the default permissions of entities created in the selected registry.
To edit permissions, select a registry from the list.

2. On the left side of the screen, select the access policy that you want to configure. The current configuration for that policy is shown on the right side of the screen. For example, the image below shows the Ontology Registry. The model creator has **Admin** permissions, the Everyone role has **View** permissions, and the Creator Default Group has **Modify** permissions.



Search registries

Anzo Graph Registry

Anzo Local Group Registry

Data Sources Registry

Elastic Search Configuration Registry

Global Linked Data Configuration

Graphmarts Registry

Linked Data Set Registry

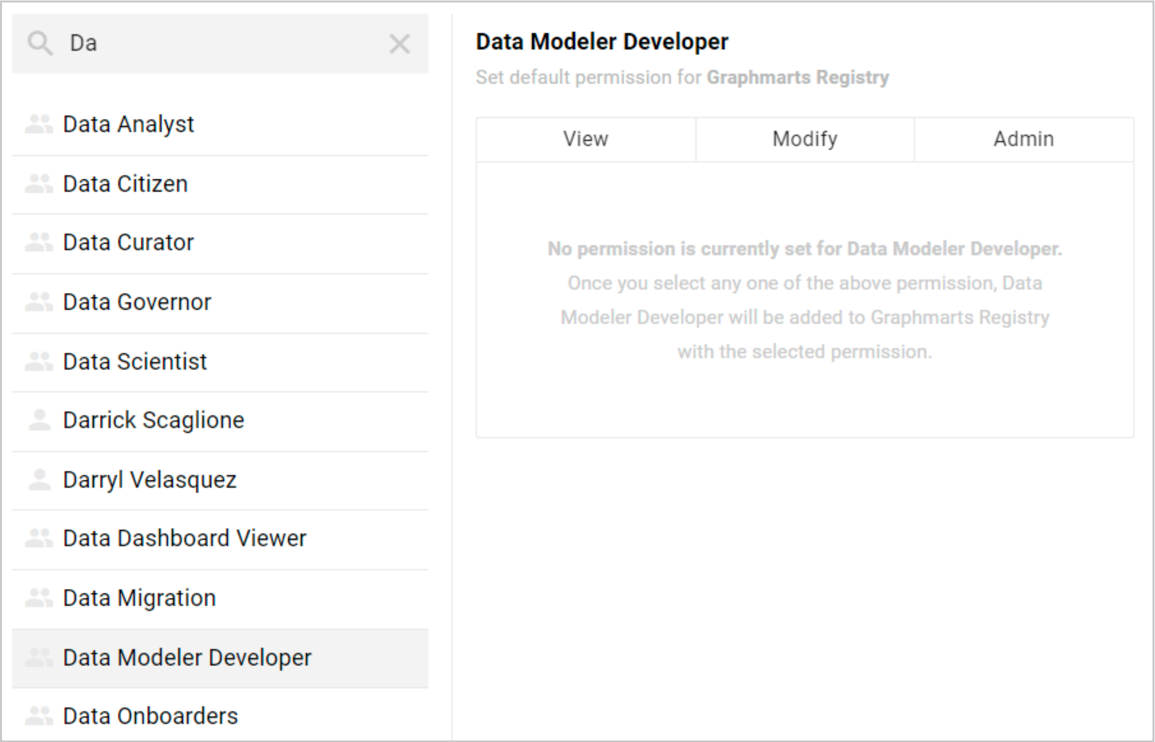
Ontology Registry

Orchestration Configuration Registry

Search users, roles or groups

Creator	Admin
Creator Default Group	Modify
Everyone	View

3. To change a configured user or group, select a name in the list to view the permissions on the right side of the screen. To add a user or group, type a term in the **Search** field. Then select a name in the result list to view the permissions details. For example, the image below shows the search results for additional groups and selects the Data Modeler Developer group:



Data Analyst

Data Citizen

Data Curator

Data Governor

Data Scientist

Darrick Scaglione

Darryl Velasquez

Data Dashboard Viewer

Data Migration

Data Modeler Developer

Data Onboarders

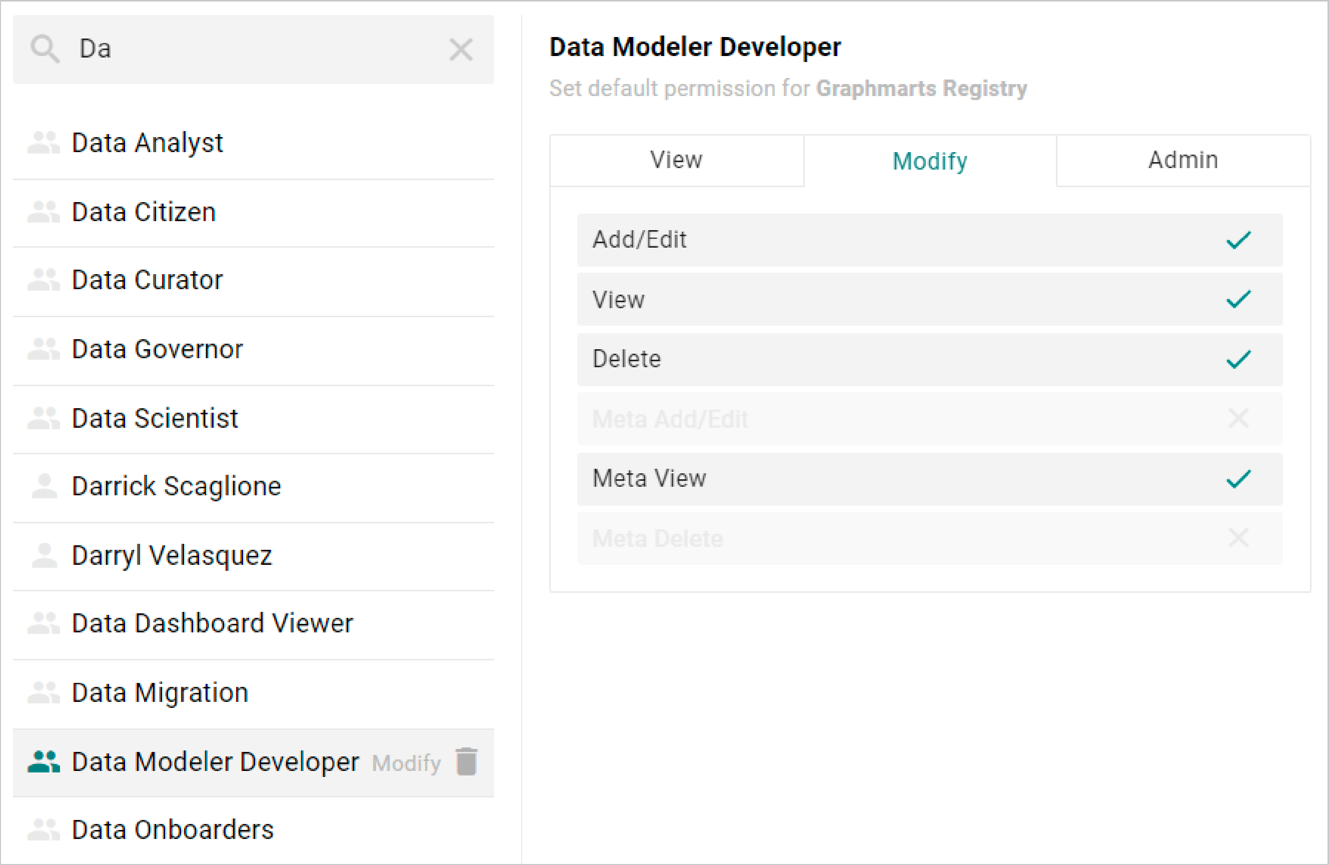
Data Modeler Developer

Set default permission for **Graphmarts Registry**

No permission is currently set for Data Modeler Developer. Once you select any one of the above permission, Data Modeler Developer will be added to Graphmarts Registry with the selected permission.

> **Note**
>
> Though Anzo is flexible and allows you to assign default access policies to roles, the recommendation is to control access to artifacts in a registry with users and groups. For more information, see User Management Concepts.

4. On the right side of the screen, click the tab for the predefined permission set that you want to assign to the selected user or group. For information about the permission sets, see Default Access Policy Permissions Reference above. For example, the image below assigns the **Modify** permission set to the Data Modeler Developer group.



Da

- Data Analyst
- Data Citizen
- Data Curator
- Data Governor
- Data Scientist
- Darrick Scaglione
- Darryl Velasquez
- Data Dashboard Viewer
- Data Migration
- Data Modeler Developer Modify
- Data Onboarders

Data Modeler Developer

Set default permission for **Graphmarts Registry**

View | Modify | Admin

Add/Edit	✓
View	✓
Delete	✓
Meta Add/Edit	✗
Meta View	✓
Meta Delete	✗

> **Tip**
>
> To clear permissions for a user or group, click the trashcan icon (🗑) next to the user, role, or group name.

5. To configure additional users or groups, select the name and then repeat the step above to apply a permission set. Changes to access control policies are automatically saved.

Related Topics

User Management and Access Control Concepts

Monitoring and Diagnostics

The topics in this section provide information about monitoring events and managing Anzo and AnzoGraph diagnostic files.

Related Topics

Enabling and Configuring the System Monitor Service

Viewing the Current Stack in a Browser

Managing Anzo Logging

The topics in this section provide general information about logging in Anzo, instructions for adding logging for new components, changing the level or type of information that is logged, and reviewing log files. This section also provides guidance on enabling the recommended Log Packages.

Introduction to Anzo Logging

This topic provides an introduction to Anzo logging concepts, an overview of the Logging user interface, and information about the type of logging that is enabled by default. It also gives a high-level overview about adding new logging, adjusting the level of information that is logged, and reviewing log files.

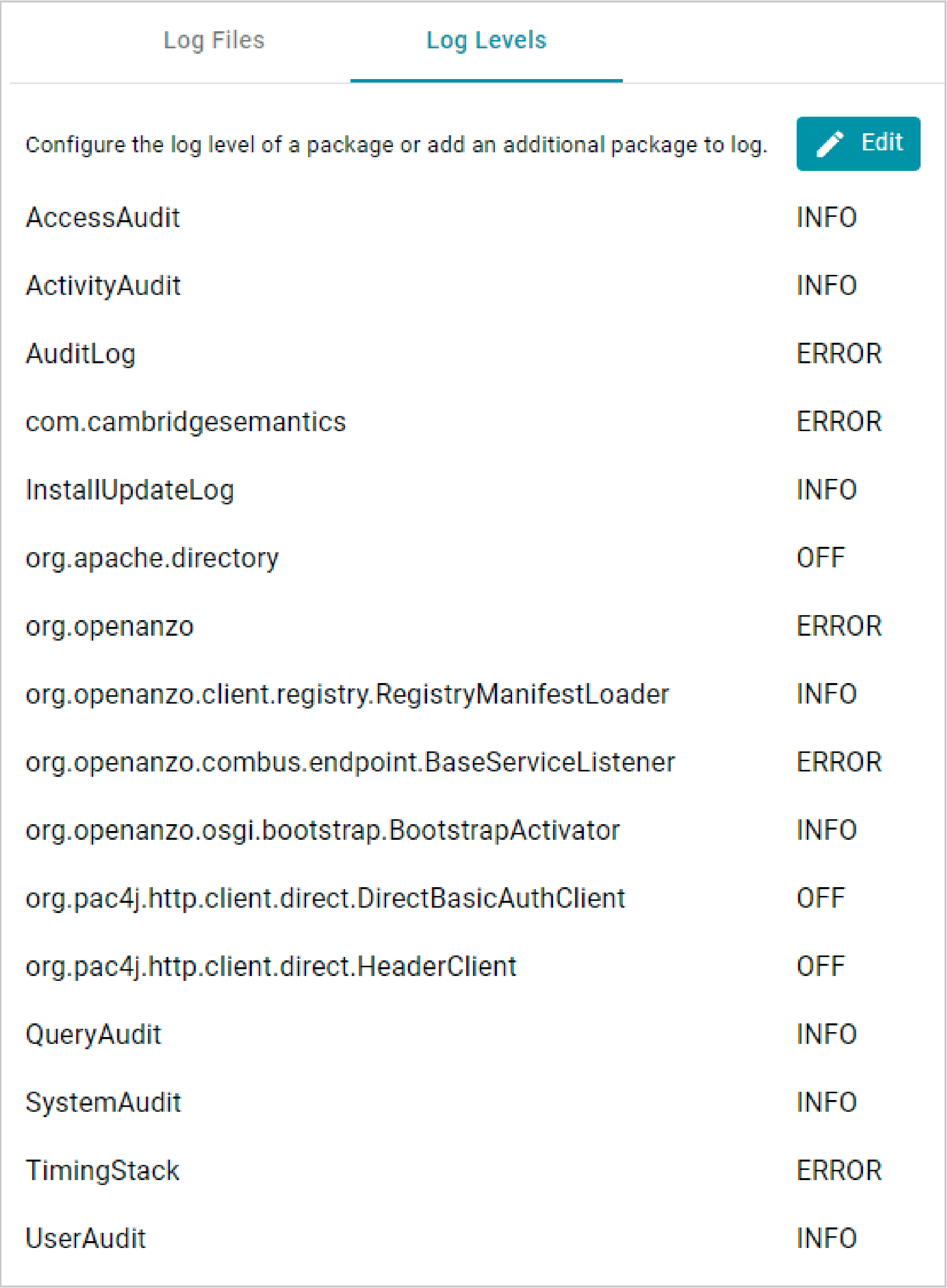
- Logging Concepts
- Default Logging Configuration
- Adding Log Packages
- Log Level Definitions
- Viewing Log Files

Logging Concepts

In order to give users granular control and flexibility over the type and breadth of information that is captured, the concept of **Log Packages** is integral to logging in Anzo. A Log Package is a listener for events that are related to a particular Semantic Service or component, such as core system, LDAP, Anzo Unstructured, or AnzoGraph events. To give users flexibility over the depth of information that is logged, each Log Package can be configured to capture events at a certain **Log Levels**, from all events to fatal events only.

Default Logging Configuration

Logging is managed in the Administration application. To view the Log Packages that are enabled for your server, expand the **Monitoring & Diagnostics** menu in the Administration application and click **Logging**. Then click the **Log Levels** tab to show the enabled Log Packages and their Log Level configuration. For example, the image below shows the default configuration for a new installation:



Configure the log level of a package or add an additional package to log.

AccessAudit	INFO
ActivityAudit	INFO
AuditLog	ERROR
com.cambridgesemantics	ERROR
InstallUpdateLog	INFO
org.apache.directory	OFF
org.openanzo	ERROR
org.openanzo.client.registry.RegistryManifestLoader	INFO
org.openanzo.combus.endpoint.BaseServiceListener	ERROR
org.openanzo.osgi.bootstrap.BootstrapActivator	INFO
org.pac4j.http.client.direct.DirectBasicAuthClient	OFF
org.pac4j.http.client.direct.HeaderClient	OFF
QueryAudit	INFO
SystemAudit	INFO
TimingStack	ERROR
UserAudit	INFO

Default Log Packages

The table below describes Log Packages that are enabled by default as well as their default Log Level. Log Levels are defined in Log Level Definitions below.

Package	Level	Description
AccessAudit	Info	Listener for access audit events such as user login

Package	Level	Description
		attempts.
ActivityAudit	Info	Listener for activity audit events.
AuditLog	Error	Logger for audit events when the appropriate packages are enabled. For more information, see Enabling and Viewing Audit Logs.
com.cambridgesemantics	Error	Like the org.openanzo package, this base package listens for core system events. Changing the Log Level of this package affects logs across Anzo components and services.
InstallUpdateLog	Info	Listener for installation and upgrade events. Captures information about bundle imports and updates.
org.apache.directory	Off	Listener for events related to the underlying internal LDAP server. **Do not modify the Log Level for this package**.
org.openanzo	Error	Like the com.cambridgesemantics package, this base package listens for core system events.

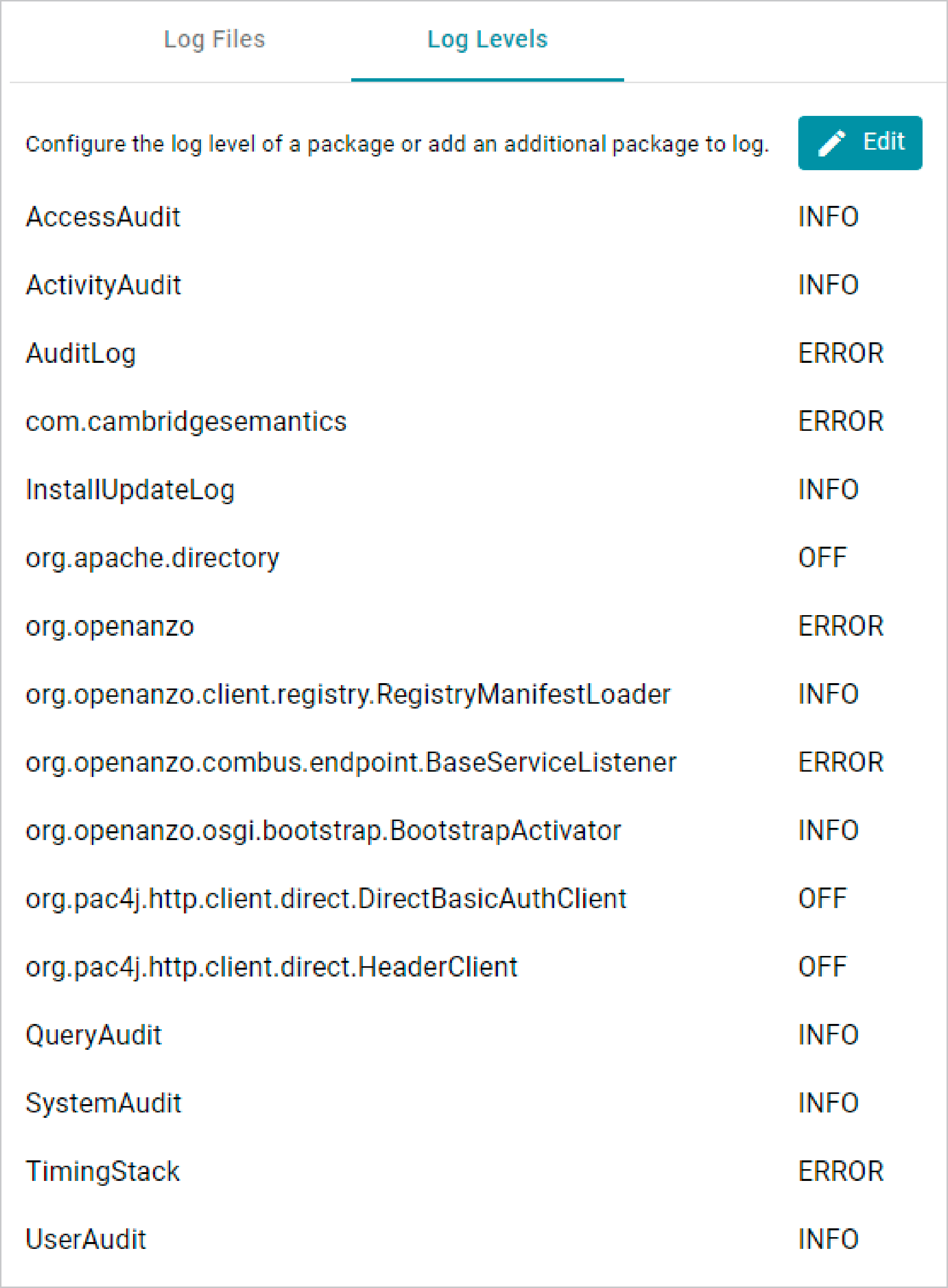
Package	Level	Description
		Changing the Log Level of this package affects logs across Anzo components and services.
org.openanzo.client.registry.RegistryManifestLoader	Info	Listener for installation and upgrade events. Captures information about bundle imports and updates.
org.openanzo.combus.endpoint.BaseServiceListener	Error	Core server listener for requests sent from clients to the server.
org.openanzo.osgi.bootstrap.BootstrapActivator	Info	Listener for installation and upgrade events. Captures information about bundle imports and updates.
org.openanzo.services.PublicLog	Off	Listener for internal Anzo events. **Do not modify the Log Level for this package**.
org.pac4j.http.client.direct.DirectBasicAuthClient	Off	Low-level listener for user login events.
org.pac4j.http.client.direct.HeaderClient	Off	Low-level listener for user login events.
QueryAudit	Info	Listener for query audit events.
SystemAudit	Info	Listener for system audit

Package	Level	Description
		events such as changes to bundle properties.
TimingStack	Error	Listener for events related to internal system journal queries.
UserAudit	Info	Listener for user administration related events, such as changes to roles.

Adding Log Packages

> Tip
> For guidance on adding the recommended Log Packages, see Adding the Recommended Log Packages.

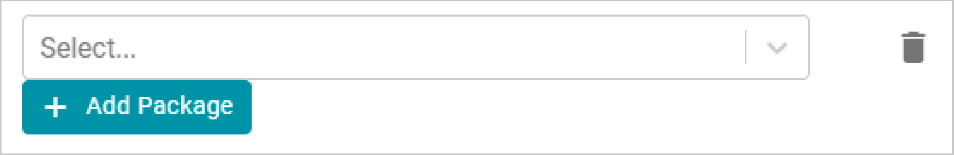
To enable additional Log Packages, click the **Edit** button on the Log Levels screen.



Configure the log level of a package or add an additional package to log.

AccessAudit	INFO
ActivityAudit	INFO
AuditLog	ERROR
com.cambridgesemantics	ERROR
InstallUpdateLog	INFO
org.apache.directory	OFF
org.openanzo	ERROR
org.openanzo.client.registry.RegistryManifestLoader	INFO
org.openanzo.combus.endpoint.BaseServiceListener	ERROR
org.openanzo.osgi.bootstrap.BootstrapActivator	INFO
org.pac4j.http.client.direct.DirectBasicAuthClient	OFF
org.pac4j.http.client.direct.HeaderClient	OFF
QueryAudit	INFO
SystemAudit	INFO
TimingStack	ERROR
UserAudit	INFO

Then click **Add Package** at the bottom of the screen.



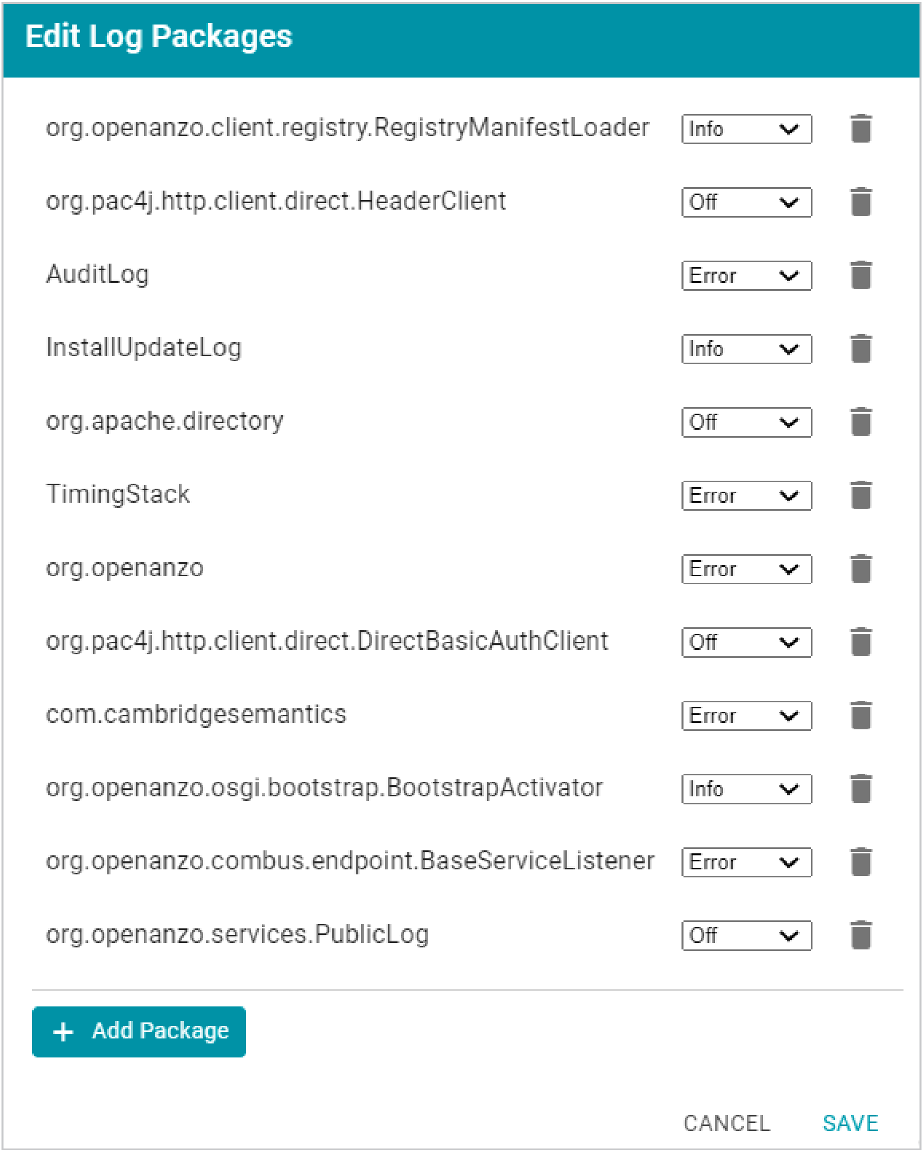
Clicking the **Select** field opens the package drop-down list. You can browse through the options, or you can start typing a keyword to search for a package. Click a package to add it to the list of packages on the Edit Log Packages screen. Adjust the Log Level as needed and then click **Save** to save the change. See Log Level Definitions below for more information about Log Levels.

