



Informatica® Metadata Command Center
November 2025

Microsoft Azure SQL Server Script Sources

© Copyright Informatica LLC 2023, 2025

This software and documentation are provided only under a separate license agreement containing restrictions on use and disclosure. No part of this document may be reproduced or transmitted in any form, by any means (electronic, photocopying, recording or otherwise) without prior consent of Informatica LLC.

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation is subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License.

Informatica, Informatica Cloud, Informatica Intelligent Cloud Services, PowerCenter, PowerExchange, and the Informatica logo are trademarks or registered trademarks of Informatica LLC in the United States and many jurisdictions throughout the world. A current list of Informatica trademarks is available on the web at <https://www.informatica.com/trademarks.html>. Other company and product names may be trade names or trademarks of their respective owners.

Portions of this software and/or documentation are subject to copyright held by third parties. Required third party notices are included with the product.

The information in this documentation is subject to change without notice. If you find any problems in this documentation, report them to us at infa_documentation@informatica.com.

Informatica products are warranted according to the terms and conditions of the agreements under which they are provided. INFORMATICA PROVIDES THE INFORMATION IN THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT.

Publication Date: 2025-11-20

Table of Contents

- Preface 4**
- Chapter 1: Introduction to Microsoft Azure SQL Script catalog sources..... 5**
 - Extraction and view process. 6
 - About the Microsoft Azure SQL Server Script catalog source. 6
 - Extracted metadata 7
- Chapter 2: Before you begin..... 8**
 - Verify permissions 8
 - Create a connection 9
- Chapter 3: Create a catalog source in Metadata Command Center 14**
 - Step 1. Register a catalog source 14
 - Step 2. Configure capabilities 16
 - Configure metadata extraction. 16
 - Configure lineage discovery. 18
 - Step 3. Associate stakeholders and asset groups. 19
 - Step 4. Run or schedule the job. 21
 - Step 5. Assign reference catalog source connections to endpoint catalog source objects. 22
- Chapter 4: View results in Data Governance and Catalog..... 24**
 - View metadata extraction results 24
 - View data lineage. 26
 - View lineage at the catalog source level. 27
 - View lineage at the data set level. 27
 - View lineage at the data element level. 28

Preface

Read *Microsoft Azure SQL Server Script Sources* to learn how to register and configure Microsoft Azure SQL Server Script sources in Metadata Command Center as catalog sources. After you configure a catalog source, you extract metadata and then view the results in Data Governance and Catalog.

CHAPTER 1

Introduction to Microsoft Azure SQL Script catalog sources

You can use Metadata Command Center to extract metadata from a source system.

A source system is any system that contains data or metadata. For example, Microsoft SQL Server is a source system from which you can extract metadata through a Microsoft Azure SQL Server Script catalog source with Metadata Command Center. A catalog source is an object that represents and contains metadata from the source system.

Before you extract metadata from a source system, you first create and register a catalog source that represents the source system. Then you configure capabilities for the catalog source. A capability is a task that Metadata Command Center can perform, such as metadata extraction, lineage discovery, data profiling, data classification, or glossary association.

When Metadata Command Center extracts metadata, Data Governance and Catalog displays the extracted metadata and its attributes as technical assets. You can then perform tasks such as analyzing the assets, viewing lineage, and creating links between those assets and their business context.

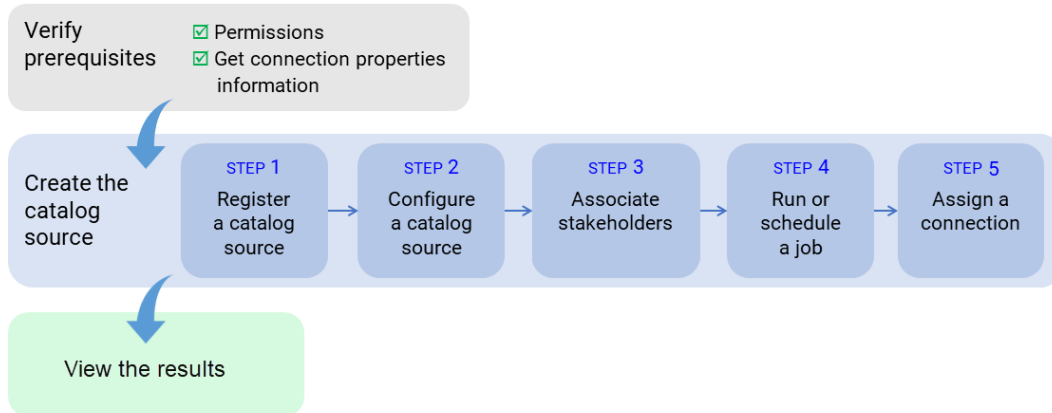
The following table describes the capabilities of the catalog source:

Capability	Description
Lineage Discovery	Builds the complete lineage of a catalog source by recommending endpoint catalog source objects to assign to reference catalog source connections. When you run the catalog source job, Metadata Command Center assigns the reference catalog source connections to CLAIRE recommended endpoint catalog source objects. You can then view the list of CLAIRE recommendations and accept or reject them.