Log Level Definitions

This section defines the Log Levels that are available to apply to a Log Package:

- **Off**: Turns logging off for the Log Package.
- **Debug**: Logs fine-grained error messages that are intended to help debug a problem with an application or the server.
- **Trace**: Logs finer-grained error information than Debug.
- **Info**: The highest level of logging. The Log Package captures all events or queries.
- **Warn**: Logs information about potentially problematic situations.
- **Error**: Logs errors that usually allow the application to continue running.
- **Fatal**: Logs severe errors that prevent the application from running.

To change the Log Level for a package, click the **Log Level** field for the Log Package that you want to change and select a level from the drop-down list. Click **Save** when you are finished making changes.



Edit Log Packages

org.openanzo.client.registry.RegistryManifestLoader	Info
org.pac4j.http.client.direct.HeaderClient	Off
AuditLog	Error
InstallUpdateLog	Info
org.apache.directory	Off
TimingStack	Error
org.openanzo	Error
org.pac4j.http.client.direct.DirectBasicAuthClient	Off
com.cambridgesemantics	Error
org.openanzo.osgi.bootstrap.BootstrapActivator	Info
org.openanzo.combus.endpoint.BaseServiceListener	Error
org.openanzo.services.PublicLog	Off

Viewing Log Files

All Anzo log files are generated in the `<install_path>/Server/logs` directory on the server. Files in that directory can be viewed and downloaded from the Administration application on the **Log Files** tab on the Logging screen.

- Viewing Logs on the Server
- Viewing Logs in the Administration Application

Viewing Logs on the Server

To avoid generating large log files that are difficult to manage (especially for Log Packages set to **Info**), Anzo starts logging to a new version of a file when any of the following events occur:

- A file size reaches 50 MB.
- Log settings are changed.
- Anzo is restarted.

The current, most recent version of a file is stored directly in the `<install_path>/Server/logs` directory. Earlier versions of the files are saved in `<year>_<month>_<day>_<part>` subdirectories in `Server/logs`. For example:

```
logs
├── 2021_04_27_0
│   ├── anzo_audit_info.log
│   ├── anzo_error.log
│   ├── anzo_full.log
│   ├── anzo_gqe_info.log
│   └── anzo_internal_error.log
├── 2021_04_27_1
│   ├── anzo_audit_info.log
│   ├── anzo_datasource_error.log
│   ├── anzo_error.log
│   ├── anzo_full.log
│   ├── anzo_gqe_error.log
│   ├── anzo_gqe_info.log
│   ├── anzo_install_error.log
│   └── anzo_install_info.log
├── 2021_04_28_0
│   ├── anzo_audit_info.log
│   ├── anzo_error.log
│   ├── anzo_full.log
│   ├── anzo_gqe_info.log
│   ├── anzo_install_error.log
│   └── anzo_install_info.log
├── 2021_04_28_1
│   ├── anzo_error.log
│   └── anzo_full.log
├── 2021_04_28_2
│   ├── anzo_audit_info.log
│   ├── anzo_error.log
│   └── anzo_full.log
├── anzo_audit_info.log
├── anzo_error.log
├── anzo_full.log
├── anzo_gqe_info.log
├── anzo_install_error.log
├── anzo_install_info.log
├── anzo_internal_error.log
```

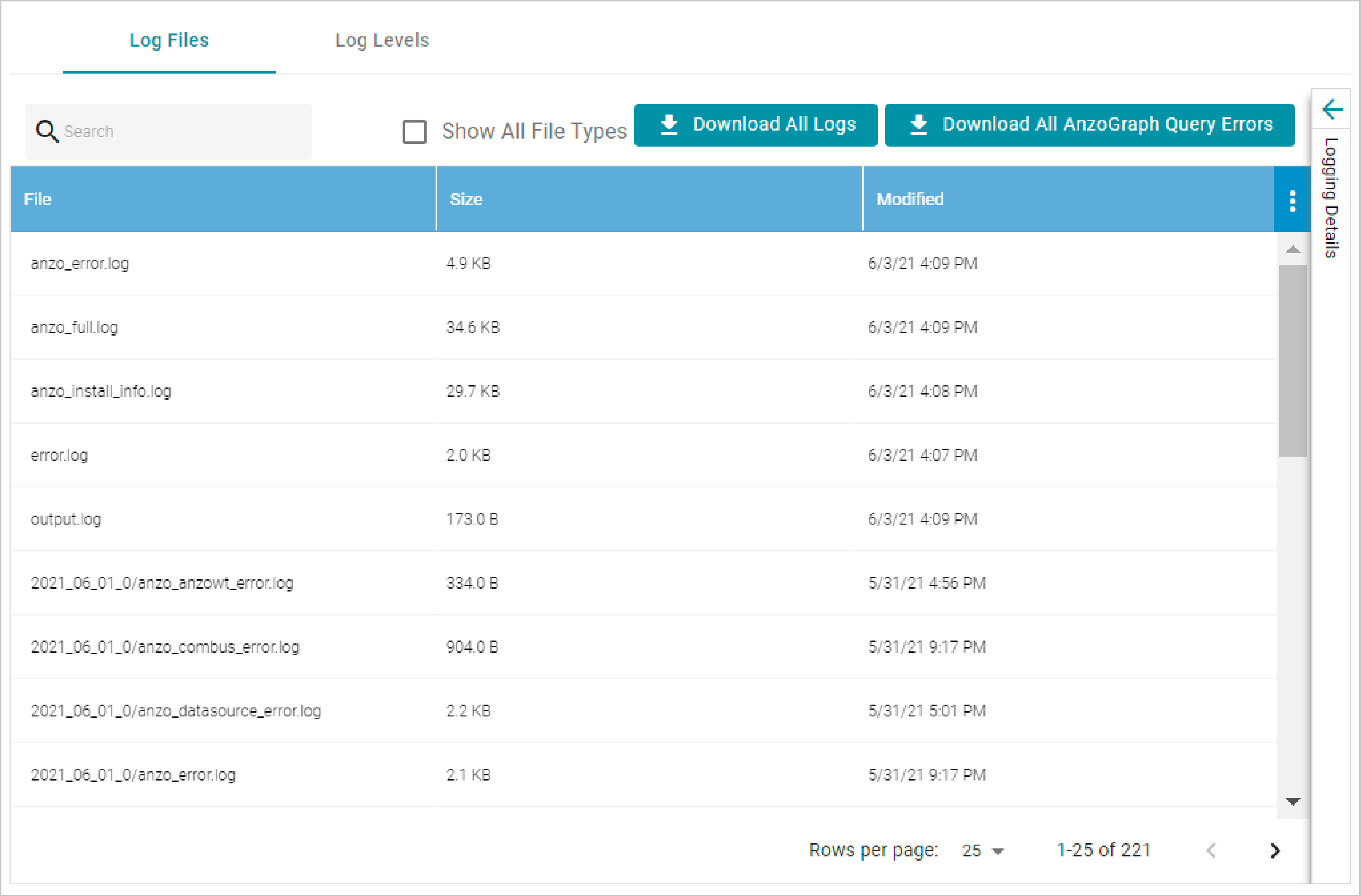
AnzoGraph query log files are stored in a directory named **gqe** in the `<install_path>/Server/logs` directory. By default all queries that are unsuccessful are captured in the **queriesError** directory. When the AnzoGraph queries Log Package is enabled, successful queries are also captured in the **queriesInfo** directory. For example:

```
logs
├── gqe
│   ├── queriesError
│   └── queriesInfo
│       ├── query_1a5548ac-6404-4321-b36b-d5eda4ca45a7_1619540406734.log
│       ├── query_1a5548ac-6404-4321-b36b-d5eda4ca45a7.log
│       ├── query_292f102e-d222-4261-a069-d7d0c8ceb823_1619469563646.log
│       ├── query_292f102e-d222-4261-a069-d7d0c8ceb823.log
│       ├── query_2ddc5f96-758d-4133-80d7-21de5f23134f_1619627154151.log
│       ├── query_2ddc5f96-758d-4133-80d7-21de5f23134f.log
│       ├── query_518ombnsruyvu8k6pf0a76y4fc-674.log
```

> **Tip**
> For instructions on enabling the AnzoGraph query Log Package, see Enabling and Viewing AnzoGraph Query Logs.

Viewing Logs in the Administration Application

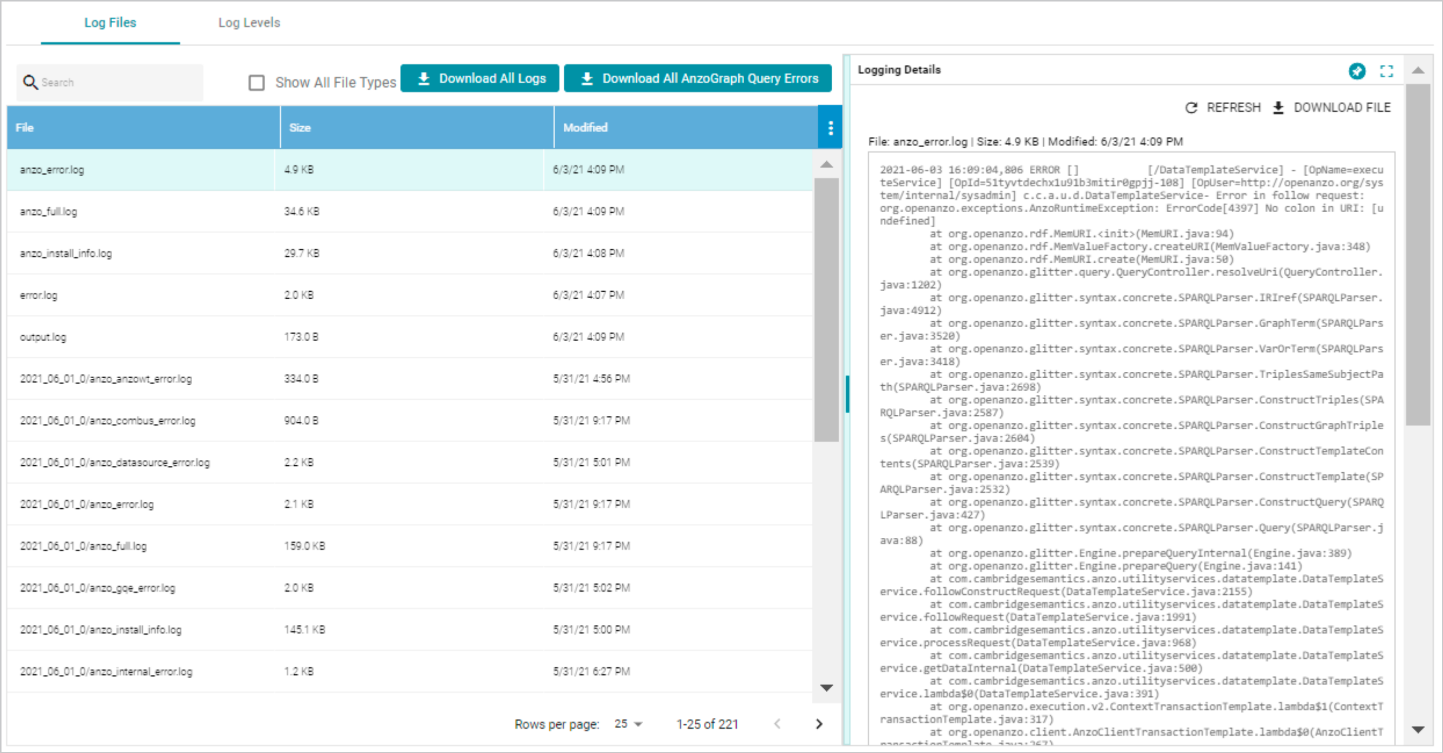
Logs in the `<install_path>/Server/logs` directory can be viewed and downloaded from the Administration application on the **Log Files** tab on the Logging screen. The Log Files tab lists the logs that are available to view. For example:



File	Size	Modified
anzo_error.log	4.9 KB	6/3/21 4:09 PM
anzo_full.log	34.6 KB	6/3/21 4:09 PM
anzo_install_info.log	29.7 KB	6/3/21 4:08 PM
error.log	2.0 KB	6/3/21 4:07 PM
output.log	173.0 B	6/3/21 4:09 PM
2021_06_01_0/anzo_anzowt_error.log	334.0 B	5/31/21 4:56 PM
2021_06_01_0/anzo_combus_error.log	904.0 B	5/31/21 9:17 PM
2021_06_01_0/anzo_datasource_error.log	2.2 KB	5/31/21 5:01 PM
2021_06_01_0/anzo_error.log	2.1 KB	5/31/21 9:17 PM

Log Packages that have the Log Level set to **Error** log events to files with the suffix **_error**. Operational information that is logged by packages that are set to **Info** is captured in files with the suffix **_info**. The current versions the log files are shown at the top of the list. Earlier versions of the logs are prefixed with the name of the `<date>_<part>` subdirectory they are saved in.

Selecting a log from the list displays its contents in the Logging Details section of the screen. For example:



File	Size	Modified
anzo_error.log	4.9 KB	6/3/21 4:09 PM
anzo_full.log	34.6 KB	6/3/21 4:09 PM
anzo_install_info.log	29.7 KB	6/3/21 4:08 PM
error.log	2.0 KB	6/3/21 4:07 PM
output.log	173.0 B	6/3/21 4:09 PM
2021_06_01_0/anzo_anzowt_error.log	334.0 B	5/31/21 4:56 PM
2021_06_01_0/anzo_combus_error.log	904.0 B	5/31/21 9:17 PM
2021_06_01_0/anzo_datasource_error.log	2.2 KB	5/31/21 5:01 PM
2021_06_01_0/anzo_error.log	2.1 KB	5/31/21 9:17 PM
2021_06_01_0/anzo_full.log	159.0 KB	5/31/21 9:17 PM
2021_06_01_0/anzo_gqe_error.log	2.0 KB	5/31/21 5:02 PM
2021_06_01_0/anzo_install_info.log	145.1 KB	5/31/21 5:00 PM
2021_06_01_0/anzo_internal_error.log	1.2 KB	5/31/21 6:27 PM

Rows per page: 25 1-25 of 221

File: anzo_error.log | Size: 4.9 KB | Modified: 6/3/21 4:09 PM

```
2021-06-03 16:09:04,806 ERROR []          [/DataTemplateService] - [OpName=execu
teService] [OpId=51tyvtdechx1u91b3mitir0gpjj-108] [OpUser=http://openanzo.org/sys
tem/internal/sysadmin] c.c.a.u.d.DataTemplateService- Error in follow request:
org.openanzo.exceptions.AnzoRuntimeException: ErrorCode[4397] No colon in URI: [u
ndefined]
        at org.openanzo.rdf.MemURI.<init>(MemURI.java:94)
        at org.openanzo.rdf.MemValueFactory.createURI(MemValueFactory.java:348)
        at org.openanzo.rdf.MemURI.create(MemURI.java:50)
        at org.openanzo.glitter.query.QueryController.resolveUri(QueryController.
java:1202)
        at org.openanzo.glitter.syntax.concrete.SPARQLParser.IRIref(SPARQLParser.
java:4912)
        at org.openanzo.glitter.syntax.concrete.SPARQLParser.GraphTerm(SPARQLPars
er.java:3520)
        at org.openanzo.glitter.syntax.concrete.SPARQLParser.VarOrTerm(SPARQLPars
er.java:3418)
        at org.openanzo.glitter.syntax.concrete.SPARQLParser.TriplesSameSubjectPa
th(SPARQLParser.java:2698)
        at org.openanzo.glitter.syntax.concrete.SPARQLParser.ConstructTriples(SPA
RQLParser.java:2587)
        at org.openanzo.glitter.syntax.concrete.SPARQLParser.ConstructGraphTriple
s(SPARQLParser.java:2604)
        at org.openanzo.glitter.syntax.concrete.SPARQLParser.ConstructTemplateCon
tents(SPARQLParser.java:2539)
        at org.openanzo.glitter.syntax.concrete.SPARQLParser.ConstructTemplate(SP
ARQLParser.java:2532)
        at org.openanzo.glitter.syntax.concrete.SPARQLParser.ConstructQuery(SPARQ
LParser.java:427)
        at org.openanzo.glitter.syntax.concrete.SPARQLParser.Query(SPARQLParser.j
ava:88)
        at org.openanzo.glitter.Engine.prepareQueryInternal(Engine.java:389)
        at org.openanzo.glitter.Engine.prepareQuery(Engine.java:141)
        at com.cambridgesemantics.anzo.utilityservices.datatemplate.DataTemplateS
ervice.followConstructRequest(DataTemplateService.java:2155)
        at com.cambridgesemantics.anzo.utilityservices.datatemplate.DataTemplateS
ervice.followRequest(DataTemplateService.java:1991)
        at com.cambridgesemantics.anzo.utilityservices.datatemplate.DataTemplateS
ervice.processRequest(DataTemplateService.java:968)
        at com.cambridgesemantics.anzo.utilityservices.datatemplate.DataTemplateS
ervice.getDataInternal(DataTemplateService.java:500)
        at com.cambridgesemantics.anzo.utilityservices.datatemplate.DataTemplateS
ervice.lambda$0(DataTemplateService.java:391)
        at org.openanzo.execution.v2.ContextTransactionTemplate.lambda$1(ContextT
ransactionTemplate.java:317)
        at org.openanzo.client.AnzoClientTransactionTemplate.lambda$0(AnzoClientT
ransactionTemplate.java:267)
```

The following options are available for viewing and downloading log files:

- To download a .zip file that contains all of the listed logs, click the **Download All Logs** button at the top of the screen.
- To download just the query error logs for AnzoGraph, click the **Download All AnzoGraph Query Errors** button at the top of the screen.
- To re-load the display with the latest version of the selected file, click the **Refresh** button at the top of the details.
- To download the file so you can view it in another editor, click **Download File** at the top of the details.

Related Topics

Adding the Recommended Log Packages

Adding the Recommended Log Packages

The Log Packages that are enabled by default cover the core Anzo server operations and services to ensure that diagnostics are generated when errors occur. Anzo includes several additional Log Packages, however, that are disabled by default but can be configured to provide valuable information for auditing purposes, such as information about user logins, user administration events, and successful AnzoGraph queries. This section describes the packages that Cambridge Semantics recommends you enable and provides information about reading the resulting log files.

Enabling and Viewing AnzoGraph Query Logs

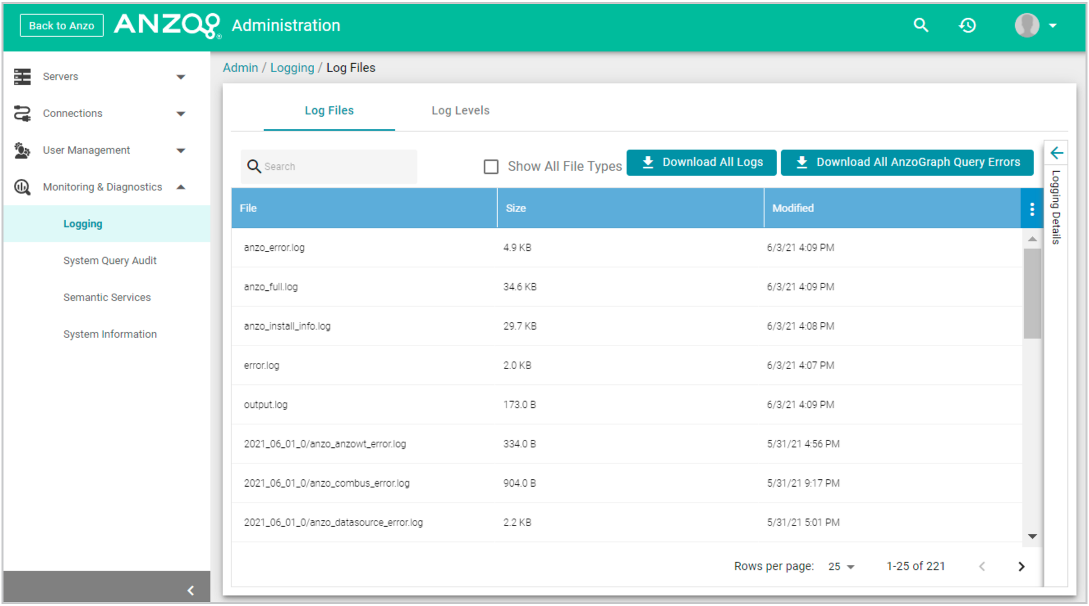
The GqeQueries Log Package listens for AnzoGraph events like connection errors, restarts, and successful and unsuccessful queries. GqeQueries is Off by default but can be enabled to monitor and log all of the queries that are sent to AnzoGraph by users through Dashboards, the Query Builder, Data Layers, etc., or sent by Anzo, such as when requesting the total number of statements in a graph.

> **Note**
> Though GqeQueries is Off by default, AnzoGraph query errors are still captured automatically in the `<install_path>/Server/logs/gqe/queriesError` directory, and connection-related errors are captured in `anzo_gqe_error.log`.

Enabling the GqeQueries Log Package

Follow the steps below to enable the GqeQueries package.

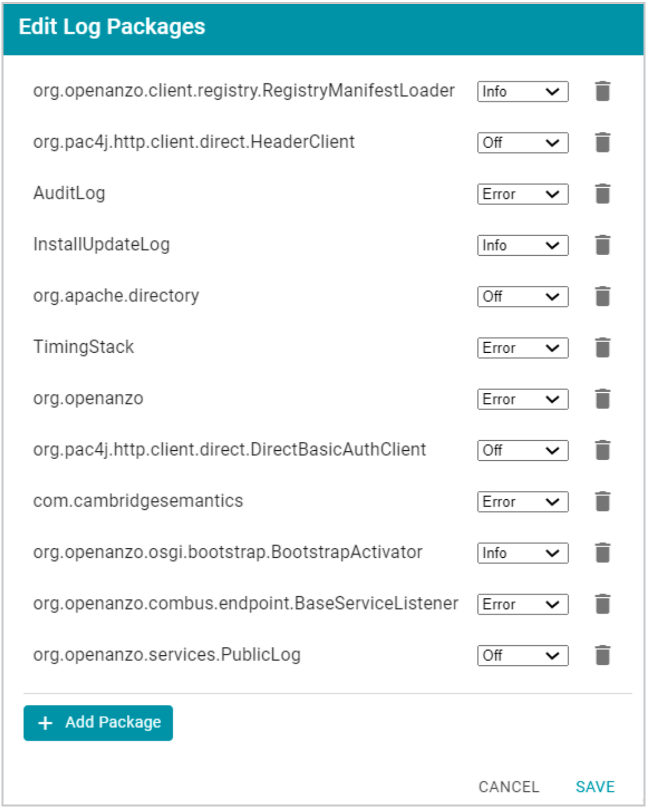
1. In the Administration application, expand the **Monitoring & Diagnostics** menu and select **Logging**. The Log Files tab is displayed on the Logging screen. For example:



File	Size	Modified
anzo_error.log	4.9 KB	6/3/21 4:09 PM
anzo_full.log	34.6 KB	6/3/21 4:09 PM
anzo_install_info.log	29.7 KB	6/3/21 4:08 PM
error.log	2.0 KB	6/3/21 4:07 PM
output.log	173.0 B	6/3/21 4:09 PM
2021_06_01_0/anzo_anzowt_error.log	334.0 B	5/31/21 4:56 PM
2021_06_01_0/anzo_combus_error.log	904.0 B	5/31/21 9:17 PM
2021_06_01_0/anzo_datasource_error.log	2.2 KB	5/31/21 5:01 PM

Rows per page: 25 1-25 of 221

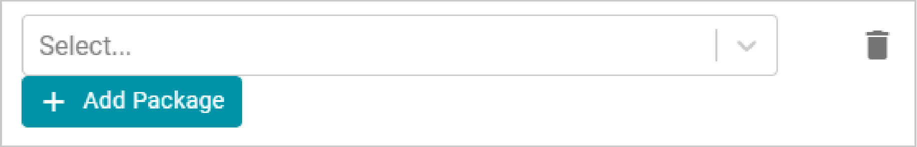
2. Click the **Edit** button to open the Edit Log Packages dialog box.



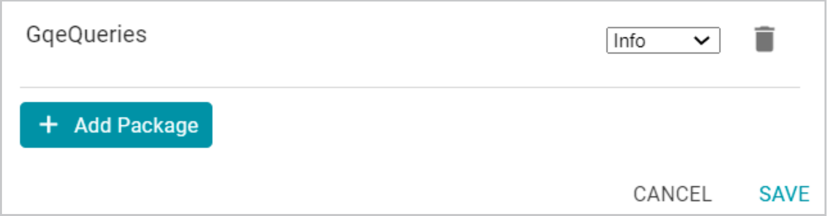
Edit Log Packages

org.openanzo.client.registry.RegistryManifestLoader	Info
org.pac4j.http.client.direct.HeaderClient	Off
AuditLog	Error
InstallUpdateLog	Info
org.apache.directory	Off
TimingStack	Error
org.openanzo	Error
org.pac4j.http.client.direct.DirectBasicAuthClient	Off
com.cambridgesemantics	Error
org.openanzo.osgi.bootstrap.BootstrapActivator	Info
org.openanzo.combus.endpoint.BaseServiceListener	Error
org.openanzo.services.PublicLog	Off

3. Click **Add Package** at the bottom of the screen. The Select field is displayed:



4. Click the **Select** field and type **GqeQueries**. Then press **Enter** to add GqeQueries to the list of Log Packages. The package is added to the list with the default Log Level of **Off**.
5. Click the Log Level drop-down list and select **Info**. Then click **Save** to save the change.

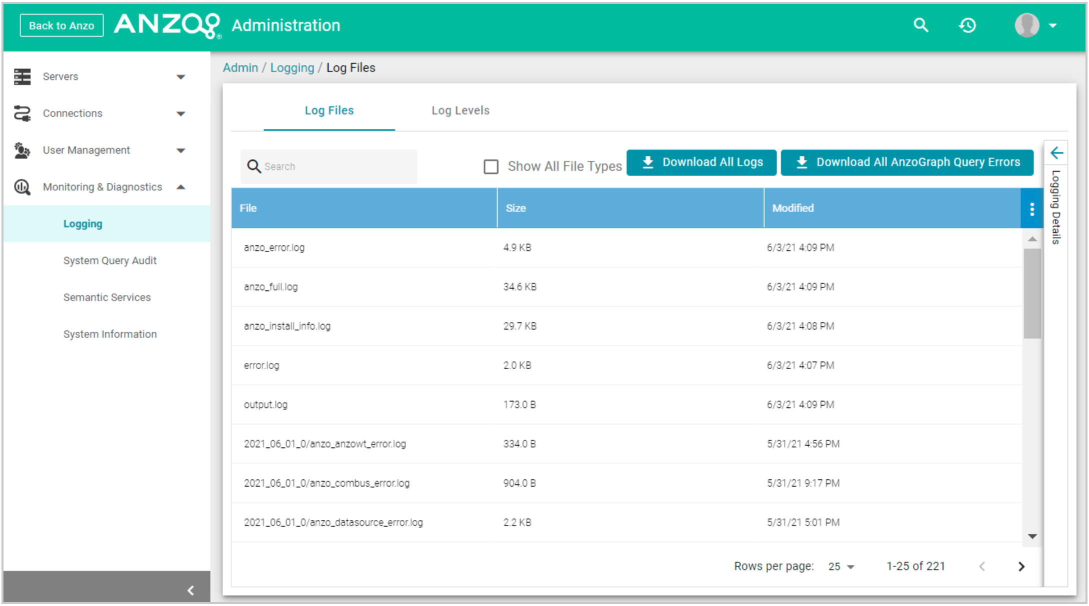

GqeQueries Info

The GqeQueries Log Package is now enabled and will start to log the events described above. The log messages for successful queries are captured in a new **anzo_gqe_info.log** file as well as in the `<install_path>/Server/logs/gqe/queriesInfo` directory on the server. Details about each request is logged to a separate file in that directory. The anzo_gqe_info.log and the files in `logs/gqe/queriesInfo` can be viewed and downloaded from the Administration application.

Viewing the AnzoGraph Query Logs

Follow the steps below to view the AnzoGraph log files in the application. For information about viewing logs on the server, see Viewing Logs on the Server.

1. In the Administration application, expand the **Monitoring & Diagnostics** menu and select **Logging**. The Log Files tab is displayed on the Logging screen. For example:

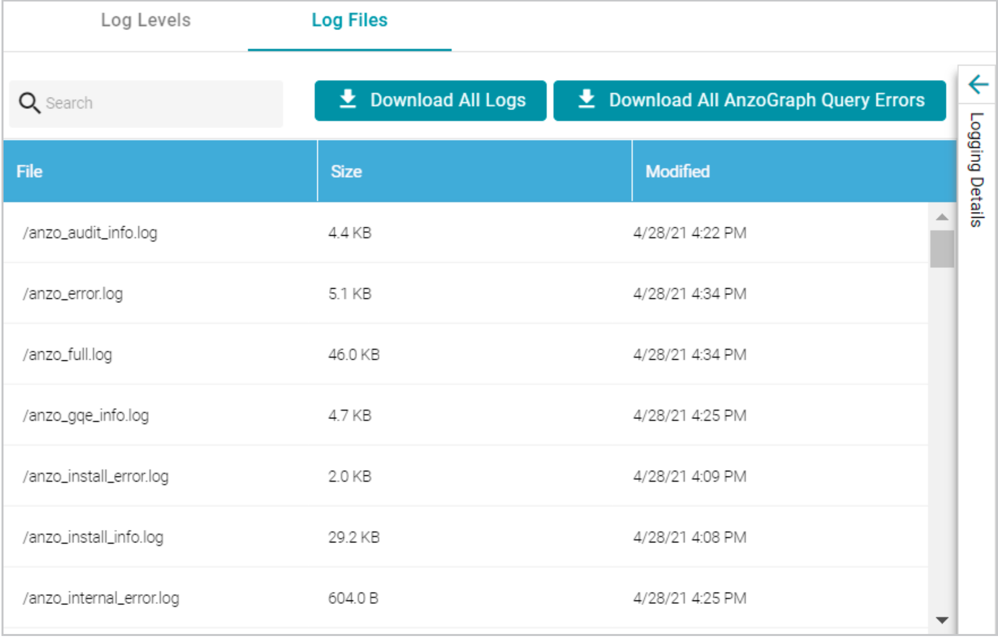


□ Show All File Types

File	Size	Modified
anzo_error.log	4.9 KB	6/3/21 4:09 PM
anzo_full.log	34.6 KB	6/3/21 4:09 PM
anzo_install_info.log	29.7 KB	6/3/21 4:08 PM
error.log	2.0 KB	6/3/21 4:07 PM
output.log	173.0 B	6/3/21 4:09 PM
2021_06_01_0/anzo_anzowt_error.log	334.0 B	5/31/21 4:56 PM
2021_06_01_0/anzo_combus_error.log	904.0 B	5/31/21 9:17 PM
2021_06_01_0/anzo_datasource_error.log	2.2 KB	5/31/21 5:01 PM

Rows per page: 25 1-25 of 221

2. Click the **Log Files** tab to view the list of files. For example:



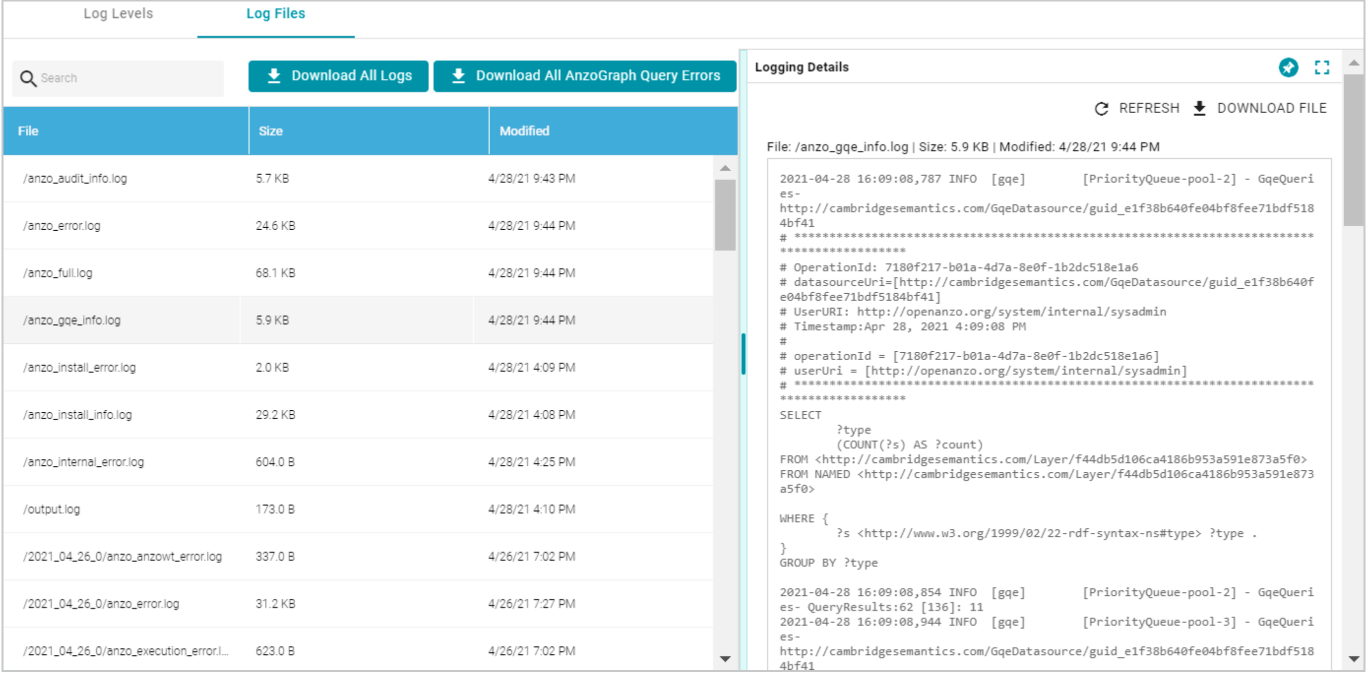
File	Size	Modified
/anzo_audit_info.log	4.4 KB	4/28/21 4:22 PM
/anzo_error.log	5.1 KB	4/28/21 4:34 PM
/anzo_full.log	46.0 KB	4/28/21 4:34 PM
/anzo_gqe_info.log	4.7 KB	4/28/21 4:25 PM
/anzo_install_error.log	2.0 KB	4/28/21 4:09 PM
/anzo_install_info.log	29.2 KB	4/28/21 4:08 PM
/anzo_internal_error.log	604.0 B	4/28/21 4:25 PM

Log Packages that have the Log Level set to **Error** log events to files with the suffix **_error**. Operational information that is logged by packages that are set to **Info** is captured in files with the suffix **_info**.

Note

The current version of **anzo_gqe_info.log** is shown toward the top of the list. Earlier versions of that log are prefixed with the name of the `<date>_<part>` subdirectory they are saved in. And individual query files are named as `/gqe/queriesInfo/<operation_ID><epoch_timestamp>`.

3. Select the **anzo_gqe_info.log** file. The contents of the file are displayed in the Logging Details section of the screen. For example:



File	Size	Modified
/anzo_audit_info.log	5.7 KB	4/28/21 9:43 PM
/anzo_error.log	24.6 KB	4/28/21 9:44 PM
/anzo_full.log	68.1 KB	4/28/21 9:44 PM
/anzo_gqe_info.log	5.9 KB	4/28/21 9:44 PM
/anzo_install_error.log	2.0 KB	4/28/21 4:09 PM
/anzo_install_info.log	29.2 KB	4/28/21 4:08 PM
/anzo_internal_error.log	604.0 B	4/28/21 4:25 PM
/output.log	173.0 B	4/28/21 4:10 PM
/2021_04_26_0/anzo_anzowt_error.log	337.0 B	4/26/21 7:02 PM
/2021_04_26_0/anzo_error.log	31.2 KB	4/26/21 7:27 PM
/2021_04_26_0/anzo_execution_error.l...	623.0 B	4/26/21 7:02 PM

File: /anzo_gqe_info.log | Size: 5.9 KB | Modified: 4/28/21 9:44 PM

```
2021-04-28 16:09:08,787 INFO  [gqe]        [PriorityQueue-pool-2] - GqeQueri
es-
http://cambridgesemantics.com/GqeDatasource/guid_e1f38b640fe04bf8fee71bdf518
4bf41
# ***************************************************************************
******************
# OperationId: 7180f217-b01a-4d7a-8e0f-1b2dc518e1a6
# datasourceUri=[http://cambridgesemantics.com/GqeDatasource/guid_e1f38b640f
e04bf8fee71bdf5184bf41]
# UserURI: http://openanzo.org/system/internal/sysadmin
# Timestamp:Apr 28, 2021 4:09:08 PM
#
# operationId = [7180f217-b01a-4d7a-8e0f-1b2dc518e1a6]
# userUri = [http://openanzo.org/system/internal/sysadmin]
# ***************************************************************************
******************
SELECT
        ?type
        (COUNT(?s) AS ?count)
FROM <http://cambridgesemantics.com/Layer/f44db5d106ca4186b953a591e873a5f0>
FROM NAMED <http://cambridgesemantics.com/Layer/f44db5d106ca4186b953a591e873
a5f0>

WHERE {
        ?s <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> ?type .
}
GROUP BY ?type

2021-04-28 16:09:08,854 INFO  [gqe]        [PriorityQueue-pool-2] - GqeQueri
es- QueryResults:62 [136]: 11
2021-04-28 16:09:08,944 INFO  [gqe]        [PriorityQueue-pool-3] - GqeQueri
es-
http://cambridgesemantics.com/GqeDatasource/guid_e1f38b640fe04bf8fee71bdf518
4bf41
```

You can expand the details view by clicking the Expand icon (⛶) in the top right corner.

The messages in **anzo_gqe_info.log** vary by the query source, such as whether the query originated in a Dashboard lens or the Query Builder. In general, GqeQueries Info messages contain the following information:

- Date and time the event was logged. For example, `2021-04-28 01:06:48`.
- The type of message, i.e., the Log Level, such as `INFO`.
- The type of log. For example, `[gqe]`.
- The area of the system or service that processed the event. For example, `[PriorityQueue-pool-2]`.
- The Log Package that was listening for the event, i.e., `GqeQueries`.

- The Data Source URI. For example, `http://cambridgesemantics.com/GqeDatasource/guid_e1f38b640fe04bf8fee71bdf5184bf41`.
- The Operation ID assigned to the query. This value can be used to track the query, such as to find the individual log file in the `logs/gqe/queriesInfo` directory. For example, `OperationId: 7b0op0wbzqeqe1s2d482xudkez-83`. The corresponding log file is named `query_7b0op0wbzqeqe1s2d482xudkez-83.log`.
- The User URI for the user who submitted the query. For example, `UserURI: ldap:///cn=Jay.Blue,ou=groups,dc=com`.
- If the query was submitted from the Hi-Res Analytics application, the message also includes details for identifying the dashboard and lens that submitted the request. For example:

  ```
  # ex_requestDashboard = [http://cambridgesemantics.com/354db630-02b6-46b2-82d0-ef4a7543ebca]
  # ex_requestSource = [http://cambridgesemantics.com/4a039bdb-bdcb-4117-830b-cb29190ce18f]
  # ex_requestSourceId = [com_cambridgesemantics_application_anzoweb_lens_grid_GridLens_7]
  ```

- The text of the query that was sent by Anzo. Note that the text is the query as rewritten by Anzo and sent to AnzoGraph. It may not be the exact text that was written by the user.
- When a query returns, a result message is also added to anzo_gqe_info.log below the query text. The QueryResults message includes the Operation ID (which matches the ID from the query that was sent), and it returns the AnzoGraph and Anzo query execution time as well as the number of results returned. In the following example, the QueryResults message is shown in bold. The first value (**2631**) is the number of milliseconds AnzoGraph spent executing the query. The value in brackets (**[13155]**) is the number of milliseconds Anzo spent executing the query. And the last value (**20**) is the number of results that were returned.

  ```
  2021-04-28 22:53:57,134 INFO  [gqe]  [PriorityQueue-pool-7] -
  [OpName=query]
  [OpId=8tt1rrc29y31z1ga30srk6t2xx-212]
  [OpUser=http://openanzo.org/system/internal/sysadmin]
  GqeQueries- QueryResults:2631 [13155]: 20
  ```

> **Note** A QueryResults message is not logged if the query uses the Anzo cache or returns an error.

A complete example message is shown below:

```
2021-04-27 19:54:25,648 INFO  [gqe]  [PriorityQueue-pool-2] - GqeQueries-
http://cambridgesemantics.com/GqeDatasource/guid_
e1f38b640fe04bf8fee71bdf5184bf41
#
*******************************************************************************
***
# OperationId: 66ed1f10-5aae-45b0-861c-3a851022d294
# datasourceUri=[http://cambridgesemantics.com/GqeDatasource/guid_
e1f38b640fe04bf8fee71bdf5184bf41]
# UserURI: http://openanzo.org/system/internal/sysadmin
# Timestamp:Apr 27, 2021 7:54:25 PM
#
# operationId = [66ed1f10-5aae-45b0-861c-3a851022d294]
# userUri = [http://openanzo.org/system/internal/sysadmin]
#
*******************************************************************************
****
SELECT
    ?type
    (COUNT(?s) AS ?count)
FROM <http://cambridgesemantics.com/Layer/f44db5d106ca4186b953a591e873a5f0>
FROM NAMED
<http://cambridgesemantics.com/Layer/f44db5d106ca4186b953a591e873a5f0>
WHERE {
    ?s <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> ?type .
}
GROUP BY ?type
2021-04-27 19:54:25,670 INFO  [gqe] [PriorityQueue-pool-2] - GqeQueries-
QueryResults:16 [100]: 11
```

Related Topics

Introduction to Anzo Logging

Enabling and Viewing Audit Logs

Retrieving AnzoGraph Diagnostic Files

Enabling and Viewing Audit Logs

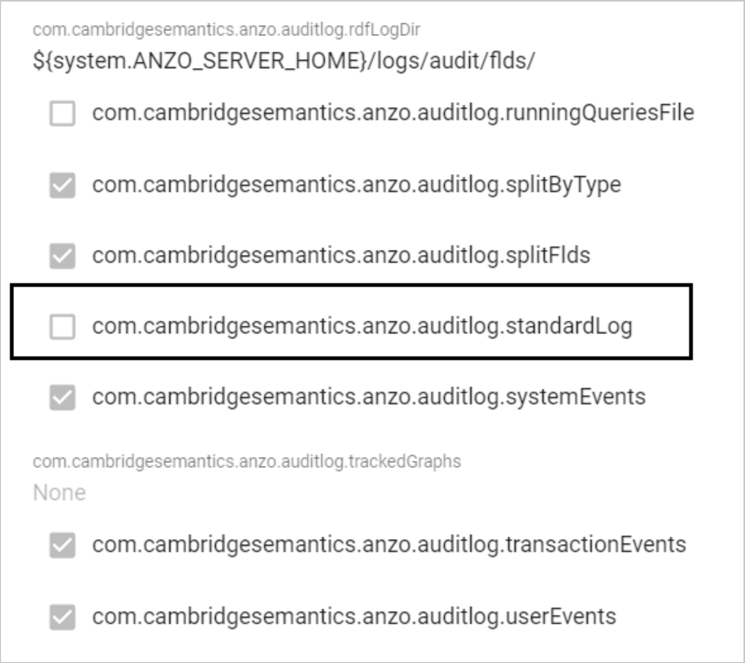
The Audit Log Packages listen for user- or security-related events such as access attempts and user administration-related events such as modifications to users, groups, and roles. The Audit Log packages are disabled by default but can be enabled to monitor and log the following types of events:

- The inactivity timeout is changed.
- A bundle's properties are changed or a bundle is restarted.
- A user successfully logs in or out or there are failed login attempts.
- A user account is created or deleted or a user's password is changed.
- A user or group is synchronized with the directory server.
- A role is created or deleted.
- A user is added to or removed from a role or group.
- A permission is added to or removed from a role.
- Data access permissions are changed on artifacts.

Enabling the Audit Log Packages

By default, the Audit Log packages (UserAudit, AccessAudit, QueryAudit, ActivityAudit, and SystemAudit) are set to the Log Level **Info**, which means they are configured to capture all audit events. However, logging the audit events are disabled by default in the Anzo Audit Logging Framework service. Follow the instructions below to configure the service to enable audit logging:

1. In the Administration application, expand the **Servers** menu and click **Advanced Configuration**. Click **I understand and accept the risk**.
2. Search for the **Anzo Audit Logging Framework** bundle and view its details.
3. Click the **Services** tab and expand **com.cambridgesemantics.anzo.AuditLog**.
4. Find the **com.cambridgesemantics.anzo.auditlog.standardLog** property towards the bottom of the list (shown below).



com.cambridgesemantics.anzo.auditlog.rdfLogDir
${system.ANZO_SERVER_HOME}/logs/audit/flds/

☐ com.cambridgesemantics.anzo.auditlog.runningQueriesFile

☑ com.cambridgesemantics.anzo.auditlog.splitByType

☑ com.cambridgesemantics.anzo.auditlog.splitFlds

☐ com.cambridgesemantics.anzo.auditlog.standardLog

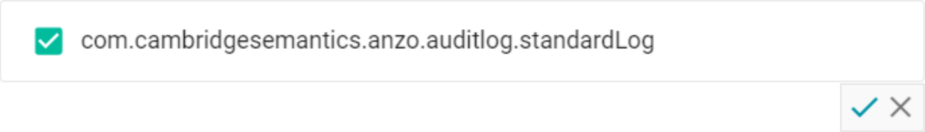
☑ com.cambridgesemantics.anzo.auditlog.systemEvents

com.cambridgesemantics.anzo.auditlog.trackedGraphs
None

☑ com.cambridgesemantics.anzo.auditlog.transactionEvents

☑ com.cambridgesemantics.anzo.auditlog.userEvents

5. Click the property to make it editable, and then select the checkbox to enable it.



☑ com.cambridgesemantics.anzo.auditlog.standardLog

6. Click the checkmark icon (✓) to save the change.
7. Restart Anzo to apply the configuration change.

The Audit Log Packages are now enabled and will start to log the events described above. The log messages are captured in **anzo_full.log** as well as a new file called **anzo_audit_info.log**. All Anzo log files are generated in the `<install_path>/Server/logs` directory on the server. Files in that directory can be viewed and downloaded from the Administration application.

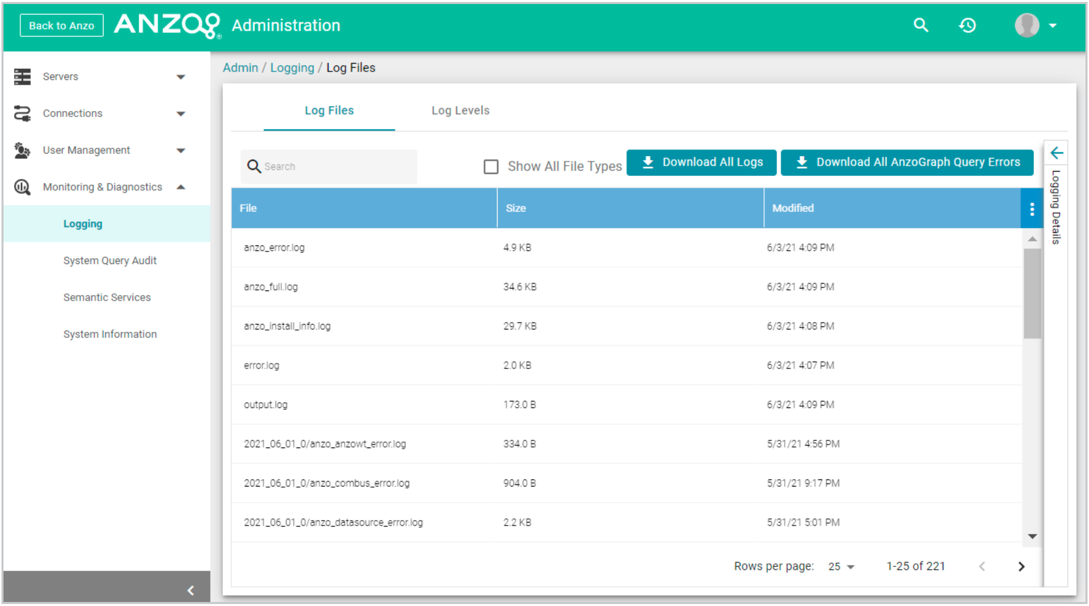
Viewing the Audit Log

Follow the steps below to view the Audit log file in the application. For information about viewing logs on the server, see Viewing Logs on the Server.

> **Tip**
> You have the option to split the Audit log into separate files based on the type of event that is being logged, such a user event or a query event. See Separating Audit Logs by Type of Event for information. The steps below refer to the default Audit Log where all types of audit events are

> recorded in a single file.