Extraction and view process

To extract metadata from a source system, configure the catalog source and run the catalog source job in Metadata Command Center. Then view the results in Data Governance and Catalog.

The following image shows the process to extract metadata from a script:



After you verify prerequisites, perform the following tasks to extract metadata from Microsoft Azure SQL Server Script:

1. Register a catalog source. Create a catalog source object, select Microsoft Azure SQL Server Script, and specify values for connection properties.
2. Configure the catalog source. Specify the runtime environment and configure parameters for metadata extraction. Optionally, add filters to include or exclude source system assets from metadata extraction. You can also configure other capabilities such as data profiling and quality, data classification, or glossary association.
3. Optionally, associate stakeholders. Associate users with technical assets, giving the users permission to perform actions determined by their roles.
4. Run or schedule the catalog source job.
5. Optionally, assign a connection to referenced source system assets.

About the Microsoft Azure SQL Server Script catalog source

You can use the Microsoft Azure SQL Server Script catalog source to extract metadata from script files that define transformations on a Microsoft SQL Server source system.

Microsoft Azure SQL Server Script is a set of Microsoft Azure SQL statements stored in files that you can use to run sequential scripts.

Compatible connectors

Before you configure the Microsoft Azure SQL Server Script catalog source, you must connect to the Microsoft SQL Server source system.

Use the SQL Server connector to connect to the Microsoft SQL Server source system. For information about configuring a connection, see *Administration*.

Extracted metadata

You can use the Microsoft Azure SQL Server Script catalog source to extract metadata from scripts.

Metadata Command Center extracts the following metadata from the Microsoft Azure SQL Server Script source system:

- Calculation
- Folder
- Script
- Statements

CHAPTER 2

Before you begin

Before you can extract catalog source metadata, complete prerequisite tasks.

Perform the following prerequisite tasks:

- Copy the Microsoft Azure SQL Server Script files from which you want to extract metadata to the machine where the Secure Agent is installed. When you configure the Microsoft Azure SQL Server Script catalog source, you provide the absolute path to the Microsoft Azure SQL Server Script files for metadata extraction.
- Verify permissions to access the Microsoft Azure SQL Server Script catalog source and the Microsoft SQL Server source system.
- Create a connection.

Verify permissions

To extract Microsoft Azure SQL Server Script metadata, you need account access and permissions to the Microsoft Azure SQL Server Script catalog source and the Microsoft SQL Server source system.

Verify that the administrator has the following permissions:

- Read permission to access the folder containing scripts.
- Permissions to configure the Microsoft SQL Server connection:
 - select on sys.all_columns
 - select on sys.all_objects
 - select on sys.all_parameters
 - select on sys.database_principals
 - select on sys.databases
 - select on sys.foreign_key_columns
 - select on sys.indexes
 - select on sys.index_columns
 - select on sys.partitions
 - select on sys.schemas
 - select on sys.sql_modules
 - select on sys.synonyms
 - select on sys.types

- select on sys.tables
- select on sys.table_types

Create a connection

Before you configure the Microsoft Azure SQL Server Script catalog source, create a connection object in Administrator.

Ensure that you have the database source information to connect to the Microsoft SQL Server.

1. In Administrator, select **Connections**.
2. Click **New Connection**.
3. Enter the following connection details specific to Microsoft SQL Server connection:

Property	Description
Connection Name	Name of the connection. Each connection name must be unique within the organization. Connection names can contain alphanumeric characters, spaces, and the following special characters: _ . + -, Maximum length is 255 characters.
Description	Description of the connection. Maximum length is 4000 characters.
Use Secret Vault	Stores sensitive credentials for this connection in the secrets manager that is configured for your organization. This property appears only if secrets manager is set up for your organization. When you enable the secret vault in the connection, you can select which credentials that the Secure Agent retrieves from the secrets manager. If you don't enable this option, the credentials are stored in the repository or on a local Secure Agent, depending on how your organization is configured. Note: If you're using this connection to apply data access policies through pushdown or proxy services, you cannot use the Secret Vault configuration option. For information about how to configure and use a secrets manager, see "Secrets manager configuration" in the Administrator help.
Runtime Environment	The name of the runtime environment where you want to run the tasks. A runtime environment is either Informatica Cloud Secure Agent or a serverless runtime environment. For more information about how to configure and use the runtime environments, see <i>Runtime Environments</i> in the Administrator help.
SQL Server Version	This property is no longer used. If you select a version, it is ignored.