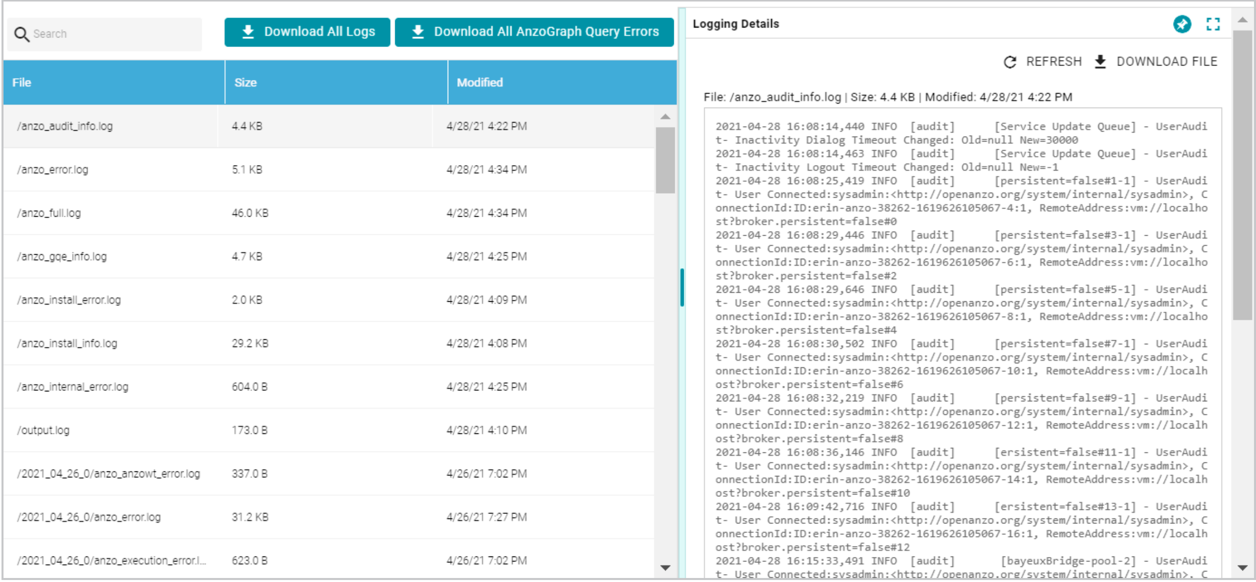
1. In the Administration application, expand the **Monitoring & Diagnostics** menu and select **Logging**. The Log Files tab is displayed on the Logging screen. For example:

File	Size	Modified
anzo_error.log	4.9 KB	6/3/21 4:09 PM
anzo_full.log	34.6 KB	6/3/21 4:09 PM
anzo_install_info.log	29.7 KB	6/3/21 4:08 PM
error.log	2.0 KB	6/3/21 4:07 PM
output.log	173.0 B	6/3/21 4:09 PM
2021_06_01_0/anzo_anzowt_error.log	334.0 B	5/31/21 4:56 PM
2021_06_01_0/anzo_combus_error.log	904.0 B	5/31/21 9:17 PM
2021_06_01_0/anzo_datasource_error.log	2.2 KB	5/31/21 5:01 PM

   Log Packages that have the Log Level set to **Error** log events to files with the suffix **_error**. Operational information that is logged by packages that are set to **Info** is captured in files with the suffix **_info**. The current versions of the log files are shown at the top of the list. Earlier versions of the logs are prefixed with the name of the `<date>_<part>` subdirectory they are saved in.

2. Select the **anzo_audit_info.log** file. The contents of the file are displayed in the Logging Details section of the screen. For example:



File	Size	Modified
/anzo_audit_info.log	4.4 KB	4/28/21 4:22 PM
/anzo_error.log	5.1 KB	4/28/21 4:34 PM
/anzo_full.log	46.0 KB	4/28/21 4:34 PM
/anzo_gqe_info.log	4.7 KB	4/28/21 4:25 PM
/anzo_install_error.log	2.0 KB	4/28/21 4:09 PM
/anzo_install_info.log	29.2 KB	4/28/21 4:08 PM
/anzo_internal_error.log	604.0 B	4/28/21 4:25 PM
/output.log	173.0 B	4/28/21 4:10 PM
/2021_04_26_0/anzo_anzowt_error.log	337.0 B	4/26/21 7:02 PM
/2021_04_26_0/anzo_error.log	31.2 KB	4/26/21 7:27 PM
/2021_04_26_0/anzo_execution_error.l...	623.0 B	4/26/21 7:02 PM

File: /anzo_audit_info.log | Size: 4.4 KB | Modified: 4/28/21 4:22 PM

```
2021-04-28 16:08:14,440 INFO  [audit]      [Service Update Queue] - UserAudi
t- Inactivity Dialog Timeout Changed: Old=null New=30000
2021-04-28 16:08:14,463 INFO  [audit]      [Service Update Queue] - UserAudi
t- Inactivity Logout Timeout Changed: Old=null New=-1
2021-04-28 16:08:25,419 INFO  [audit]      [persistent=false#1-1] - UserAudi
t- User Connected:sysadmin:<http://openanzo.org/system/internal/sysadmin>, C
onnectionId:ID:erin-anzo-38262-1619626105067-4:1, RemoteAddress:vm://localho
st?broker.persistent=false#0
2021-04-28 16:08:29,446 INFO  [audit]      [persistent=false#3-1] - UserAudi
t- User Connected:sysadmin:<http://openanzo.org/system/internal/sysadmin>, C
onnectionId:ID:erin-anzo-38262-1619626105067-6:1, RemoteAddress:vm://localho
st?broker.persistent=false#2
2021-04-28 16:08:29,646 INFO  [audit]      [persistent=false#5-1] - UserAudi
t- User Connected:sysadmin:<http://openanzo.org/system/internal/sysadmin>, C
onnectionId:ID:erin-anzo-38262-1619626105067-8:1, RemoteAddress:vm://localho
st?broker.persistent=false#4
2021-04-28 16:08:30,502 INFO  [audit]      [persistent=false#7-1] - UserAudi
t- User Connected:sysadmin:<http://openanzo.org/system/internal/sysadmin>, C
onnectionId:ID:erin-anzo-38262-1619626105067-10:1, RemoteAddress:vm://localh
ost?broker.persistent=false#6
2021-04-28 16:08:32,219 INFO  [audit]      [persistent=false#9-1] - UserAudi
t- User Connected:sysadmin:<http://openanzo.org/system/internal/sysadmin>, C
onnectionId:ID:erin-anzo-38262-1619626105067-12:1, RemoteAddress:vm://localh
ost?broker.persistent=false#8
2021-04-28 16:08:36,146 INFO  [audit]      [ersistent=false#11-1] - UserAudi
t- User Connected:sysadmin:<http://openanzo.org/system/internal/sysadmin>, C
onnectionId:ID:erin-anzo-38262-1619626105067-14:1, RemoteAddress:vm://localh
ost?broker.persistent=false#10
2021-04-28 16:09:42,716 INFO  [audit]      [ersistent=false#13-1] - UserAudi
t- User Connected:sysadmin:<http://openanzo.org/system/internal/sysadmin>, C
onnectionId:ID:erin-anzo-38262-1619626105067-16:1, RemoteAddress:vm://localh
ost?broker.persistent=false#12
2021-04-28 16:15:33,491 INFO  [audit]       [bayeuxBridge-pool-2] - UserAudi
t- User Connected:sysadmin:<http://openanzo.org/system/internal/sysadmin>, C
```

You can expand the details view by clicking the Expand icon (⛶) in the top right corner.

The elements included in each message vary by message type. In general, UserAudit Info messages contain the following information:

- Date and time the event was logged. For example, `2021-04-28 01:06:48`.
- The type of message, i.e., the Log Level, such as `INFO`.
- The type of log. For example, `[audit]`.
- The area of the system or service that processed the event. For example, `[UniformSaveService]`.
- The Log Package that was listening for the event, i.e., `UserAudit`.
- The message text, such as `User Connected` or `Authentication Failed`.
- The unique Operation ID assigned for the operation. For example, `[OpId=518ombnsruyvu8k6pf0a76y4fc-1414]`.
- The name of the service that performed the operation. For example, `[OpName=executeService]`.
- The user who performed the operation. For example, `[OpUser=http://openanzo.org/system/internal/sysadmin]`.

Below are examples of the types of messages that are logged (line breaks added for readability):

Successful User Login

```
2021-04-27 16:12:28,754 INFO [audit] [persistent=false#1-1] - UserAudit-
User Connected:sysadmin:<http://openanzo.org/system/internal/sysadmin>,
ConnectionId:ID:anzo-36673-1619539948446-4:1,
RemoteAddress:vm://localhost?broker.persistent=false#0
```

Failed User Login

```
2021-04-28 01:06:48,341 INFO  [audit] [erverThreadPool-3323] -
[OpName=ServerRealm.Authenticate]
[OpId=a876f781-5ddf-424d-8d54-c2ea07c87561]
UserAudit-
Authentication Failed:test,
Message:ErrorCode[3844] User test not found.
```

Inactivity Timeout Value Changed

```
2021-04-27 19:50:17,316 INFO  [audit] [Service Update Queue] -
[OpName=executeService]
[OpId=518ombnsruyvu8k6pf0a76y4fc-1802]
[OpUser=http://openanzo.org/system/internal/sysadmin]
UserAudit- Inactivity Logout Timeout Changed: Old=-1 New=900000
```

New Role Created

```
2021-04-27 18:58:38,276 INFO  [audit] [r/UniformSaveService] -
[OpName=executeService]
[OpId=518ombnsruyvu8k6pf0a76y4fc-1414]
[OpUser=http://openanzo.org/system/internal/sysadmin]
UserAudit-
Role Created:
<http://cambridgesemantics.com/Role/952810ffb74a42f8b502adc422608e64>
```

Permission Added to a Role

```
2021-04-28 20:41:10,926 INFO  [audit] [r/UniformSaveService] -
[OpName=executeService]
[OpId=5q6p7zmp9xn2xujksz4l7pzzl-1808]
[OpUser=http://openanzo.org/system/internal/sysadmin]
UserAudit-
Permission <http://cambridgesemantics.com/permissions/feature/e5c11e5b-afb2-
4af0-b1d7-0e4b620a0378>
added to Role
<http://cambridgesemantics.com/Role/952810ffb74a42f8b502adc422608e64>
```

Related Topics

Introduction to Anzo Logging

Limiting the Age (and Size) of Audit Logs

Separating Audit Logs by Type of Event

Configuring a User Inactivity Timeout

Enabling and Viewing AnzoGraph Query Logs

Retrieving AnzoGraph Diagnostic Files

When Cambridge Semantics Support requests AnzoGraph diagnostic files for troubleshooting an issue, you can quickly retrieve the files from the Diagnostics tab on the AnzoGraph page in the Anzo Administration application. This topic provides information about the AnzoGraph diagnostics and instructions for retrieving the files.

Introduction to AnzoGraph Diagnostic Files

There are two types of AnzoGraph diagnostic files:

- **XRay**: XRays are generated on-demand. If you encounter an error and the database remains running, you generate an XRay to produce the diagnostic files.
- **Crash Dump**: If you encounter an error that crashes the database, AnzoGraph automatically generates a crash dump that contains diagnostic information about the crash.

Xrays and crash dumps are valuable tools that enable Cambridge Semantics to diagnose and fix issues without access or any other visibility into a customer’s data or database system. They can also be used to report on overall and detailed system performance, resulting in improved query performance for future releases of AnzoGraph.

Xrays and crash dumps harvest the diagnostic data that is stored in AnzoGraph's system tables. They include information such as:

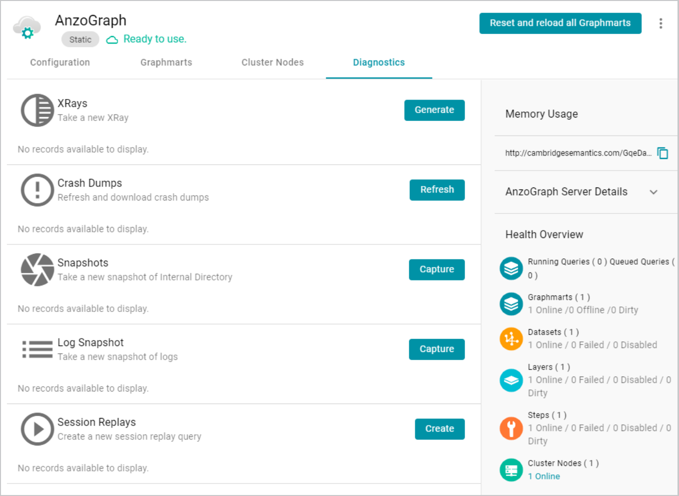
- A low level, de-identified log of the requests that were sent to the database.
- Statistics like query operation step execution times, number of rows processed, and amount of memory used.
- Detailed but de-identified trace information for errors that were encountered.
- Configuration information such as the number of nodes in the cluster and AnzoGraph system settings values.

Xrays and crash dumps are designed to be anonymous and can be safely shared with Cambridge Semantics Support. They do NOT capture user information or any of the data that is loaded into memory by a user, nor do they expose details that could be used to reveal the nature of the data being queried.

Retrieving the Files

Follow the instructions below to download an xray or crash dump to send to Cambridge Semantics Support.

1. In the Administration application, expand the **Connections** menu and select **AnzoGraph**. Anzo displays the AnzoGraph screen, which lists the connected AnzoGraph instances.
2. Click the name of the AnzoGraph instance for which you want to download an xray or crash dump. Anzo displays the Graphmarts screen for the instance.
3. Click the **Diagnostics** tab. Anzo displays the available options. For example:



No records available to display.

No records available to display.

No records available to display.

No records available to display.

No records available to display.

http://cambridgesemantics.com/GqeDa...

Running Queries (0) Queued Queries (0)

Graphmarts (1)
1 Online /0 Offline /0 Dirty

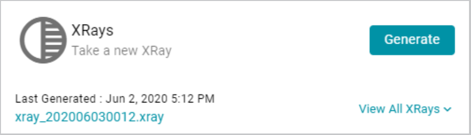
Datasets (1)
1 Online / 0 Failed / 0 Disabled

Layers (1)
1 Online / 0 Failed / 0 Disabled / 0 Dirty

Steps (1)
1 Online / 0 Failed / 0 Disabled / 0 Dirty

Cluster Nodes (1)
1 Online

4. If you want to retrieve an xray, click the **Generate** button for Xrays. Anzo creates the xray and produces a tarball with a .xray extension. For example:



Last Generated : Jun 2, 2020 5:12 PM
xray_202006030012.xray

Click the xray file name to download the tarball to you computer for sending to Cambridge Semantics.

> **Note**
>
> The files in the tarball are compressed. Do not compress the .xray file before sending it to Cambridge Semantics.

5. If you want to retrieve a crash dump, click the **Refresh** button next to Crash Dumps to refresh the list of available crash dump files. Click the file name that you want to download. Anzo downloads the file to your computer.

Related Topics

Monitoring AnzoGraph

AnzoGraph Server Administration

Monitoring AnzoGraph

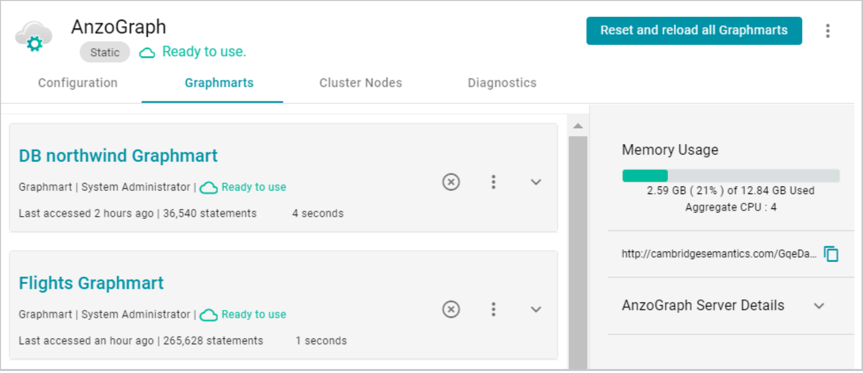
This topic provides information about viewing AnzoGraph's memory usage, query performance statistics, and network bandwidth.

- Viewing Current Memory Usage
- Reviewing Query Performance Statistics
- Evaluating Network Performance on Clusters

Viewing Current Memory Usage

Follow the steps below to view AnzoGraph's current memory usage.

1. In the Administration application, expand the **Connections** menu and select **AnzoGraph**. Anzo displays the AnzoGraph screen, which lists the connected AnzoGraph instances.
2. Click the name of the instance that you want to evaluate. Anzo displays the Graphmarts screen for that instance. The memory usage details are displayed in the top right corner on all of the tabs. For example, the test instance below shows that 21% of the available memory is in use:



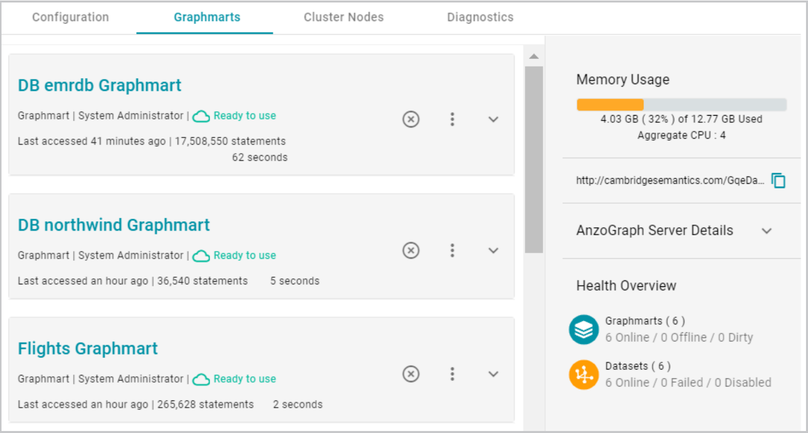
DB northwind Graphmart
Graphmart | System Administrator | Ready to use
Last accessed 2 hours ago | 36,540 statements 4 seconds

Flights Graphmart
Graphmart | System Administrator | Ready to use
Last accessed an hour ago | 265,628 statements 1 seconds

2.59 GB (21%) of 12.84 GB Used
Aggregate CPU : 4

http://cambridgesemantics.com/GqeDa...

Ideally, the data at rest should use only 25%-30% of the available memory because query execution and intermediate result storage can temporarily consume a very large amount of RAM, especially when multiple users run queries concurrently. When memory usage increases so that the data uses more than 25% - 30% of the available memory, the status bar changes color to orange as a warning . For example:



DB emrdb Graphmart

Graphmart | System Administrator | Ready to use

Last accessed 41 minutes ago | 17,508,550 statements 62 seconds

DB northwind Graphmart

Graphmart | System Administrator | Ready to use

Last accessed an hour ago | 36,540 statements 5 seconds

Flights Graphmart

Graphmart | System Administrator | Ready to use

Last accessed an hour ago | 265,628 statements 2 seconds

Memory Usage

4.03 GB (32%) of 12.77 GB Used

Aggregate CPU : 4

http://cambridgesemantics.com/GqeDa...

AnzoGraph Server Details

Health Overview

Graphmarts (6)
6 Online / 0 Offline / 0 Dirty

Datasets (6)
6 Online / 0 Failed / 0 Disabled

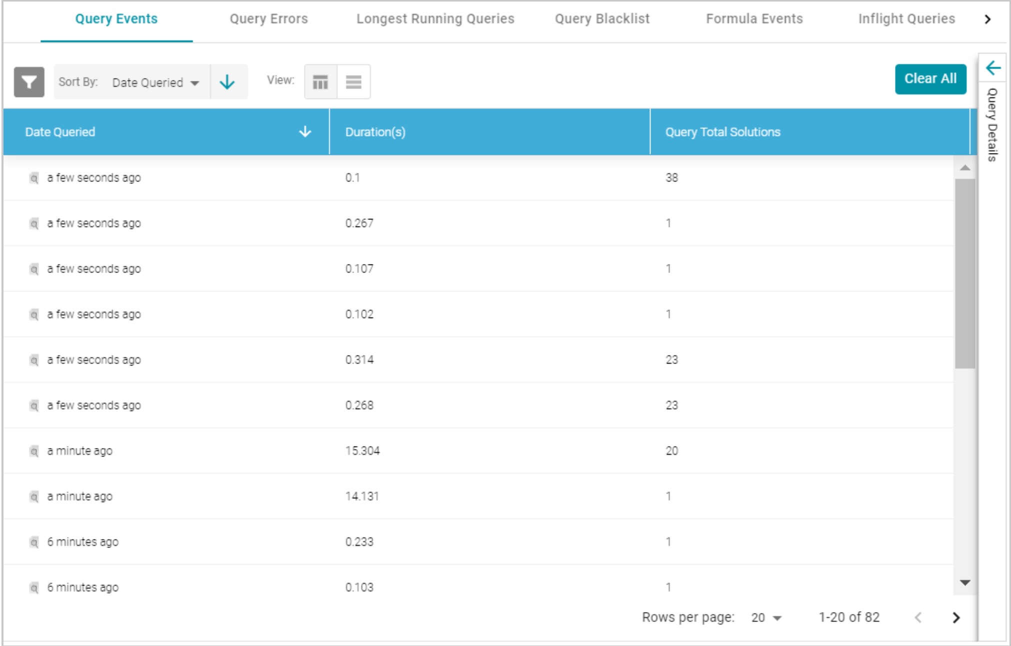
If memory usage for the data at rest remains above 50%, Cambridge Semantics recommends that you increase the amount of RAM available. For more information about memory usage, see Sizing Guidelines for In-Memory Storage in the Deployment Guide.

Reviewing Query Performance Statistics

The System Query Audit log provides details about all system events. Users can filter the query audit log to view query execution times for AnzoGraph queries.

Viewing AnzoGraph Query Statistics

1. In the Administration application, expand the **Monitoring & Diagnostics** menu and select **System Query Audit**. Anzo displays the Query Events log. For example:

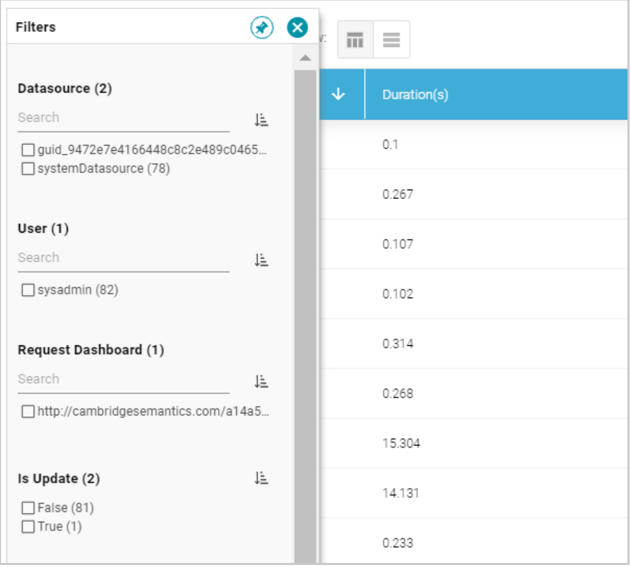


Date Queried	Duration(s)	Query Total Solutions
a few seconds ago	0.1	38
a few seconds ago	0.267	1
a few seconds ago	0.107	1
a few seconds ago	0.102	1
a few seconds ago	0.314	23
a few seconds ago	0.268	23
a minute ago	15.304	20
a minute ago	14.131	1
6 minutes ago	0.233	1
6 minutes ago	0.103	1

Rows per page: 20 1-20 of 82

By default, the log shows an overview of all query events for all data sources. The table lists the date queried, the duration in milliseconds, and total number of solutions returned for each query event. You can select an event in the table to view details about that event, such as the target data source and query text, on the right side of the screen.

2. To filter the events to display only AnzoGraph queries, open the Filters panel by clicking the filter icon (▼) in the top left corner of the screen. For example:



Filters

Datasource (2)

Search

□ guid_9472e7e4166448c8c2e489c0465...
□ systemDatasource (78)

User (1)

Search

□ sysadmin (82)

Request Dashboard (1)

Search

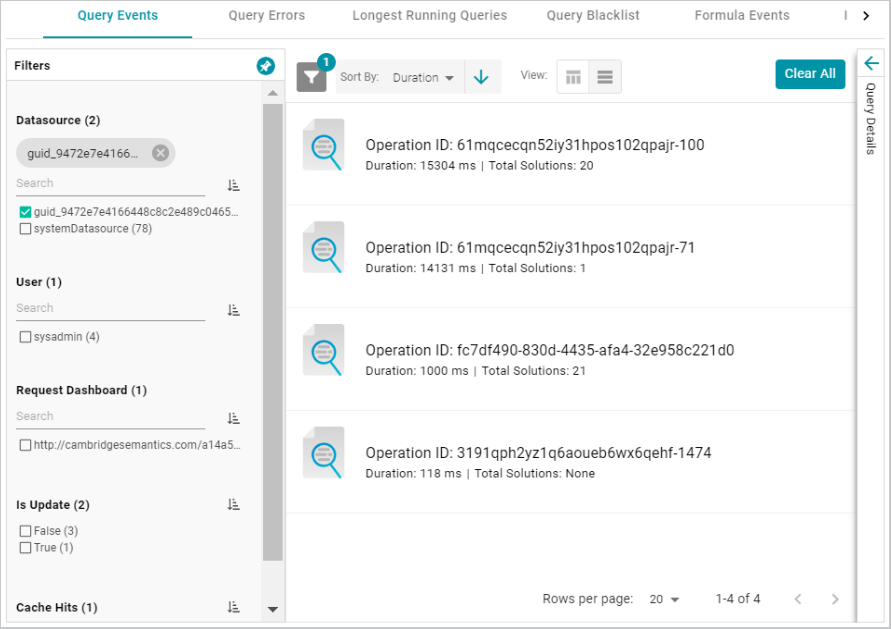
□ http://cambridgesemantics.com/a14a5...

Is Update (2)

□ False (81)
□ True (1)

Duration(s)
0.1
0.267
0.107
0.102
0.314
0.268
15.304
14.131
0.233

3. In the Filters panel under **Datasource**, select the checkbox for the AnzoGraph data source. Typically the name starts with **guid_**. The table of events is filtered to display AnzoGraph events. At the top of the screen, you can choose between a table view (▦) or list view (≡), and you can sort by date, duration, or total solutions. For example, the image below shows a list view of AnzoGraph query events sorted by duration:



Datasource (2)

guid_9472e7e4166...

☑ guid_9472e7e4166448c8c2e489c0465...
□ systemDatasource (78)

User (1)

□ sysadmin (4)

Request Dashboard (1)

□ http://cambridgesemantics.com/a14a5...

Is Update (2)

□ False (3)
□ True (1)

Cache Hits (1)

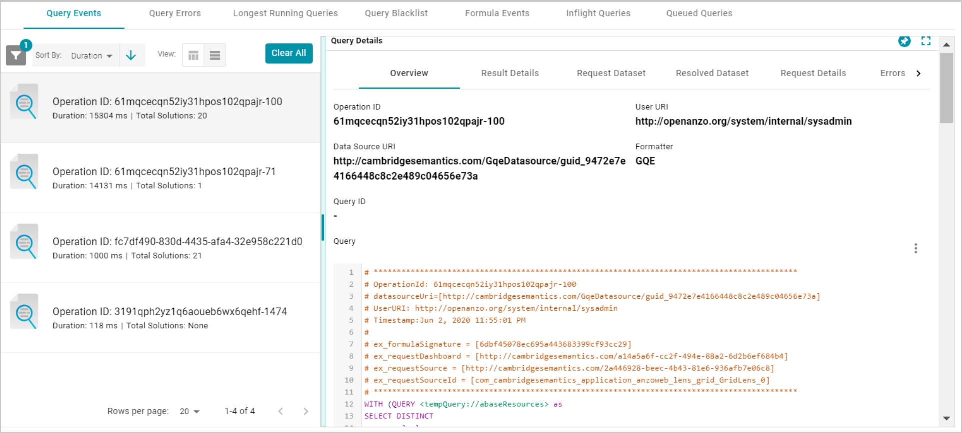
Operation ID: 61mqcecqn52iy31hpos102qpajr-100
Duration: 15304 ms | Total Solutions: 20

Operation ID: 61mqcecqn52iy31hpos102qpajr-71
Duration: 14131 ms | Total Solutions: 1

Operation ID: fc7df490-830d-4435-afa4-32e958c221d0
Duration: 1000 ms | Total Solutions: 21

Operation ID: 3191qph2yz1q6aoueb6wx6qehf-1474
Duration: 118 ms | Total Solutions: None

4. Select any query in the list to view the event overview on the right side of the screen. For example:



Operation ID: 61mqcecqn52iy31hpos102qpajr-100
Duration: 15304 ms | Total Solutions: 20

Operation ID: 61mqcecqn52iy31hpos102qpajr-71
Duration: 14131 ms | Total Solutions: 1

Operation ID: fc7df490-830d-4435-afa4-32e958c221d0
Duration: 1000 ms | Total Solutions: 21

Operation ID: 3191qph2yz1q6aoueb6wx6qehf-1474
Duration: 118 ms | Total Solutions: None

Operation ID
61mqcecqn52iy31hpos102qpajr-100

User URI
http://openanzo.org/system/internal/sysadmin

Data Source URI
http://cambridgesemantics.com/GqeDatasource/guid_9472e7e4166448c8c2e489c04656e73a

Formatter
GQE

Query ID
-

Query

```
# ***********************************************************************************************
# OperationId: 61mqcecqn52iy31hpos102qpajr-100
# datasourceUri=[http://cambridgesemantics.com/GqeDatasource/guid_9472e7e4166448c8c2e489c04656e73a]
# UserURI: http://openanzo.org/system/internal/sysadmin
# Timestamp:Jun 2, 2020 11:55:01 PM
#
# ex_formulaSignature = [6dbf45078ec695a443683399cf93cc29]
# ex_requestDashboard = [http://cambridgesemantics.com/a14a5a6f-cc2f-494e-88a2-6d2b6ef684b4]
# ex_requestSource = [http://cambridgesemantics.com/2a446928-beec-4b43-81e6-936afb7e06c8]
# ex_requestSourceId = [com_cambridgesemantics_application_anzoweb_lens_grid_GridLens_0]
# ***********************************************************************************************
WITH (QUERY <tempQuery://abaseResources> as
SELECT DISTINCT
```

To view more details about the query event, click the additional tabs to the right of the Overview tab.

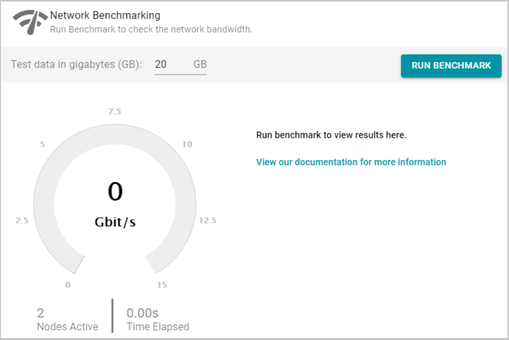
Evaluating Network Performance on Clusters

The AnzoGraph Diagnostics screen provides a network benchmark that you can run to evaluate the network bandwidth of a cluster.

> **Note**
> Network performance is not applicable for single servers. The benchmark described below is not available for single-server AnzoGraph deployments.

Running the Network Benchmark

1. In the Administration application, expand the **Connections** menu and select **AnzoGraph**. Anzo displays the AnzoGraph screen, which lists the connected AnzoGraph instances.
2. Click the name of the cluster that you want to evaluate. Anzo displays the Graphmarts screen for the cluster.
3. Click the **Diagnostics** tab and find the Network Benchmarking option at the bottom the screen. For example:



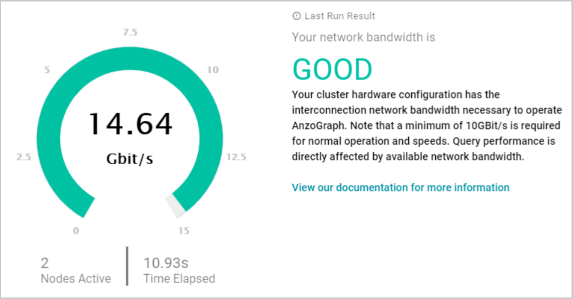
Run Benchmark to check the network bandwidth.

Test data in gigabytes (GB): 20 GB

Run benchmark to view results here.

2 Nodes Active | 0.00s Time Elapsed

4. By default, the benchmark is set to distribute 20 GB of data per node over the network. Each node in the cluster sends 20 GB to every other node. You can specify a different size if necessary. Note that increasing the value also increases the time to run the benchmark.
5. To run the test, click the **Run Benchmark** button. Anzo runs the benchmark and displays the results. For example:



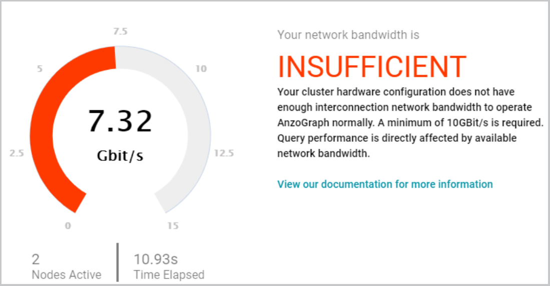
Your network bandwidth is

GOOD

Your cluster hardware configuration has the interconnection network bandwidth necessary to operate AnzoGraph. Note that a minimum of 10GBit/s is required for normal operation and speeds. Query performance is directly affected by available network bandwidth.

2 Nodes Active | 10.93s Time Elapsed

If the bandwidth is less than 10 Gbit/s, Anzo displays an "Insufficient" result. For example:



Your network bandwidth is

INSUFFICIENT

Your cluster hardware configuration does not have enough interconnection network bandwidth to operate AnzoGraph normally. A minimum of 10GBit/s is required. Query performance is directly affected by available network bandwidth.

2 Nodes Active | 10.93s Time Elapsed

When the results are insufficient, Cambridge Semantics recommends that you increase the network bandwidth. You can continue to use the cluster with the expectation of slower performance for network-bound operations.

Related Topics

Retrieving AnzoGraph Diagnostic Files

AnzoGraph Server Administration

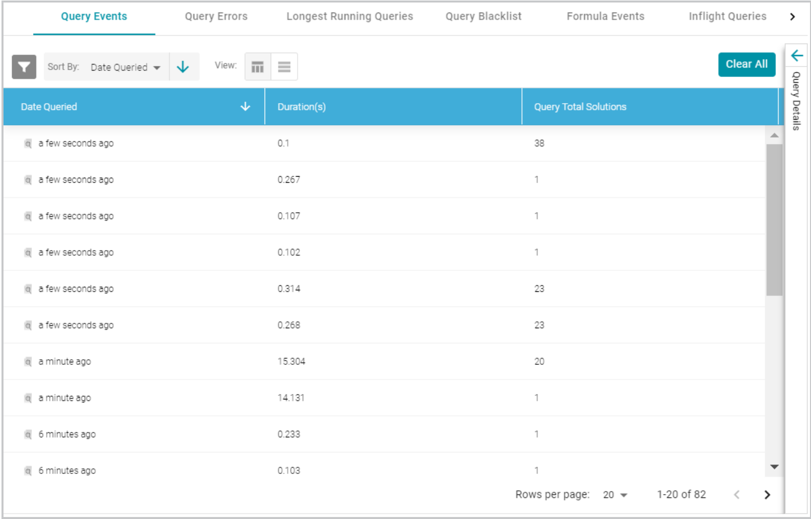
System Query Audit

The System Query Audit screen enables administrators to quickly view a log of query events, query errors, the duration time for the longest running queries, and a list of any queries that have been blacklisted. The audit log also includes a Queued Queries tab that displays a list of the queries that are queued behind currently running queries. Administrators can cancel queries from the list and remove them from the queue. This topic provides information about using the System Query Audit log.

- Viewing the System Query Audit Log
- AnzoGraph Detailed Query Timing Reference

Viewing the System Query Audit Log

In the Administration application, expand the **Monitoring & Diagnostics** menu and select **System Query Audit**. Anzo displays the Query Events log. For example:



Date Queried	Duration(s)	Query Total Solutions
a few seconds ago	0.1	38
a few seconds ago	0.267	1
a few seconds ago	0.107	1
a few seconds ago	0.102	1
a few seconds ago	0.314	23
a few seconds ago	0.268	23
a minute ago	15.304	20
a minute ago	14.131	1
6 minutes ago	0.233	1
6 minutes ago	0.103	1

Rows per page: 20 1-20 of 82

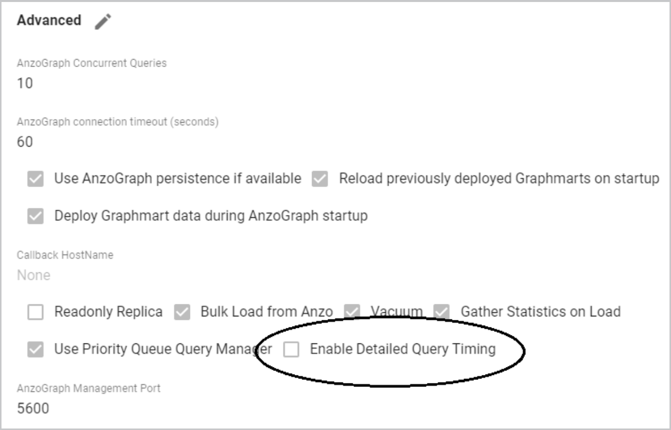
By default, the log shows an overview of all query events for all data sources. The table lists the date queried, the duration in milliseconds, and total number of solutions returned for each query event. You can select an event in the table to view details about that event, such as the target data source and query text, on the right side of the screen.

> **Note**
> The System Query Audit log does not report on queries that complete in less than 100 milliseconds. In addition, queries that reuse the query cache from a previous run are not captured in the log. However, if a query takes less than 100 ms and uses cache, the original entry for the query is

> updated to increase the Cache Hit count.

AnzoGraph Detailed Query Timing Reference

In the Advanced settings for the AnzoGraph connection configuration, there is an **Enable Detailed Query Timing** setting (shown in the image below) that controls the level of information that is displayed for AnzoGraph queries in the System Query Audit log. This section describes the differences in logging when the setting is enabled and disabled.



Advanced

AnzoGraph Concurrent Queries
10

AnzoGraph connection timeout (seconds)
60

☑ Use AnzoGraph persistence if available ☑ Reload previously deployed Graphmarts on startup

☑ Deploy Graphmart data during AnzoGraph startup

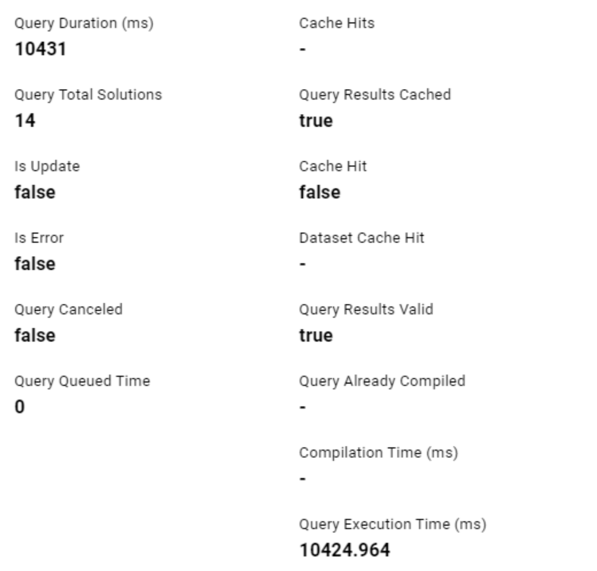
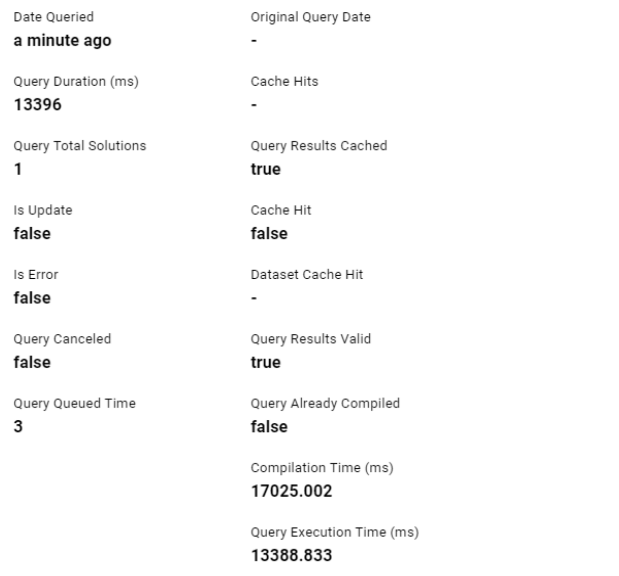
Callback HostName
None

□ Readonly Replica ☑ Bulk Load from Anzo ☑ Vacuum ☑ Gather Statistics on Load

☑ Use Priority Queue Query Manager □ Enable Detailed Query Timing

AnzoGraph Management Port
5600

Enable Detailed Query Timing is disabled by default, meaning that Anzo will not run the additional statistics gathering queries unless you enable the setting. When Enable Detailed Query Timing is disabled, the System Query Audit log displays fewer query timing details. For example, the images in the table below show a comparison between the **Result Details** tab when Enable Detailed Query Timing is disabled versus enabled. When the setting is disabled, details such as query Compilation Time are not recorded.

Enable Detailed Query Timing Disabled	Enable Detailed Query Timing Enabled
	

Query Duration (ms)
10431

Query Total Solutions
14

Is Update
false

Is Error
false

Query Canceled
false

Query Queued Time
0

Cache Hits
-

Query Results Cached
true

Cache Hit
false

Dataset Cache Hit
-

Query Results Valid
true

Query Already Compiled
-

Compilation Time (ms)
-

Query Execution Time (ms)
10424.964

Date Queried
a minute ago

Query Duration (ms)
13396

Query Total Solutions
1

Is Update
false

Is Error
false

Query Canceled
false

Query Queued Time
3

Original Query Date
-

Cache Hits
-

Query Results Cached
true

Cache Hit
false

Dataset Cache Hit
-

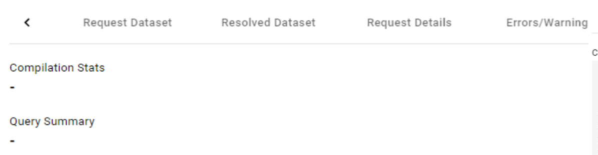
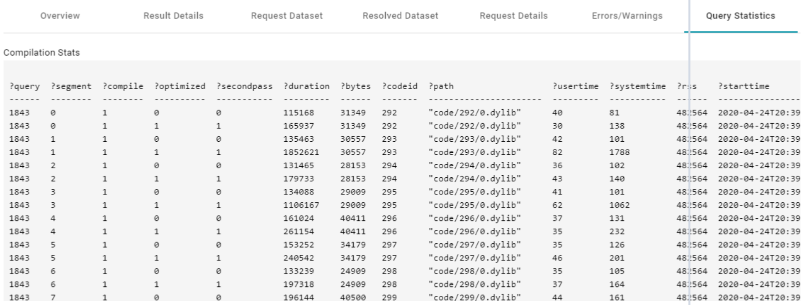
Query Results Valid
true

Query Already Compiled
false

Compilation Time (ms)
17025.002

Query Execution Time (ms)
13388.833

In addition, the images in the following table show a comparison between the **Query Statistics** tab when Enable Detailed Query Timing is disabled versus enabled. When the setting is disabled, the Compilation Stats and Query Summary tables are empty.

Enable Detailed Query Timing Disabled	Enable Detailed Query Timing Enabled
	

Compilation Stats
-

Query Summary
-

Compilation Stats

```
?query  ?segment  ?compile  ?optimized  ?secondpass  ?duration  ?bytes  ?codeid  ?path                   ?usertime  ?systemtime  ?rss    ?starttime
------  --------  --------  ----------  -----------  ---------  ------  -------  ----------------------  ---------  -----------  ------  ----------------
1843    0         1         0           0            115168     31349   292      "code/292/0.dylib"      40         81           482564  2020-04-24T20:39
1843    0         1         1           1            165937     31349   292      "code/292/0.dylib"      30         138          482564  2020-04-24T20:39
1843    1         1         0           0            135463     30557   293      "code/293/0.dylib"      42         101          482564  2020-04-24T20:39
1843    1         1         1           1            1852621    30557   293      "code/293/0.dylib"      82         1788         482564  2020-04-24T20:39
1843    2         1         0           0            131465     28153   294      "code/294/0.dylib"      36         102          482564  2020-04-24T20:39
1843    2         1         1           1            179733     28153   294      "code/294/0.dylib"      43         140          482564  2020-04-24T20:39
1843    3         1         0           0            134088     29009   295      "code/295/0.dylib"      41         101          482564  2020-04-24T20:39
1843    3         1         1           1            1106167    29009   295      "code/295/0.dylib"      62         1062         482564  2020-04-24T20:39
1843    4         1         0           0            161024     40411   296      "code/296/0.dylib"      37         131          482564  2020-04-24T20:39
1843    4         1         1           1            261154     40411   296      "code/296/0.dylib"      35         232          482564  2020-04-24T20:39
1843    5         1         0           0            153252     34179   297      "code/297/0.dylib"      35         126          482564  2020-04-24T20:39
1843    5         1         1           1            240542     34179   297      "code/297/0.dylib"      46         201          482564  2020-04-24T20:39
1843    6         1         0           0            133239     24909   298      "code/298/0.dylib"      35         105          482564  2020-04-24T20:39
1843    6         1         1           1            197318     24909   298      "code/298/0.dylib"      37         164          482564  2020-04-24T20:39
1843    7         1         0           0            196144     40500   299      "code/299/0.dylib"      44         161          482564  2020-04-24T20:39
```

To enable detailed query timing, edit the AnzoGraph connection and select the **Enable Detailed Query Timing** checkbox. You do not need to restart Anzo or AnzoGraph after changing the setting.

> **Important**
> Enabling detailed query timing increases the AnzoGraph workload and may decrease overall query performance.

Related Topics

AnzoGraph Server Administration

The topics in this section provide reference information and instructions for performing administrative tasks on an AnzoGraph server. Some tasks, such as modifying server configuration settings, cannot be done via the Anzo Administration application. Other tasks, such as starting and stopping AnzoGraph using the system manager, are documented as alternate methods of managing AnzoGraph if the Administration application is unavailable or you prefer to use the AnzoGraph command line interface.

Starting and Stopping AnzoGraph

This topic provides instructions for starting and stopping AnzoGraph.

> **Note**
> The system management daemon, **azgmgrd**, should remain running at all times. When you restart the database, do not stop and start the daemon. There are two circumstances that require you to restart azgmgrd:
>
> 1. When Upgrading AnzoGraph.
> 2. When making changes to the `<install_path>/config/ip_addrs.conf` file if you add or remove servers from an AnzoGraph cluster.

Follow the appropriate instructions below, depending on the current state of AnzoGraph and your use case:

- Stop the Database and Leave the System Management Daemon Running
- Start the Database (the System Management Daemon is Running)
- Stop the Database and the System Management Daemon
- Start the System Management Daemon and the Database
- Reinitializing the Database

Stop the Database and Leave the System Management Daemon Running

To stop the database, run one of the following commands from the **leader server**:

- If services are set up, run the following command:

  ```
  sudo systemctl stop anzograph
  ```

- If services are not set up, stop the database with the following system manager command:

  ```
  <install_path>/bin/azgctl -stop
  ```

  > **Important**
  > Make sure that you are logged in as the Anzo service account user any time you start and stop AnzoGraph using the system manager commands.

If queries are running, the system manager waits the number of seconds in stop_timeout (the default value is 30 seconds) for any outstanding queries to complete and then stops the database.

Start the Database (the System Management Daemon is Running)

To start the database, run one of the following commands from the **leader server**:

- If services are set up, run the following command:

  ```
  sudo systemctl start anzograph
  ```

- If services are not set up, start the database with the following system manager command:

  ```
  <install_path>/bin/azgctl -start
  ```

  > **Important**
  > Make sure that you are logged in as the Anzo service account user any time you start and stop AnzoGraph using the system manager commands.

Stop the Database and the System Management Daemon

To stop the database and system management daemon, run the appropriate commands from the **leader server**:

- If services are set up, run the following commands on the leader server to stop the database and daemon on all servers in the cluster:

  ```
  sudo systemctl stop anzograph
  ```

  ```
  sudo systemctl stop azgmgrd
  ```

- If services are not set up, run the following commands on the leader server to stop the database and daemon on all servers in the cluster:

  ```
  <install_path>/bin/azgctl -stop
  ```

  ```
  <install_path>/bin/azgctl -stopdaemon
  ```

> **Important**
> Make sure that you are logged in as the Anzo service account user any time you start and stop AnzoGraph using the system manager commands.

Start the System Management Daemon and the Database

To start the system management daemon, run one of the following commands. On clusters, run the command on **each server in the cluster**:

- If services are set up, run the following command on all servers in the cluster:

  ```
  sudo systemctl start azgmgrd
  ```

- If services are not set up, run the following command on all servers in the cluster:

  ```
  <install_path>/bin/azgmgrd
  ```

  > **Important**
  > Make sure that you are logged in as the Anzo service account user any time you start and stop AnzoGraph using the system manager commands.

To start the database after the system management daemon is running, run one of the following commands on the **leader node**:

- If services are set up, run the following command:

  ```
  sudo systemctl start anzograph
  ```

- If services are not set up, start the database with the following system manager command:

  ```
  <install_path>/bin/azgctl -start
  ```

Reinitializing the Database

If you need to reinitialize the database to remove the generated code and any persisted data, run the following command. The system management daemon (azgmgrd) should be running.

```
<install_path>/bin/azgctl -start -init
```

Configuring AnzoGraph for Kerberos Authentication

If you plan to load data to AnzoGraph from an HDFS file store that uses Kerberos authentication, follow the steps below to configure AnzoGraph for Kerberos authentication.

1. In order to be able to generate an authentication token for requesting encrypted ticket-granting tickets (TGT) from the key distribution center (KDC), each AnzoGraph host server must include the Kerberos workstation package, **krb5-workstation**. On each server in the cluster, run the following command to install the package:

   ```
   sudo yum install -y krb5-workstation
   ```

2. In order to establish a connection to the KDC, AnzoGraph must have a copy of the KDC's **krb5.conf** file. Place a copy of krb5.conf in the **/etc** directory on each AnzoGraph host server.
3. In addition to krb5.conf, each AnzoGraph server needs a copy of the **.keytab** file from the principal node. The keytab file and principal name are used to generate an authentication token.

   > **Note**
   > To find the location of the .keytab file and the principal name, you can look up the `dfs.web.authentication.kerberos.keytab` and `dfs.web.authentication.kerberos.principal` values in **hdfs-site.xml** on the HDFS master node.

   Copy the .keytab file to any location on each AnzoGraph host server, and then run the following command to generate the authentication token:

   ```
   kinit -p <principal_name> -k -t <path>/<keytab_file>
   ```

   Where <principal_name> is the Kerberos principal name and <path>/<keytab_file> is the location and name of the .keytab file.

Related Topics

Connecting to a File Store

Using the AnzoGraph CLI

You can use the **azgi** command line interface (CLI) in the `<install_path>/bin` directory to issue commands directly to the database.

> **Important**
> The azgi CLI works on the SPARQL HTTPS port and is enabled only when SSL protocol is enabled. SSL access is controlled by the enable_ssl_protocol setting. If HTTPS access is disabled and you want to enable it so that you can use the CLI, see Changing AnzoGraph Server Settings for instructions.

This section describes the available azgi commands. To view the help, run `azgi -help`.

AZGI Usage

```
azgi [-f <filename>] [-c "<command>"] [-set <param>=<value>] [-h <host_url>] [-p
<port>]
     [-u <username>:<password>] [-v] [-timer] [-raw] [-csv] [-json] [-xml] [-
silent]
     [-nohead] [-noprogress] [-maxwid <width>] [-wide]
     [-nossl] [-o <file>] [-certs <directory>] [-context <json_file>]
```

-f <filename>

Runs the specified SPARQL query file. For example, the following command runs the query or queries in the query.rq file:

```
azgi -f /home/user/query.rq
```

-c "<command>"

Runs the command in quotation marks. For example, this command runs a query:

```
azgi -c "select distinct ?eventname from
<http://cambridgesemantics.com/tickit>
where {?event <http://cambridgesemantics.com/tickit/eventname> ?eventname}
limit 100"
```

You can include multiple -c options to run multiple commands. For example, this command runs two queries:

```
azgi -c "select * from <http://cambridgesemantics.com/tickit> where {?s ?p
?o} limit 100"
-c "select distinct ?likes from <http://cambridgesemantics.com/tickit> where
{?person <http://cambridgesemantics.com/like> ?likes}"
```

And this command sets the query_label configuration setting to "events" before running the query:

```
azgi -c "set query_label to 'events'" -c "select distinct ?eventname
from <http://cambridgesemantics.com/tickit> where {?event
<http://cambridgesemantics.com/eventname> ?eventname}
limit 100"
```

-set <param>=<value>

Sets or changes parameter values in query files. For example this command runs the query in the query_summary.rq file with the $query parameter set to 2:

```
azgi -set query=2 -f query_summary.rq
```

-h <host_url>

Connects to a remote AnzoGraph server. For example, the following statement runs a query against AnzoGraph on host 10.104.55.27:

```
azgi -h 10.104.55.27 -c "select * from <http://cambridgesemantics.com/tickit>
where {?s ?p ?o} limit 100"
```

-p <port>

Used to connect to AnzoGraph on a non-default port. The default azgi port is 8256.

-u <username>:<password>

Connects to the database with credentials (basic authentication). If you type -u <username> and exclude the password, the client prompts for the password. For example, the following command uses basic authentication to run a query:

```
azgi -u admin:Passw0rd1 -c "select ?g where {graph ?g {?s ?p ?o}} limit 100"
```

-v

Displays verbose output such as client connection details. For example:

```
azgi -v -c "select distinct ?p from <http://cambridgesemantics.com/tickit>
where {<http://cambridgesemantics.com/person1> ?p ?o}"
```

```
Connecting to host=localhost port=8256
IPv4: connected
POST /sparql HTTP/1.1
Host: Anon
Accept: application/sparql-results+xml
User-Agent: azgi
Connection: keep-alive
Content-Length: 38
Content-Type: application/sparql-query
select distinct ?p from <http://cambridgesemantics.com/tickit> where
{<http://cambridgesemantics.com/person1> ?p ?o}
HTTP/1.1 200 OK
Date: Tue, 30 Jun 2020 00:24:42 GMT
Server: AnzoGraph
Access-Control-Allow-Origin: *
X-AnzoGraph-QueryExecution-Time: 20
Connection: close
Content-Type: application/sparql-results+xml; charset=utf-8
...
```

-timer

Reports query execution time in milliseconds.

-raw

Displays query results in raw XML, JSON, or CSV format, depending on what format you request.

-csv

Displays results in CSV format.

-json

Displays results in JSON format.

-xml

Displays results in XML format.

-silent

Suppresses the query output.

-nohead

Suppresses headings in query results.

-noprogress

Suppresses the progress messages that are displayed for queries that are in flight.

-maxwid <width>

Overrides the default maximum column width of 50 characters for tabular query results. Using the -wide option described below is equivalent to `maxwid 60000`.

-wide

Increases the column width for tabular query results from the default 50 characters to 60,000 characters. Equivalent to `-maxwid 60000`.

-nossl

Instructs the client to make a non-SSL (HTTP) connection to the database. When using AZGI to send a request to a remote AnzoGraph server, include the -h <host_url> and -p <port> options when using -nossl. The default HTTP port is 7070. For example:

```
azgi -nossl -h 10.100.0.20 -p 7070 -c "select (count(*) as ?cnt) where {?s ?p ?o}"
```

-o <file>

Writes the response to the specified file. If the file exists, it is overwritten.

> **Note**
> When you specify this option to redirect output to a file, all progress messages will also be written to the file unless you also specify the -noprogress option. Cambridge Semantics recommends that you include -noprogress any time you output results to a file.

-certs <directory>

Instructs the client to make a certified secure connection to the database. The AnzoGraph certificates are **ca.crt**, **serv.crt** (public key), and **serv.key** (private key) in the `<install_path>/config` directory. When sending requests to a remote AnzoGraph server, you can copy the AnzoGraph certificates to the server where you are using AZGI. For example, the following command runs a query on a remote AnzoGraph server. The command makes a certified connection using the AnzoGraph certificates, which were copied to the `/home/user/certs` directory:

```
azgi -h 10.10.10.01 -certs /home/user/certs
-c "select ?g where {graph ?g {?s ?p ?o}} limit 100"
```

This command runs the same query from the AnzoGraph server.

```
azgi -certs /opt/cambridgesemantics/anzograph/config -c "select ?g where
{graph ?g {?s ?p ?o}} limit 100"
```

-context <json_file>

Specifies the query context file on the AnzoGraph server file system to use with the request. Context files are JSON-formatted files with key-value pairs that provide connection details, such as user credentials, keys, and tokens, for authentication against data sources. For example:

```
{
  "url": "jdbc:mysql://10.111.4.9:3306/NORTHWIND",
  "username": "sysadmin",
  "password": "admin123"
}
```

Related Topics

Changing AnzoGraph Server Settings

AnzoGraph Settings Reference

AnzoGraph Settings Reference

This topic provides reference information for each of the AnzoGraph system configuration settings. The configuration file, `<install_path>/config/settings.conf`, categorizes the settings as either **Basic** or **Advanced**. The advanced-level settings should only be configured by system administrators or users with an advanced level of knowledge about AnzoGraph or databases in general. For instructions on changing settings, see Changing AnzoGraph Server Settings.

Basic

- enable_persistence
- enable_sparql_protocol
- enable_ssl_protocol
- internal_directory
- max_memory
- output_format
- persistence_directory
- sparql_protocol_port
- sparql_spec_default_graph
- spill_directory
- ssl_protocol_port
- startup_info
- stop_timeout
- truncate_clob
- use_custom_ssl_files
- user_queues

Advanced

- anzo_protocol_port
- auto_restart_directory
- auto_restart_max_attempts
- auto_restart_time

- aws_log_level
- aws_search_regions
- azgmgrd_client_auth
- azgmgrd_password
- bits_per_pred_index
- bits_per_uri_index
- blank_node_name
- call_home_for_updates
- comm_port_base
- compile_concurrent
- compile_max_memory
- compile_max_seconds
- compile_optimized
- copy_file_size
- enable_owlstats
- enable_refresh_stats_on_update
- enable_root_user
- enable_unbound_variables
- float_decimals
- float_format
- grpc_token_expiry
- ignore_deniedlist_queries
- jvm_max_memory
- jvm_options
- log_directory
- paged_data
- paged_cache_memory_percent
- policy_file_enabled

enable_persistence

Default Value: **false** (boolean)

This setting controls whether AnzoGraph's saves a copy of the data in memory to disk. For more information, see Using AnzoGraph Persistence (Preview).

enable_sparql_protocol

Default Value: **false** (boolean)

This setting controls whether to enable the HTTP SPARQL endpoint. The sparql_protocol_port setting controls the port to use to access the endpoint.

> **Note**
> Enabling the SPARQL HTTP protocol opens the standard SPARQL-compliant HTTP endpoint. Unlike the Anzo protocol endpoint, the SPARQL HTTP endpoint is not secured.

enable_ssl_protocol

Default Value: **false** (boolean)

This setting controls whether to enable the secure HTTPS SPARQL endpoint. The ssl_protocol_port setting controls the port to use.

> **Note**
> Enabling the SPARQL HTTPS protocol opens the standard SPARQL-compliant HTTPS endpoint. Unlike the Anzo protocol endpoint, the SPARQL HTTPS endpoint is encrypted but not authenticated.

internal_directory

Default Value: Not set (char). The default directory is **<install_path>/internal**.

The directory where AnzoGraph should save internal database-related files such as generated code, logs, and query plans. For more information, see Relocating AnzoGraph Directories.

max_memory

Default Value: System-based (int)

Specifies the amount of memory (in MB) that is available for AnzoGraph. The default is system-based; at startup, AnzoGraph determines the amount of RAM that is available and sets max_memory. In test environments where AnzoGraph may be co-located with other programs, you can set the max_memory value to put a limit on the amount of memory AnzoGraph can use. However, Cambridge Semantics recommends that you do not set max_memory unless instructed by Support.

output_format

Default Value: **xml** (char)

Specifies the default output format for AnzoGraph responses. Valid values are **xml**, **json**, or **csv**.

persistence_directory

Default Value: Not set (char). The default directory is **<install_path>/persistence**.

The directory where AnzoGraph should save data when enable_persistence is **true** and data is persisted to disk. For more information, see Relocating AnzoGraph Directories.

sparql_protocol_port

Default Value: **7070** (int)

This setting specifies the port to use to access the SPARQL HTTP endpoint when enable_sparql_protocol is **true**.

sparql_spec_default_graph

Default Value: **false** (boolean)

Controls the default scope of SPARQL queries when FROM clauses are excluded from a query. When **false**, queries without FROM clauses target the default graph (DEFAULTSET) only. Triples in named graphs will not be included in the scope of the query. When **true**, AnzoGraph conforms to the SPARQL specification and includes the default graph and all named graphs in the scope of a query that omits the FROM clause. For more information, see Changing the Default FROM Clause Behavior.

spill_directory

Default Value: Not set (char). The default directory is **<install_path>/spill**.

The directory where AnzoGraph should save temporary query files that spill to disk. For more information, see Relocating AnzoGraph Directories.

> **Important**
> AnzoGraph uses O_DIRECT to read the spill files into the database. If you relocate the spill directory, make sure to place it on an ext4 file system that supports O_DIRECT.

ssl_protocol_port

Default Value: **8256** (int)

This setting specifies the port to use to access the SPARQL HTTPS endpoint when enable_ssl_protocol is **true**.

startup_info

Default Value: **1** (int)

Specifies how verbose the database startup message is: **- 0**-quiet, **1**-ready, **2**-ports, **3**-more.

stop_timeout

Default Value: **30** (int)

When the database stop command is issued, this setting specifies the number of seconds to wait for queries to finish before stopping the database.

truncate_clob

Default Value: **false** (boolean)

Specifies whether to automatically truncate large strings to the maximum string size (2 MB).

use_custom_ssl_files

Default Value: **false** (boolean)

Indicates whether you are replacing AnzoGraph's self-signed certificates with your own custom certificates. To configure AnzoGraph to use your certificates, follow the instructions in Replace the Default Self-Signed Certificates with Trusted Certificates in the Deployment Guide.

> **Important**
> Anzo also needs to trust the new certificates. Make sure you have Trust All TLS Certificates enabled on the AnzoGraph connection or make sure Anzo's trust store has either the certificate for the CA that signed the certificate or the certificate itself.

user_queues

Default Value: **40** (int)

Sets the limit on the number of queries that can run concurrently.

anzo_protocol_port

Default Value: **5700** (int)

The Anzo protocol (gRPC) port for secure communication between AnzoGraph and Anzo.

auto_restart_directory

Default Value: Not set (char). The default directory is **<install_path>/internal**.

Specifies the base location of the **auto_restart** directory, which contains the denied_list, warned_list, and unanalyzed_list directories. For more information about the auto-restart feature, see Managing the Automatic Restart Feature.

auto_restart_max_attempts

Default Value: **5** (int)

Specifies the number of times the system manager should attempt to start the database after a crash. The default value is **5**, which means the system manager will attempt to restart the database a maximum of 5 times. Changing auto_restart_max_attempts to **0** disables the auto-restart feature. For more information about the auto-restart feature, see Managing the Automatic Restart Feature.

auto_restart_time

Default Value: **600** (int)

Specifies the number of seconds to spend attempting to restart the database. If all attempts fail and this time limit is reached, the system manager stops trying to restart the database. The default value is **600**, which means that the system manager will attempt to restart the database for a maximum of 600 seconds (10 minutes). For more information about the auto-restart feature, see Managing the Automatic Restart Feature.

aws_log_level

Default Value: **2** (int)

AnzoGraph uses an AWS C++ SDK for loading data from S3. This setting controls the logging level for the AWS SDK. The default value is **2**, which is Error level logging. Valid values are:

- 0 (Off)
- 1 (Fatal)
- 2 (Error)
- 3 (Warn)
- 4 (info)
- 5 (Debug)
- 6 (Trace)

aws_search_regions

Default Value: Not set (char)

Lists the regions to search for AWS S3 buckets that are listed as file locations for LOAD queries.

azgmgrd_client_auth

Default Value: **false** (boolean)

Controls whether the system management daemon (azgmgrd) and system manager (azgctl) use authentication in addition to encryption when connecting to other system managers over the system management gRPC port (5600). The default value is **false**, which means the system management connections are encrypted but not authenticated. For more information about azgmgrd authentication, see Enable System Manager Authentication in the Deployment Guide.

azgmgrd_password

This is the password that the system management daemon (azgmgrd) uses for gRPC access to the database. Typically this value is not changed as it is only used internally for authentication between the system manager and the database. If you do want to change the password, you cannot change it directly in the settings.conf file. See Change the System Manager Password in the Deployment Guide for instructions.

bits_per_pred_index

Default Value: **16** (int)

This setting specifies the maximum number of unique graph and predicate URIs that can be stored in AnzoGraph. The maximum number is two to the power of this value. The default value (16) for bits_per_pred_index is set to the maximum value and should not be changed. 2^16 = 64k unique predicate and graph URIs.

bits_per_uri_index

Default Value: **32** (int)

This setting specifies the maximum number of unique subject URIs that can be stored in AnzoGraph. The maximum number is two to the power of this value. The default value (32) for bits_per_uri_index is set to the maximum value and should not be changed. 2^32=4+ trillion unique subject URIs.

blank_node_name

Default Value: **genid** (char)

This setting specifies the default name basis for blank nodes. By default, AnzoGraph generates a number ID for the node. For example, inserting `_:a` results in a URI such as `bnode:a__63`.

call_home_for_updates

Default Value: **false** (boolean)

This setting controls whether AnzoGraph checks for updates over the internet.

comm_port_base

Default Value: **9100** (int)

This setting specifies the port to use for internal cluster communication.

compile_concurrent

Default Value: **8** (int)

This setting specifies the maximum number of generated code compilations to perform concurrently.

compile_max_memory

Default Value: **500** (int)

Sets the limit on the amount of memory (in MB) that AnzoGraph can allocate for compiling generated code before switching from optimized compile to non-optimized compile.

compile_max_seconds

Default Value: **30** (int)

Sets the limit on the number of seconds to spend compiling generated code before switching from optimized compile to non-optimized compile.

compile_optimized

Default Value: **background** (char)

Specifies the type of optimized compile to perform.

copy_file_size

Default Value: **5** (int)

This setting controls the size (in MB) of the Turtle files that are generated when Graphmart contents are exported to files.

enable_owlstats

Default Value: **true** (boolean)

In order to generate query execution plans, AnzoGraph needs to gather statistics about the data, such as the number of triples per graph and number of distinct subjects and predicates. This setting controls whether advanced statistics gathering, called OWL stats, is enabled. OWL stats use the metadata from data models to generate statistics. Cambridge Semantics recommends that you leave enable_owlstats enabled unless otherwise instructed.

enable_refresh_stats_on_update

Default Value: **true** (boolean)

AnzoGraph's internal statistics gathering queries are triggered automatically when data is loaded. And the resulting statistics are used for all subsequent queries against the data. This setting controls whether AnzoGraph re-runs the statistics gathering queries whenever the data is updated, not just on the initial load.

enable_root_user

Default Value: **false** (boolean)

This setting controls whether to allow a user running with root privileges to start AnzoGraph.

enable_unbound_variables

Default Value: **false** (boolean)

Controls whether AnzoGraph returns an empty result or an error if a query references a missing graph or includes unbound variables. This value is set to **false** by default, which means AnzoGraph returns an error. For more information, see Ignoring Missing Graphs.

float_decimals

Default Value: **6** (int)

> **Important**
> This setting does not apply to results that are returned from AnzoGraph to Anzo over gRPC protocol. Anzo converts floating point values to Java native float objects with 6 – 7 total digits of precision. This setting would only affect results that are returned directly from AnzoGraph to another application over HTTP/S protocol.

AnzoGraph formats floating point types using the printf format string **%.precision format**, where **precision** is the value of the **float_decimals**, and **format** is the value of float_format.

> **Note**
> The interpretation of `float_decimals` differs depending on the value in float_format. For fixed point formats (f and F), `float_decimals` specifies the number of digits to include after the decimal point, padded with zeros if necessary. For floating point formats (e, E, g, and G), `float_decimals` specifies the number of significant digits to round the result to.

float_format

Default Value: **g** (char). In the default configuration, a value of 10000000000.123 is returned as 1e+10.

> **Important**
> This setting does not apply to results that are returned from AnzoGraph to Anzo over gRPC protocol. Anzo converts floating point values to Java native float objects with 6 – 7 total digits of precision. This setting would only affect results that are returned directly from AnzoGraph to another application over HTTP/S protocol.

AnzoGraph formats floating point types using the printf format string **%.precision format**, where **format** is the value of the **float_format**, and **precision** is the value of float_decimals. Valid values for **float_format** are **e**, **E**, **f**, **F**, **g**, or **G**.

grpc_token_expiry

Default Value: **0** (int).

This setting controls how often (in seconds) the gRPC token expires. A value of 0 means the token never expires.

ignore_deniedlist_queries

Default Value: **true** (boolean)

Controls whether denied list queries are blocked from running or are allowed to be run when the database is returned to normal operation. The default value is **true**, which means denied list queries are ignored. Incoming queries are not compared with the denied list and are permitted to run. If ignore_deniedlist_queries is **false**, denied list queries are not ignored and are therefore blocked from running until they are removed from the denied list. For more information about the auto-restart feature, see Managing the Automatic Restart Feature.

jvm_max_memory

Default Value: Not set (char). When not set, the default is 5% or 4g, depending on which value is smaller.

Specifies the maximum size of the heap that can be used by the embedded Java virtual machine (JVM). Use **k**, **m**, or **g** (case insensitive) for KiB, MiB, or GiB. You can also specify **%** to indicate a percentage of the total memory that is available to AnzoGraph. By default, this value is not set, which means jvm_max_memory defaults to either **5%** of the total memory or **4g**, whichever value is smaller.

jvm_options

Default Value: Not set (char)

Lists any optional parameters to use for configuring the embedded JVM. Use a semicolon-delimited (;) list to specify multiple parameters. For information about JVM options, see Options in the Java Documentation.

log_directory

Default Value: Not set (char). When not set, the default location is **/tmp**.

Specifies where to write system management daemon (azgmgrd) log files. These types of logs (azgmgrd.log, azgctl-<user>.log, azgpidmgr.log, and azgpids.log) are created before the system is initialized and may be written before the `<install_path>/internal/log` directory exists. Therefore, they are located outside of the AnzoGraph file system, **/tmp** by default. If you change the log_directory value, Cambridge Semantics recommends that you choose another location that is outside the internal AnzoGraph directories.

paged_data

Default Value: **false** (boolean)

Enables or disables AnzoGraph's paged data feature, which controls whether data is stored in memory or on disk. When this option is **false** (the default value), data is stored in memory. Setting this option to true changes data storage from in-memory to on-disk (in the persistence_directory).

> **Important**
> Enabling this option changes underlying database operations. Before enabling paged data, make sure that the performance and storage impacts are well-understood and that your environment meets the requirements. See Sizing Guidelines for Disk-Based Storage (Preview) in the Deployment Guide for details.

paged_cache_memory_percent

Default Value: **20** (int)

When paged_data is enabled, this setting controls the amount of memory (as a percentage of total memory) to use for caching the most recently requested data. The default value is **20**, which means AnzoGraph is configured to use 20% of the total available memory for caching data for analytics. For example, if you have 1 TB of data on disk and 300 GB of available RAM, AnzoGraph caches in memory 60 GB of the most recently accessed data. If a query requests data that is not currently cached, AnzoGraph releases the least accessed data from memory and loads the relevant data into memory. Note that a portion of the paged cache memory percent is used for the overhead of tracking the pages that are accessed. For more information, see Enabling Paged Data Mode (Preview).

> **Important**
> Cambridge Semantics recommends that you do not set this value higher than 30.

policy_file_enabled

Default Value: **false** (boolean)

Enables or disables file system access control policies. When `policy_file_enabled` is **false** (the default value), AnzoGraph does not perform file path access checks when a query reads or writes files or directories on the file system. When `policy_file_enabled` is **true** and a query attempts to access a file or directory on the file system, AnzoGraph performs the file path access checks that are configured in the `file_policy_*` settings and returns an access denied error message if the path is not accessible. For instructions on configuring file access policies and the file_policy_read, write, delete, and deny settings, see Managing AnzoGraph File Access Policies.

Related Topics

Changing AnzoGraph Server Settings

Changing AnzoGraph Server Settings

The default AnzoGraph system configuration is optimized for most AnzoGraph installations. If Cambridge Semantics Support recommends that you change the configuration, you can edit the configuration file, `<install_path>/config/settings.conf`, to modify or add settings. Each time you start the database, AnzoGraph reads this file and stores the configuration in memory. **On a cluster, change settings.conf on the leader server only**. See the AnzoGraph Settings Reference for information about the units of measurement for the settings as well as any special instructions.

- The commented lines in the file show the default configuration values. To customize the value for a setting that is commented out, uncomment the line and edit the value portion of `setting_name=`**`value`**.
- To add settings to settings.conf, add the setting and new value in the format below. Type each setting and value pair on a new line.

  ```
  setting_name=value
  ```

  > **Note**
  > AnzoGraph applies settings from the top to the bottom of the file. If the same setting appears more than once, AnzoGraph applies the value for the last instance of the setting. The last instance overrides any previous instances.

- To revert AnzoGraph to a previous configuration from a backup file, rename the existing settings.conf file and then change the name of the desired backup file to **settings.conf**.

> **Important**
> After you change settings.conf, you must restart AnzoGraph for the settings to take effect. See Starting and Stopping AnzoGraph for instructions.

Related Topics

Managing AnzoGraph File Access Policies

Relocating AnzoGraph Directories

Using AnzoGraph Persistence (Preview)

Ignoring Missing Graphs

Changing the Default FROM Clause Behavior

Managing the Automatic Restart Feature

Enabling Paged Data Mode (Preview)

AnzoGraph Settings Reference

Managing AnzoGraph File Access Policies

In AnzoGraph Version 2.5.6 and later, you can configure file system access control policies to ensure that only certain files or directories are accessible to AnzoGraph during the execution of a query. This topic describes the configuration settings that define the file access policies and provides instructions for setting up policies.

- File Access Policy Settings Reference
- File Access Control Behavior
- Setting Up File Access Policies

File Access Policy Settings Reference

policy_file_enabled

The **policy_file_enabled** setting is the parent setting that controls whether or not file system access policies are enabled and followed. When policy_file_enabled is **false** (the default value), AnzoGraph does not perform file path access checks when a query references files or directories on the file system. When policy_file_enabled is **true** and a query attempts to access a file or directory on the file system, AnzoGraph performs the file path access checks that are configured in the policy_file_read, write, delete, and deny settings described below.

policy_file_read, write, delete, and deny

The **policy_file_read, write, delete, and deny** settings specify the paths to directories and/or files on the file system that AnzoGraph requests are allowed to read from, write to, or delete from. For each of the "allowed" read, write, and delete settings, there is a corresponding **deny** setting that configures the paths for which requests are denied read, write, and delete access. This enables you to allow broad access to parent directories, if desired, and then use the deny settings to restrict access to certain subdirectories under them if needed.

The values for the settings are wildcard patterns that AnzoGraph uses to match directories and/or file names. Patterns are specified using basic file globbing syntax as described in the glob(7) Linux manual page. Each policy_file_* setting accepts one or more patterns. Separate multiple patterns with a semicolon (`;`). For readability, you can also include spaces between patterns.

> **Important**
> Prior to matching paths in an incoming request to the configured access policy patterns, AnzoGraph resolves the paths in the request to canonical paths (using the `std::filesystem::weakly_canonical` function described here at cppreference.com). That means segments such as `/./` or `/../` are fully expanded prior to being compared to patterns. If a segment in the request path is a symlink, that segment is also expanded prior to checking for a match. **Make sure that all access policy patterns match absolute paths**. Otherwise, expanded relative path or symlink segments in a request will not match any patterns. For example, if users normally include a path like `/source-files/` in a request but `/source-files/` is a symlink to `/mnt/anzoshare/data/source-files/`, include the path to `/mnt/anzoshare/data/source-files/` in the pattern.

The following list describes the settings and provides sample pattern values. The File Access Control Behavior section below includes specifics about pattern matching and access checks.

- **policy_file_read**: Specifies the pattern(s) to match for paths that queries have permission to read from. For example, a value such as the following gives AnzoGraph requests read-only access to all files and directories under the `/opt/anzoshare` and `/mnt/data` directories:

  ```
  policy_file_read=/opt/anzoshare/* ; /mnt/data/*
  ```

- **policy_file_read_deny**: Specifies the pattern(s) to match for paths that queries should not be allowed to read. For example, the following value means requests will not be allowed to read any files or directories under `/etc` or `/root`:

  ```
  policy_file_read_deny=/etc/* ; /root/*
  ```

- **policy_file_write**: Specifies the pattern(s) to match for paths that queries have permission to write to. For example, the following value gives requests write access to the `/tmp` and `/home` directories in addition to the `/opt/anzoshare/store` and `/mnt/data/store` directories.

  ```
  policy_file_write=/tmp/* ; /home/* ; /opt/anzoshare/store/* ;
  /mnt/data/store/*
  ```

  > **Important**
  > If you have Graphmarts with Export Steps, make sure the write policy gives AnzoGraph write access to the appropriate Anzo Data Store.

- **policy_file_write_deny**: Specifies the pattern(s) to match for paths that queries are denied write access to.
- **policy_file_delete**: Specifies the pattern(s) to match for paths that queries have permission to delete.
- **policy_file_delete_deny**: Specifies the pattern(s) to match for paths that queries are denied delete access to.

> **Note**
> The AnzoGraph installation path (`<install_path>/*`) is automatically added to each of the `*_deny` policies.

File Access Control Behavior

When a query that includes a path to a file or directory is run (such as in a GDI query with `s:url "/opt/anzoshare/data/csv"` or in a `LOAD <dir:/mnt/data/rdf.ttl.gz>` statement), AnzoGraph resolves that path (for example, if the path includes `./` or `../` segments) to a canonical path prior to checking whether it matches a policy_file pattern. If any segment of the path is a symlink, that segment is also expanded prior to being matched to a pattern. If the specified file or directory matches one of the allowed access patterns and it is not matched to a deny pattern, the query is executed. If the specified path is matched to a denied pattern or is not matched to any of the allowed patterns, the query is aborted and AnzoGraph returns an access denied error message.

Setting Up File Access Policies

1. Stop the database. See Stop the Database and Leave the System Management Daemon Running for instructions.
2. **On the leader node**, open the AnzoGraph settings file, **settings.conf**, in a text editor. The file is in the `<install_path>/config` directory.
3. In settings.conf, uncomment the `policy_file_enabled=false` line and change the value to true:

   ```
   policy_file_enabled=true
   ```

4. Locate the additional `policy_file_*` settings:

   ```
   # File system paths that may be deleted (';' delimited) ()
   # policy_file_delete=
   ```

```
# File system paths that may not be deleted (';' delimited) ()
# policy_file_delete_deny=

# File system paths that may be read from (';' delimited) ()
# policy_file_read=

# File system paths that may not be read from (';' delimited) ()
# policy_file_read_deny=

# File system paths that may be written to (';' delimited) ()
# policy_file_write=

# File system paths that may not be written to (';' delimited) ()
# policy_file_write_deny=
```

5. Uncomment each of the `policy_file_*=` lines that you want to set, and add the wildcard pattern or patterns that you want to match for each of the policies.
6. Save and close settings.conf.
7. Restart the database to apply the configuration change. See Start the Database (the System Management Daemon is Running) for instructions.

Related Topics

Changing AnzoGraph Server Settings

AnzoGraph Settings Reference

Starting and Stopping AnzoGraph

Relocating AnzoGraph Directories

Follow the instructions in this section to designate alternate locations for certain directories included in the AnzoGraph installation. You have the option to relocate the **persistence** directory where the system saves the data in memory to the file system, the **internal** directory where the system saves database-related files such as logs and generated code, and the **spill** directory where the system saves any temporary query files that spill to disk.

You can change the settings described in this section at any time. Once you restart the database, AnzoGraph starts saving any new files in the directory locations that you specify.

> **Note**
> The system does not relocate any existing directories or files. You can move the existing files manually if needed.

1. Stop the database. See Stop the Database and Leave the System Management Daemon Running for instructions.
2. **On the leader node**, open the AnzoGraph settings file, **settings.conf**, in a text editor. The file is in the `<install_path>/config` directory.
3. Uncomment the lines for any of the following settings in settings.conf. Then edit the value portion of `setting=value` to specify the desired directory.
   - internal_directory: The directory where you want AnzoGraph to save internal database-related files such as generated code, logs, and query plans. The default value is `<install_ path>/internal`.
   - persistence_directory: The directory where you want AnzoGraph to save data when writing data to disk. The default value is `<install_path>/persistence`.
   - spill_directory: The directory where you want the AnzoGraph to save any temporary query files that spill to disk. The default value is `<install_path>/spill`.

   > **Important**
   > AnzoGraph uses O_DIRECT to read the spill files into the database. If you relocate the spill directory, make sure to place it on an ext4 file system that supports O_DIRECT.
4. Save and close settings.conf.
5. Restart the database to apply the configuration change. See Start the Database (the System Management Daemon is Running) for instructions.

Related Topics

Changing AnzoGraph Server Settings

Starting and Stopping AnzoGraph

Using AnzoGraph Persistence (Preview)

By default, Anzo manages the data in AnzoGraph by automatically reloading Graphmart data into memory when AnzoGraph is restarted. You also have the option to enable persistence on the AnzoGraph instance. When persistence is enabled, AnzoGraph saves the data in memory to disk after every transaction. Each time AnzoGraph is restarted, the persisted data is automatically loaded back into memory. Once the data is loaded into memory, rather than automatically reloading active Graphmarts, Anzo checks to see if the last updated timestamp in AnzoGraph matches the last updated value in Anzo. If the timestamps match, Anzo does not initiate a reload. If there is a mismatch, Anzo reloads the active Graphmarts to update the data in memory to the latest version.

> **Note**
> The AnzoGraph persistence feature is available as a **Preview** release, which means the implementation has recently been completed but is not yet thoroughly tested with Anzo and could be unstable. The feature is available for trial usage, but Cambridge Semantics recommends that you do not rely on Preview features in production environments.

This topic lists important information to consider before enabling persistence and provides instructions for enabling persistence in the AnzoGraph configuration file.

Important Considerations

Before enabling persistence, consider the following important notes:

- In general, each AnzoGraph server needs access to about twice as much disk space as RAM on the server. By default, AnzoGraph saves data to the `<install_path>/persistence` directory on the local file system. You can also configure AnzoGraph to save data to a mounted file system. For more information, see Relocating AnzoGraph Directories.
- Persisted data is unique to each AnzoGraph version and cannot be re-used after an upgrade. If you upgrade AnzoGraph and persistence is enabled, the database will not start until it is reinitialized to remove the persisted data. See Reinitializing the Database for instructions.
- When persistence is enabled, transactional workloads that perform many concurrent write operations may experience a performance degradation due to the overhead of writing the data from each transaction to disk.

Enabling Persistence

Follow the steps below to enable the AnzoGraph save to disk option.

1. Stop the database. See Stop the Database and Leave the System Management Daemon Running for instructions.
2. **On the leader node**, open the AnzoGraph settings file, **settings.conf**, in a text editor. The file is in the `<install_path>/config` directory.
3. In settings.conf, find the following line in the file:

   ```
   enable_persistence=false
   ```

4. Change the enable_persistence value to **true**:

   ```
   enable_persistence=true
   ```

5. Save and close settings.conf.
6. Restart the database to apply the configuration change. See Start the Database (the System Management Daemon is Running) for instructions.

After each transaction, AnzoGraph saves the data in memory to disk in the location specified in the persistence_directory setting. Each time AnzoGraph is restarted, the persisted data is automatically loaded back into memory.

> **Note**
> To avoid unnecessary reloads, make sure that the AnzoGraph connection in Anzo is configured to enable the **Use AnzoGraph persistence if available** option. See Connecting to AnzoGraph for more information.

Related Topics

Connecting to AnzoGraph

Relocating AnzoGraph Directories

Starting and Stopping AnzoGraph

Ignoring Missing Graphs

By default, AnzoGraph returns a "No such graph or view" error and aborts the query if a query references a graph that does not exist. You can configure AnzoGraph to conform to the SPARQL specification and return an empty result instead of an error, however, if a query references a missing graph. Follow the instructions below to configure the system to return empty results instead of an error when a referenced graph does not exist.

1. Stop the database. See Stop the Database and Leave the System Management Daemon Running for instructions.
2. **On the leader node**, open the AnzoGraph settings file, **settings.conf**, in a text editor. The file is in the `<install_path>/config` directory.
3. In settings.conf, uncomment the `enable_unbound_variables=false` line and change the value to true:

   ```
   enable_unbound_variables=true
   ```

4. Save and close settings.conf.
5. Restart the database to apply the configuration change. See Start the Database (the System Management Daemon is Running) for instructions.

Note

In addition to allowing queries that reference non-existent graphs to succeed, setting enable_unbound_variables to true also configures AnzoGraph to ignore unbound variables elsewhere in queries. For example, by default (when enable_unbound_variables=false), if a query includes a variable in the SELECT list that is not referenced in a WHERE clause pattern, AnzoGraph aborts the query and returns a "Named variable not in contained WHERE clause" error. When enable_unbound_variables=true, AnzoGraph does not warn the user about unbound variables. Instead, the results are empty for the unbound variable. For example:

```
SELECT ?unbound ?person ?name
FROM <http://cambridgesemantics.com/people>
WHERE {?person <http://cambridgesemantics.com/people#firstname> ?name}
LIMIT 5
```

```
 unbound | person       | name
---------+-------------+---------
```

```
         | person35632 | Ross
         | person20216 | Quin
         | person35859 | Kellie
         | person2551  | Maris
         | person24963 | Madonna
5 rows
```

Related Topics

Changing AnzoGraph Server Settings

AnzoGraph Settings Reference

Changing the Default FROM Clause Behavior

By default, if a query omits FROM clauses, the scope of the query is limited to the default graph (DEFAULTSET). Triples in named graphs will not be included in the scope of the query. The default behavior is controlled by the sparql_spec_default_graph configuration setting. To configure AnzoGraph to conform to the SPARQL specification and include the default graph and all named graphs in the scope of a query that omits the FROM clause, follow the instructions below.

1. Stop the database. See Stop the Database and Leave the System Management Daemon Running for instructions.
2. **On the leader node**, open the AnzoGraph settings file, **settings.conf**, in a text editor. The file is in the `<install_path>/config` directory.
3. In settings.conf, uncomment the `sparql_spec_default_graph=false` line and change the value to true:

   ```
   sparql_spec_default_graph=true
   ```

4. Save and close settings.conf.
5. Restart the database to apply the configuration change. See Start the Database (the System Management Daemon is Running) for instructions.

Related Topics

Changing AnzoGraph Server Settings

AnzoGraph Settings Reference

Managing the Automatic Restart Feature

AnzoGraph can be configured so that the system manager automatically restarts the database and evaluates the queries that were running if AnzoGraph shuts down unexpectedly. This topic describes the process that occurs when AnzoGraph automatically restarts and provides information about the configuration settings that control the functionality as well as administrative information for managing the evaluated queries.

- Automated Restart Procedure
- Automated Restart System Settings
- Removing a Query from the Block List

Automated Restart Procedure

The steps below describe what occurs during the automatic restart process after AnzoGraph has crashed:

1. The system manager restarts the database in **safe mode**. In safe mode, AnzoGraph is locked to users and returns the following message if a user runs a query: "AnzoGraph is running in safe-mode. Cannot execute query." In addition, running `azgctl -status` to check the status of the database returns the message "AnzoGraph is running in safe-mode." If persistence is enabled, the data that was in memory at the time of the crash is reloaded into memory.
2. While in safe mode, AnzoGraph runs any queries that were inflight at the time of the crash. By executing the queries that were running, AnzoGraph tries to determine if the crash was directly caused by one of the inflight queries.
3. Depending on the outcome of running the inflight queries, AnzoGraph does the following:
   - If all inflight queries run to completion in safe mode, they are all added to the **warned_list**. In addition, each query is copied to a file named `<query_ID>.txt` in the `<install_path>/internal/auto_restart/<timestamp>/warned_list` directory.

     > **Note**
     > When all inflight queries complete successfully, that means it is unlikely that any one of the queries on its own is the culprit for the crash. However, all of the queries are added to the warned list because it is possible that the combination of queries run concurrently could have caused the crash.

- If any of the inflight queries fail or crash the database in safe mode, those queries are added to the **denied_list**. In addition, each query is copied to a file named `<query_ID>.txt` in the `<install_path>/internal/auto_restart/<timestamp>/denied_list` directory.

  > **Note**
  > If an inflight query fails, none of the inflight queries are added to the warned list. Instead, the failed queries are added to the denied list.

- If AnzoGraph runs a query in safe mode and cannot determine if it should be added to the denied or warned list, those queries are copied to a file named `<query_ID>.txt` in the `<install_path>/internal/auto_restart/<timestamp>/unanalyzed_list` directory.
- Metadata about the warned_list, denied_list, and unanalyzed_list queries is captured in the **stc_blocklist** system table.

> **Note**
> The **auto_restart_directory** setting in the system configuration file, `<install_path>/config/settings.conf`, controls the location of the auto_restart directories listed above. For more information about the setting, see the Automated Restart System Settings section below.

4. After the inflight queries have been run, AnzoGraph restarts the database, loads the persisted data back into memory, and returns the system to normal operation.

To help prevent the circumstance that caused the database to crash, any queries that were added to the **denied** list are blocked from being executed when the system returns to normal operation. When a user runs a query, AnzoGraph compares that query with the denied list. If the query is on the list, the query is terminated and AnzoGraph returns an "Attempting to execute a denied-listed query" error message. Queries on the warned list are not blocked. A denied list query cannot be run unless it is removed from the denied list. This behavior is controlled by the **ignore_deniedlist_queries** setting. For more information about the setting, see the Automated Restart System Settings section below. For information about removing queries from the denied list, see Removing a Query from the Block List below.

Automated Restart System Settings

The automatic restart feature is controlled by the following four settings in `<install_path>/config/settings.conf`:

- **auto_restart_max_attempts**: This setting specifies the number of times the system manager should attempt to start the database after a crash. The default value is **5**, which means the system manager will attempt to restart the database a maximum of 5 times. Changing auto_restart_max_attempts to **0** disables the auto-restart feature.
- **auto_restart_time**: This setting specifies the number of seconds to spend attempting to restart the database. If all attempts fail and this time limit is reached, the system manager stops trying to restart the database. The default value is **600**, which means that the system manager will attempt to restart the database for a maximum of 600 seconds (10 minutes).
- **auto_restart_directory**: This setting specifies the base location of the **auto_restart** directory, which contains the denied_list, warned_list, and unanalyzed_list directories. The default value is `<install_path>/internal`.
- **ignore_deniedlist_queries**: This setting controls whether denied list queries are blocked from running or are allowed to be run when the database is returned to normal operation. The default value is **false**, which means denied list queries are not ignored and are therefore blocked from running. If ignore_deniedlist_queries is **true**, incoming queries are not compared with the denied list and are run.

> **Important**
> Changing the **auto_restart_max_attempts**, **auto_restart_time**, or **auto_restart_directory** values requires a restart of the system management daemon, **azgmgrd**, as well as the database. See Starting and Stopping AnzoGraph for instructions.

Removing a Query from the Block List

AnzoGraph stores metadata about the denied and warned list queries in the **stc_blocklist** system table. To remove a query from either list, you remove the entry from the stc_blocklist table by running the REMOVE_FROM_BLOCKIST command.

```
REMOVE_FROM_BLOCKLIST '<list_name>' <query_ID>
```

Where <list_name> is the name of the list that the query is on and <query_ID> is the ID number for the query. To retrieve the list name and query ID values, run the following query to return the stc_blocklist contents:

```
SELECT * WHERE { TABLE 'stc_blocklist'} ORDER BY ?blocklist
```

For example:

```
/opt/anzograph/bin/azgi -c "select * where {table 'stc_blocklist'} order by
?blocklist"
```

```
query | blocklist   | updated             | query_text                | part
------+-------------+---------------------+---------------------------+------
3587  | denied_list | 2020-08-25 14:29:27 | select * from <http://an..|    0
3592  | denied_list | 2020-08-25 14:29:32 | select * where {?s ?p ?o} |    0
3612  | warned_list | 2020-08-25 14:32:15 | select * from <http://an..|    0
```

In the results, the <list_name> is the value in the **blocklist** column, and <query_id> is the value in the **query** column. Running the following command removes the first entry from the stc_blocklist table, which removes that query from the denied list.

```
REMOVE_FROM_BLOCKLIST 'denied_list' 3587
```

Related Topics

Changing AnzoGraph Server Settings

AnzoGraph Settings Reference

Starting and Stopping AnzoGraph

Enabling Paged Data Mode (Preview)

By default, AnzoGraph is configured as an in-memory database. In memory mode, all graphs are stored in memory and all queries are run against the data in memory. Data is persisted to disk only for backup purposes as well as automatic loading of graphs back into memory when the database is restarted. You have the option, however, to configure AnzoGraph as a disk-based database, where all of the data is stored on disk and then paged into memory on-demand for running analytics.

> **Note**
> The Paged Data feature is available as a **Preview** release, which means the implementation has recently been completed but is not yet thoroughly tested and could be unstable. The feature is available for trial usage, but Cambridge Semantics recommends that you do not rely on Preview features in production environments.

How Does Paged Data Mode Work?

The procedure below gives an overview of how AnzoGraph operates in paged data mode:

1. First, just like in-memory mode, you load data into AnzoGraph before running queries.
2. As data is loaded, it passes through memory to be converted to AnzoGraph's internal storage format, and then it is saved to disk in the persistence directory. The persistence directory location is configurable, and the speed of the disk that hosts the directory has an impact on query performance. For the best performance, store the persistence directory on a fast disk, such as SSD.
3. AnzoGraph keeps the most recently accessed data cached in memory for queries. By default, the size of the cache is 20% of the total available memory. The percentage of memory to use for paged data caching is configurable. For more information, see paged_cache_memory_percent.
4. As queries are run, AnzoGraph keeps track of the data that is accessed most often and keeps that data cached in memory. If a query requests data that is not currently cached, AnzoGraph releases the least accessed data from memory and loads the relevant data into memory.

Enabling and Configuring Paged Data Mode

Follow the steps below to configure AnzoGraph for paged data storage. Before changing the configuration, make sure that your environment meets the requirements in Sizing Guidelines for Disk-Based Storage (Preview) in the Deployment Guide.

> **Important**
> Though enabling paged data does not change the way users interact with the database, i.e., data loading and query operations remain the same, the performance of user operations will likely be slower compared to the default in-memory operation. In addition, enabling paged data requires you to re-initialize the database to remove the existing persistence.

1. Stop the database. See Stop the Database and Leave the System Management Daemon Running for instructions.
2. **On the leader node**, open the AnzoGraph settings file, **settings.conf**, in a text editor. The file is in the `<install_path>/config` directory.
3. In settings.conf, locate the `# paged_data=false` line. This setting enables and disables paged data storage. Uncomment the line and change the value to **true** to enable paged data.

   ```
   paged_data=true
   ```

4. The following settings are also related to paged data operations. If necessary, uncomment the lines for any of these settings and modify the values as needed:
   - paged_cache_memory_percent: This setting controls the amount of memory (as a percentage of total memory) to use for caching the most often accessed data. The default value is **20**, which means AnzoGraph is configured to use 20% of the total available memory for caching data for analytics. If a query requests data that is not currently cached, AnzoGraph releases the least used data from memory and loads the relevant data into memory.

     > **Important**
     > Cambridge Semantics recommends that you do not set this value higher than 30.

   - enable_persistence: Persistence must be enabled when using paged data mode. This setting is **false** by default. See Using AnzoGraph Persistence (Preview) for information about enabling AnzoGraph persistence.
   - persistence_directory: The directory where AnzoGraph saves the data that is persisted to disk. By default, the data is saved in the `<install_path>/persistence` directory. To persist data to an alternate disk, such as a separate SSD, specify the path and directory name.
5. Save and close settings.conf.

6. Restart and re-initialize the database to apply the configuration change and remove any existing persisted data. See Reinitializing the Database for instructions. When AnzoGraph starts, reload the database from your original files or insert queries.

Related Topics

Changing AnzoGraph Server Settings

AnzoGraph Settings Reference

Generating Diagnostic Files with the System Manager

When Cambridge Semantics Support requests AnzoGraph diagnostic files for troubleshooting an issue, you can use the AnzoGraph system manager to generate the required system information. If you encounter an error and the database remains running, you run an XRAY command to produce the diagnostic files. If you encounter an error that crashes the database, you run a CRASHFETCH command to produce a "crashdump" that includes the diagnostic files. This section provides instructions for generating the diagnostic files using the AnzoGraph system manager. For instructions on retrieving diagnostic files from the Anzo Administration application, see Retrieving AnzoGraph Diagnostic Files.

- Generating an X-ray on a Running Database
- Generating a Crashdump after a Crash

Generating an X-ray on a Running Database

If you encounter an error and the database remains running, run the following command to take an x-ray from the command line on the AnzoGraph leader server. This command creates a tarball that includes the necessary diagnostic files:

```
<install_path>/bin/azgctl -xray /<path>/<name>.xray
```

- **path**: The location on the server where you want to save the tarball.
- **name**: The name for the tarball. The name must be unique; AnzoGraph will not overwrite existing files.
- **.xray**: All x-ray files must be named with the .xray extension.

For example, this command runs an x-ray on the leader server:

```
/opt/cambridgesemantics/anzograph/bin/azgctl -xray /tmp/query_error.xray
```

Generating a Crashdump after a Crash

If you encounter an issue that stops the database, AnzoGraph automatically generates diagnostic files for Support. Follow the instructions below to retrieve the files after a crash.

> **Note**
> The database does not need to be running to collect the crashdump.

1. Run the following command on the leader server to view a list of the available crash diagnostics.

   ```
   <install_path>/bin/azgctl -crashlist
   ```

   The results show a list of available crash dumps by timestamp. For example:

   ```
   Crash ID               Time
   ----------------------------------
   520460982     2021-06-28 20:30:35
   520457655     2021-06-28 20:28:25
   ```

2. Run the following command to retrieve the appropriate crash files. This command creates a tarball that includes the necessary files:

   ```
   <install_path>/bin/azgctl -crashfetch <crash_id> /<path>/<name>.xray
   ```

   - **crash_id**: The ID for the crash that you want to retrieve, as shown in the crashlist from the previous step. To automatically retrieve the latest crash files, omit the crash_id.
   - **path**: The location on the server where you want to save the tarball.
   - **name**: The name for the tarball. The name must be unique; AnzoGraph will not overwrite existing files.
   - **.xray**: All crashdumps files must be named with the .xray extension.

   For example, this command runs a crashfetch to capture the diagnostics with the ID 520457655:

   ```
   /opt/cambridgesemantics/anzograph/bin/azgctl -crashfetch 520457655
   /tmp/query_crash.xray
   ```

   This command captures the most recent crash diagnostic files:

   ```
   /opt/cambridgesemantics/anzograph/bin/azgctl -crashfetch /tmp/query_
   crash.xray
   ```

> **Tip**
>
> You can run the following command to remove all crash dumps from the server.
>
> ```
> <install_path>/bin/azgctl -crashtoss
> ```

Related Topics

Retrieving AnzoGraph Diagnostic Files

Anzo Admin CLI

The Anzo command line interface (CLI) utility, called **anzo**, is an advanced administration tool for managing Anzo. It is primarily used for migrations and deployments. The topics in this section provide information about the CLI.

> **Note**
> To script user interface operations or control Anzo with the CLI, please contact Cambridge Semantics.

Setting up the Admin CLI

> **Important**
> The anzo CLI is an advanced administration tool for managing Anzo. It is primarily used for migrations and deployments. To script user interface operations or control Anzo with the CLI, please contact Cambridge Semantics.

This topic provides instructions for configuring the admin command line interface, **anzo**, and viewing the help menu. The anzo client is in the `<install_path>/Client` directory.

- Adding the CLI to the Anzo Service User PATH
- Configuring the CLI
- Viewing the CLI Help Menu

Adding the CLI to the Anzo Service User PATH

Follow the instructions below to configure the PATH environment variable to include the Client directory so that you call the anzo CLI from anywhere.

1. If necessary, run the following command to become the Anzo service user:

   ```
   sudo su - <anzo_user_name>
   ```

   For example:

   ```
   sudo su - anzo
   ```

2. Open **~/.bash_profile** in a text editor.
3. Change the PATH to the following value:

   ```
   PATH=$PATH:$HOME/.local/bin:$HOME/bin:Anzo_install_path/Client
   ```

   For example:

   ```
   PATH=$PATH:$HOME/.local/bin:$HOME/bin:/opt/Anzo/Client
   ```

4. Save and close the file, and then run the following command:

   ```
   source ~/.bash_profile
   ```

5. Type **anzo** to verify that you can access the CLI. For example:

   ```
   [anzo@anzo-server ~]$ anzo
   Anzo Command Line Client.
   Copyright (c) 2017 - 2023 Cambridge Semantics Inc and others.
   All rights reserved.
   Version: 5.4.1.r202303240943
   Type anzo help for usage
   ```

Configuring the CLI

Follow the instructions below to configure a settings file that specifies the default Anzo CLI configuration values for parameters such as host, port, user, and password. Specifying these details in the settings file eliminates the need to include those options in subsequent commands.

To create and populate the settings file, **settings.trig**, in your home directory, run the following command:

```
anzo setup <options>
```

Where *options* include the following choices:

```
-beep , --beep                   beep when command is completed
-ds , --datasource <datasource>  URI of the datasource to query, if other than
primary datasource.
                                 Option not available for dataset queries.
-h , --host <hostname>           anzo server hostname
-http , --http                   Use http connection to server.
-p , --port <int>                anzo server port
-pause , --pause-exit            Wait for a user key entry before an abnormal
exit.
-ssl , --use-ssl                 Use SSL for connection.
-t , --timeout <timeout>         override the default 30 second timeout for
operations
-timer , --timer                 Print out the total operation time
-trace , --show-trace            Show stack trace for errors.
```

```
-trust , --trust-all             Trust all certificates including invalid ones
-u , --user <string>             username to connect with
-w , --password <string>         user's password
-x , --exclude-prefixes          Do not use prefixes defined in user settings to
expand options,
                                 arguments, or to write RDF.
-z , --settings <file>           override the default settings file location
```

For example:

```
anzo setup -h localhost -p 61616 -u sysadmin -w @nz0
```

Anzo creates the settings.trig file in the `~/user/.anzo` directory. You can edit the file as needed. The installation also includes a sample settings file, **settings_example.trig**, in the `Client` directory. You can view the sample file for reference. For example:

```
### standard prefixes
@prefix foaf      : <http://xmlns.com/foaf/0.1/> .
@prefix rdfs      : <http://www.w3.org/2000/01/rdf-schema#> .
@prefix dc        : <http://purl.org/dc/elements/1.1/> .
@prefix xsd       : <http://www.w3.org/2001/XMLSchema#> .
@prefix rdf       : <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
#### anzo prefixes:
@prefix cli : <http://openanzo.org/cli/> .
@prefix system : <http://openanzo.org/ontologies/2008/07/System#> .
@prefix anzo : <http://openanzo.org/ontologies/2008/07/Anzo#> .
@prefix ld : <http://cambridgesemantics.com/ontologies/2009/05/LinkedData#> .
@prefix anzowt :
<http://cambridgesemantics.com/ontologies/2009/05/AnzoWebToolkit#> .
@prefix reg : <http://cambridgesemantics.com/registries/> .
@prefix ontserv :
<http://cambridgesemantics.com/semanticServices/OntologyService#> .
@prefix ldserv : <http://cambridgesemantics.com/semanticServices/LinkedData#> .
cli:config {
  cli:config
#       system:user "" ;
#       system:password "" ;
  system:timeout  "0";
  system:useSsl "false";
  system:port "61616";
  system:keystoreFile "${ANZO_CLI_HOME}/../Common/ssl/client.ks";
  system:keystoreType "JCEKS";
  system:keystorePassword "p@ssw0rd";
  system:truststoreFile "${ANZO_CLI_HOME}/../Common/ssl/client.ts";
  system:truststoreType "JCEKS";
  system:truststorePassword "p@ssw0rd";
  .
}
```

Viewing the CLI Help Menu

The CLI help menu lists all of the available subcommands. To view the subcommands, run **anzo help**.

```
usage: anzo <subcommand> [options] [args]
Anzo Command Line Client.
Type 'anzo help <subcommand>' for help with a specific subcommand.
Available subcommands:
acls              Ensure the graphs in a dataset inherit their ACLs from the
dataset
analyze           Provides several flavors of analysis for Anzo request/response
logs
call              Calls an anzo semantic service and prints the service response
to the console
collapse          Collapse all URI arguments to prefixed URIs (CURIEs) using
user defined prefixes
collapseGraph     In specified graph(s), collapse object properties with only
one literal value
                  into a datatype property
convert           Converts between the various RDF file formats
count             Counts the statements in an RDF file
create            Creates named graphs in the repository from the provided RDF
csv               Export instances of an ontology class with all of their
property values
deploy            Import, export, or delete a linked data set and related
components
deregister       Deregister given resource from appropriate registries based on
rdf:type of resource
```

```
expand            Expands all prefixed URI (CURIE) arguments to expanded URIs
using
                  user defined prefix map
find              Retrieves statements from the server via simple pattern find
gen               Generates code for the ontologies as supplied by the input RDF
or arguments
get               Retrieves named graphs from the server
graph2lds         Creates a Linked Data Set from the statements in a graph(s)
import            Imports statements into the repository, creating graphs in the
repository as needed
inspectOntology   Inspects a dataset for an ontology
link              Link an excel workbook using a layout
load              Loads file based linked datasets
loadXML           Imports xml as statements into a graph in the repository as
needed
ls               List resources from appropriate registries based on type of resource
play              Play back a sequence of recorded requests
query             Executes a SPARQL query against the repository or a local RDF
file
rdfformats        Show available rdf formats
register          Register given resource to appropriate registries based on
rdf:type of resource.
                 Supported types:[
                       http://cambridgesemantics.com/ontologies/2009/05/LinkedData#Lin

http://cambridgesemantics.com/ontologies/2009/05/LinkedData#LinkedDataCollection

http://cambridgesemantics.com/ontologies/2009/05/LinkedData#LinkedDataCollectio
nInstance
                       http://www.w3.org/2002/07/owl#Ontology
                       http://openanzo.org/ontologies/2008/07/SemanticService#Semantic
                       http://cambridgesemantics.com/ontologies/2009/05/Spreadsheets#Li
                       http://cambridgesemantics.com/ontologies/2009/05/AnzoWebToolkit
                       http://cambridgesemantics.com/ontologies/Graphmarts#Graphmart
                       http://cambridgesemantics.com/ontologies/Graphmarts#Step
                       http://cambridgesemantics.com/ontologies/Graphmarts#Layer
                       http://cambridgesemantics.com/ontologies/Graphmarts#View
                       ]
remove            Removes named graphs from the repository
```

```
replace           Replaces named graphs in the repository with the provided RDF
reset             Resets the repository, replacing all contents of repository
with rdf provided
retrieve          Retrieves content from the binary store and saves it in a
local file
setup             Set up settings.trig file
sortedConvert     Converts between the various RDF file formats
store             Stores a local file in the Anzo server's binary store
union             Unions RDF from the arguments and optionally from STDIN as
well
update            Updates existing graphs in the repository
uploadBundle      Upload bundle to server
uploadCertificate Upload trusted certificate to server
watch             Listens for changes to a graph and prints them out
xray             Export system tables into trig file


URI arguments to commands may either be fully qualified URIs ("http://...") or
prefixed URIs ("dc:title").
The prefix mapping is defined in the users settings file.
User settings are loaded from a user's "~/.anzo/settings.trig" file.
See documentation for details.
```

To view the help for a specific subcommand, run **anzo help *command_name***. For example, the following command displays help for the find command:

```
[user@anzo Client]# ./anzo help find
usage: anzo find [options] [NAMED-GRAPH-URI...]
Retrieves statements from the server via simple pattern find.
-beep , --beep                     beep when command is completed
-ds , --datasource <datasource>    URI of the datasource to query, if other
than primary datasource.
                                   Option not available for dataset queries.
-f , --output-file <file>          write the find results to a file
-h , --host <hostname>             anzo server hostname
-http , --http                     Use http connection to server.
-lang , --literal-language <string> The literal language
-lit , --literal-object <string>   The literal object of find pattern
-n , --count                       Outputs only the total number of matching
```

```
statements
-o , --output-format <rdf-Format>   Override the default RDF format associated
with the RDF output(s)
-p , --port <int>                   anzo server port
-pause , --pause-exit               Wait for a user key entry before an abnormal
exit.
-pred , --predicate <URI>           The predicate of find pattern
-pretty , --pretty-print            PrettyPrint output (currently only json)
-ssl , --use-ssl                    Use SSL for connection.
-sub , --subject <subject>          The subject of find pattern
-t , --timeout <timeout>            override the default 30 second timeout for
operations
-timer , --timer                    Print out the total operation time
-trace , --show-trace               Show stack trace for errors.
-trust , --trust-all                Trust all certificates including invalid
ones
-type , --literal-datatype <URI>    The literal datatype
-u , --user <string>                username to connect with
-uri , --uri-object <URI>           The uri object of find pattern
-w , --password <string>            user's password
-x , --exclude-prefixes             Do not use prefixes defined in user settings
to expand options,
                                    arguments, or to write RDF.
-z , --settings <file>              override the default settings file location

'help rdfformats' for list of available RDF formats.
Filename arguments default to the file format matching their filename extension.
STDIN and STDOUT default to 'trig'.
```

Querying Graphmart Data

> **Important**
> The anzo CLI is an advanced administration tool for managing Anzo. It is primarily used for migrations and deployments. To script user interface operations or control Anzo with the CLI, please contact Cambridge Semantics.

This topic provides information about using the anzo CLI to query graphmart data in AnzoGraph.

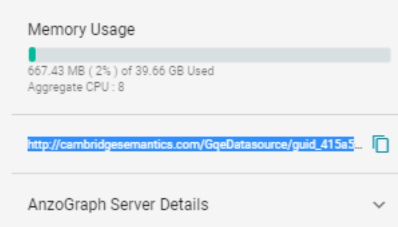
Use the **query** subcommand to access the data in graphmarts that are loaded in AnzoGraph:

```
anzo query "<query_text>" -ds <AZG_URI> -dataset <graphmart_URI>
```

If you saved the query in a file, run the following command to run the query in the file:

```
anzo query -f <filename>.rq -ds <AZG_URI> -dataset <graphmart_URI>
```

Where <filename>.rq is the path to and name of the query file and <AZG_URI> is the Datasource URI shown on the **Connections** > **AnzoGraph** screen in the Administration application. For example:



Memory Usage

667.43 MB (2%) of 39.66 GB Used
Aggregate CPU : 8

http://cambridgesemantics.com/GqeDatasource/guid_415a5...

AnzoGraph Server Details

And <graphmart_URI> is the URI for graphmart. How do I find the URI for a graphmart?

Examples

The example below queries a data set to list its classes:

```
anzo query "SELECT DISTINCT ?p WHERE { ?s ?p ?o.} LIMIT 100"
-ds http://cambridgesemantics.com/GqeDatasource/guid_
b833b32453694342c7bbc22422035e07
-dataset
http://cambridgesemantics.com/Graphmart/f4bc354ebe9540329eef561f66e42454
```

This example runs a query in a file:

```
anzo query -f /home/user/queries/classes.rq
-ds http://cambridgesemantics.com/GqeDatasource/guid_
b833b32453694342c7bbc22422035e07
-dataset
http://cambridgesemantics.com/Graphmart/f4bc354ebe9540329eef561f66e42454
```

Accessing a Graph's Metadata

> **Important**
> The anzo CLI is an advanced administration tool for managing Anzo. It is primarily used for migrations and deployments. To script user interface operations or control Anzo with the CLI, please contact Cambridge Semantics.

Each graph has a metadata graph associated with it. The metadata graph includes details such as ACL information, the last modified date, and which user created and modified the graph. To include the metadata graph when you retrieve graph details, use the **get** subcommand with the **-m** option:

```
anzo get -m <URI>
```

The **-m** option indicates that you want to see the metadata graph for the specified URI. For example, the following command retrieves the metadata graph for a graphmart:

```
anzo get -m
http://cambridgesemantics.com/Graphmart/89baf53cc5644600961778c88bd3d7fd
```

In addition to showing the graphmart details for the `<http://cambridgesemantics.com/Graphmart/89baf53cc5644600961778c88bd3d7fd>` graph, the results include the additional metadata for the graph:

```
...
<http://openanzo.org/metadataGraphs
(http%3A%2F%2Fcambridgesemantics.com%2FGraphmart%2F89baf53cc5644600961778c88bd3
d7fd)>
{
  <http://cambridgesemantics.com/Graphmart/89baf53cc5644600961778c88bd3d7fd> a
anzo:NamedGraph ;
    anzo:createdBy <http://openanzo.org/system/internal/sysadmin> ;
    anzo:lastModifiedByUser <http://openanzo.org/system/internal/sysadmin> ;
    anzo:created "2020-03-24T17:25:48.004Z"^^xsd:dateTime ;
    anzo:datasource datasource:systemDatasource ;
...
}
```

Specifying an Output Format

> **Important**
> The anzo CLI is an advanced administration tool for managing Anzo. It is primarily used for migrations and deployments. To script user interface operations or control Anzo with the CLI, please contact Cambridge Semantics.

The Anzo CLI enables you to request results in the following formats: TriG (default), RDF, RDFS, XML, NT, N3, TTL, TriX, and JSON. To change the format for results, you use the -o option with Anzo subcommands such as find, get, query, call, and analyze.

For example, the following get subcommand returns data set details in XML format:

```
anzo get -o xml
http://csi.com/FileBasedLinkedDataSet/059060234accd1d2d44b6bbb4207ee54
```

```
<?xml version="1.0" encoding="UTF-8"?>
<rdf:RDF
    xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:ld="http://cambridgesemantics.com/ontologies/2009/05/LinkedData#"
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:dc="http://purl.org/dc/elements/1.1/">
<rdf:Description
rdf:about="http://csi.com/DataLocation/059060234accd1d2d44b6bbb4207ee54">
<fileConnection xmlns="http://cambridgesemantics.com/ontologies/DataSources#"
    rdf:resource="http://cambridgesemantics.com/File_Connection/local"/>
<filePath xmlns="http://cambridgesemantics.com/ontologies/DataSources#">
    /nfs/data/store/LoadMovies_223d3/</filePath>
<isPrimary xmlns="http://cambridgesemantics.com/ontologies/DataSources#"
    rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean">true</isPrimary>
<rdf:type
rdf:resource="http://cambridgesemantics.com/ontologies/DataSources#DataLocatio
n"/>
<rdf:type
rdf:resource="http://cambridgesemantics.com/ontologies/DataSources#PathConnecti
on"/>
</rdf:Description>
```

Troubleshooting

The topics in this section provide troubleshooting information for Anzo components.

Getting Information from the Anzo Log Files

You can review the Anzo log files to get more detailed information about errors or to obtain more granular information about server operations. The server writes logs to the `<install_path>/Server/logs` directory and adds timestamps to all logged statements. Major issues are logged in files with the suffix "error," and other server information is logged in files with the suffix "info." For information about viewing and managing Anzo logs, see Managing Anzo Logging.

Related Topics

Viewing the Current Stack in a Browser

Error Message Reference

Viewing the Current Stack in a Browser

When the System Monitor service is configured to save heap and/or stack dumps, those dumps are saved to disk and cannot be viewed from the Administration application. However, the sysadmin user can quickly review the stack for the current state of the JVM in a browser. Follow the instructions below to view the stack.

> **Note**
> Only a user with sysadmin access can view the stack in a browser. The sysadmin credentials are required to log in to the stack page.

To review the stack for the current state, go to the following URL in a browser:

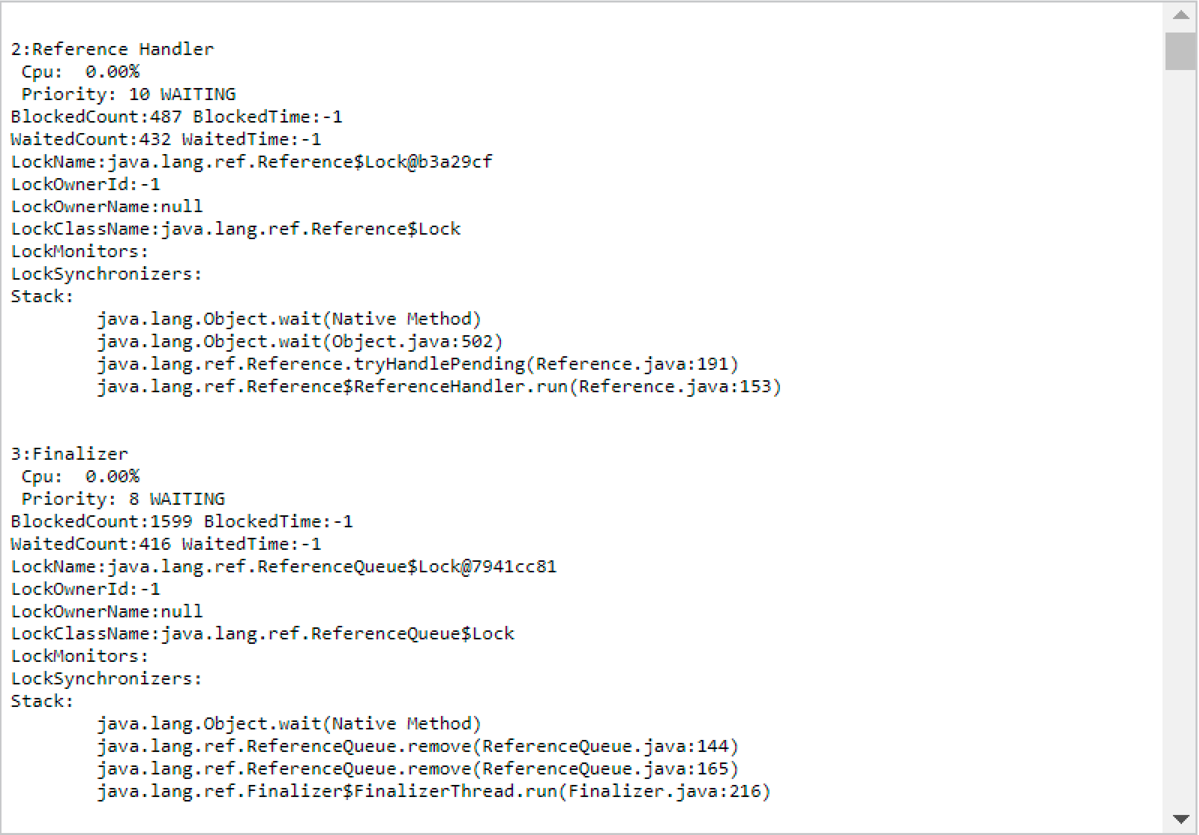
```
https://<Anzo_server>:<HTTPS_admin_port>/status?stack
```

Where <Anzo_server> is the IP address or host name for the Anzo server and <HTTPS_admin_port> Is the HTTPS port for the Administration application. For example:

```
https://10.11.0.12:8946/status?stack
```

The browser prompts you to log in as the **sysadmin** user. Supply the credentials and click **Sign in**.

The current state is displayed. For example:



```
2:Reference Handler
 Cpu:  0.00%
 Priority: 10 WAITING
BlockedCount:487 BlockedTime:-1
WaitedCount:432 WaitedTime:-1
LockName:java.lang.ref.Reference$Lock@b3a29cf
LockOwnerId:-1
LockOwnerName:null
LockClassName:java.lang.ref.Reference$Lock
LockMonitors:
LockSynchronizers:
Stack:
        java.lang.Object.wait(Native Method)
        java.lang.Object.wait(Object.java:502)
        java.lang.ref.Reference.tryHandlePending(Reference.java:191)
        java.lang.ref.Reference$ReferenceHandler.run(Reference.java:153)


3:Finalizer
 Cpu:  0.00%
 Priority: 8 WAITING
BlockedCount:1599 BlockedTime:-1
WaitedCount:416 WaitedTime:-1
LockName:java.lang.ref.ReferenceQueue$Lock@7941cc81
LockOwnerId:-1
LockOwnerName:null
LockClassName:java.lang.ref.ReferenceQueue$Lock
LockMonitors:
LockSynchronizers:
Stack:
        java.lang.Object.wait(Native Method)
        java.lang.ref.ReferenceQueue.remove(ReferenceQueue.java:144)
        java.lang.ref.ReferenceQueue.remove(ReferenceQueue.java:165)
        java.lang.ref.Finalizer$FinalizerThread.run(Finalizer.java:216)
```

You can also check specifically for blocked or deadlocked threads by replacing **stack** in the URL with **block** or **deadlock**. To check for blocked threads, go to the following URL:

```
https://<Anzo_server>:<HTTPS_admin_port>/status?block
```

For example:

```
https://10.11.0.12:8946/status?block
```

To check for deadlocks, go to the URL below:

```
https://<Anzo_server>:<HTTPS_admin_port>/status?deadlock
```

For example:

```
https://10.11.0.12:8946/status?deadlock
```

Related Topics

Managing Anzo Logging

Enabling and Configuring the System Monitor Service

Error Message Reference

This topic provides information about Anzo and AnzoGraph and error messages.

- Anzo Error Messages
- AnzoGraph Error Messages

Anzo Error Messages

This section includes the possible causes and solutions for Anzo error messages. Click a message in the list below to view details about that error:

- Application Service Failure
- Elasticsearch exception [type=circuit_breaking_exception, reason=[parent] Data too large, data for [<http_request>]...
- Sparkler Exception: java.io.IOException: Unable to connect to provided ports 10000~10010

Application Service Failure

This message indicates that the Anzo server cannot bind to the Application Port defined on the Server Settings page in the Administration application. The problem has two likely causes:

- Another program is bound to the defined Anzo Server Application Port.
- You are not running as the root user and lack the required permission.

To resolve this issue, make sure that no other application is running on the defined Application port and log in as the root user if Anzo is installed on a UNIX operating system.

Elasticsearch exception [type=circuit_breaking_exception, reason=[parent] Data too large, data for [<http_request>]...

This message indicates that the Elasticsearch heap size is not large enough to process the request. By default, Elasticsearch is configured to use a maximum heap size of 1 GB. Cambridge Semantics recommends that you increase the amount to 50% of the memory that is available on the server. To change the configuration, open the `<elasticsearch_install_dir>/config/jvm.options` file in an editor. At the top of the file, modify the **Xms** and **Xmx** values to replace the **1** with the new value. For example:

```
# Xms represents the initial size of total heap space
# Xmx represents the maximum size of total heap space

-Xms15g
-Xmx15g
```

Sparkler Exception: java.io.IOException: Unable to connect to provided ports 10000~10010

This message indicates that the Sparkler Livy RSC client ran out of the ports that it uses internally for running jobs. Increase the range of ports by adjusting the **livy.rsc.launcher.port.range** value in the **livy-client.conf** file. If you use the embedded Anzo Sparkler compiler, the file is in the `<install_path>/Server/spark/csi-livy-spark/conf` directory.

Cambridge Semantics recommends that you set **livy.rsc.launcher.port.range = 10000~10110**. Restart the Livy server after changing the configuration file.

AnzoGraph Error Messages

This section includes the possible causes and solutions for AnzoGraph error messages. Click a message in the list below to view details about that error:

- Exiting: Error - Cannot execute as user 'root'. To override this security protection, set 'enable_root_user=true': Invalid user id
- Invalid Certificate
- "Compilation Failed" at Startup
- Fatal Error. Caught Signal 15

Exiting: Error - Cannot execute as user 'root'. To override this security protection, set 'enable_root_user=true': Invalid user id

This message indicates that you tried to start AnzoGraph as the root user and root access is disabled. Log in as the correct user, and then run the command again.

Invalid Certificate

This message indicates that you replaced the default AnzoGraph certificates with your own trusted certificates and the certificates are invalid. Certificates can be invalid because they expired or they were generated or signed incorrectly. For information about replacing certificates, see Replace the Default Self-Signed Certificates with Trusted Certificates in the Deployment Guide.

"Compilation Failed" at Startup

If AnzoGraph fails to start and you receive a "Compilation failed" message, it may indicate that some of the required GNU Compiler Collection (GCC) libraries are missing. Specifically, AnzoGraph requires the **glibc**, **glibc-devel**, and **gcc-c++** libraries. Typically when you install GCC by running `yum install gcc` those libraries are included as part of the package. In some cases, depending on the host server configuration, installing GCC excludes certain libraries. To install the missing libraries, run the following command:

```
sudo yum install glibc glibc-devel gcc-c++
```

Then start AnzoGraph again.

Fatal Error. Caught Signal 15

This error indicates that a process external to AnzoGraph stopped the AnzoGraph processes, such as if the host machine was shut down while AnzoGraph was running. Restart AnzoGraph to proceed with normal usage.