4. You can configure one of the following authentication modes to connect to Microsoft SQL Server databases:
 - SQL Server authentication

- Windows authentication (Deprecated)
 - Active Directory Password authentication
 - Windows Authentication V2
 - Kerberos authentication
5. Select the required authentication type and then configure the authentication-specific parameters. Default is SQL Server Authentication.

The following table describes the basic connection properties for SQL Server authentication:

Property	Description
User Name	User name for the database login. The user name can't contain a semicolon. To connect to Microsoft Azure SQL Database, specify the user name in the following format: <code>username@host</code>
Password	Password for the database login. The password can't contain a semicolon.
Host	Name of the machine hosting the database server. To connect to Microsoft Azure SQL Database, specify the fully qualified host name. For example, <code>vmjcmwxsfbheng.westus.cloudapp.azure.com</code> .
Port	Network port number used to connect to the database server. Default is 1433.
Instance Name	Instance name of the Microsoft SQL Server database.
Database Name	Database name for the Microsoft SQL Server target connection. Database name is case-sensitive if the database is case-sensitive. Maximum length is 100 characters. Database names can include alphanumeric and underscore characters.
Schema	The schema name to select tables during object selection in a mapping. When you select objects in a mapping, all schemas you have access to are displayed, and you can select tables from the available schemas.
Code Page	The code page of the database server.
Domain	This property is not applicable for SQL Server authentication.

The following table describes the basic connection properties for Windows authentication:

Property	Description
Host	Name of the machine hosting the database server. To connect to Microsoft Azure SQL Database, specify the fully qualified host name. For example, <code>vmjcmwxsfbheng.westus.cloudapp.azure.com</code> .
Port	Network port number used to connect to the database server. Default is 1433.

Property	Description
Instance Name	Instance name of the Microsoft SQL Server database.
Database Name	Database name for the Microsoft SQL Server target connection. Database name is case-sensitive if the database is case-sensitive. Maximum length is 100 characters. Database names can include alphanumeric and underscore characters.
Schema	The schema name to select tables during object selection in a mapping. When you select objects in a mapping, all schemas you have access to are displayed, and you can select tables from the available schemas.
Code Page	The code page of the database server.
Domain	This property is not applicable for Windows authentication.

The following table describes the basic connection properties for Active Directory Password authentication:

Property	Description
User Name	User name for the database login. The user name can't contain a semicolon. To connect to Microsoft Azure SQL Database, specify the user name in the following format: <code>username@host</code>
Password	Password for the database login. The password can't contain a semicolon.
Host	Name of the machine hosting the database server. To connect to Microsoft Azure SQL Database, specify the fully qualified host name. For example, <code>vmjcmwxsfbheng.westus.cloudapp.azure.com</code> .
Port	Network port number used to connect to the database server. Default is 1433.
Instance Name	Instance name of the Microsoft SQL Server database.
Database Name	Database name for the Microsoft SQL Server target connection. Database name is case-sensitive if the database is case-sensitive. Maximum length is 100 characters. Database names can include alphanumeric and underscore characters.
Schema	The schema name to select tables during object selection in a mapping. When you select objects in a mapping, all schemas you have access to are displayed, and you can select tables from the available schemas.

Property	Description
Code Page	The code page of the database server.
Domain	This property is not applicable for Active Directory Password authentication.

The following table describes the basic connection properties for Windows Authentication V2:

Property	Description
User Name	<p>User name for the database login. The user name can't contain a semicolon.</p> <p>To connect to Microsoft Azure SQL Database, specify the user name in the following format: username@host</p> <p>If you use Windows Authentication v2 on Windows, the user name is used as follows:</p> <ul style="list-style-type: none"> - During design time, the agent uses the user name specified here to test the connection. - During runtime, the Microsoft SQL server driver ignores the user name specified in this field and uses the credentials of the user who started the Secure Agent service. <p>If you use Windows Authentication v2 on Linux, the user name specified here is used both during design time and runtime.</p>
Password	<p>Password for the database login. The password can't contain a semicolon.</p> <p>If you use Windows Authentication v2 on Windows, the password is used as follows:</p> <ul style="list-style-type: none"> - During design time, the agent uses the password specified here to test the connection. - During runtime, the Microsoft SQL server driver ignores the password specified in this field and uses the credentials of the user who started the Secure Agent service. <p>If you use Windows Authentication v2 on Linux, the password specified here is used both during design time and runtime.</p>
Host	<p>Name of the machine hosting the database server.</p> <p>To connect to Microsoft Azure SQL Database, specify the fully qualified host name.</p> <p>For example, vmjcmwxsfbheng.westus.cloudapp.azure.com.</p>
Port	<p>Network port number used to connect to the database server.</p> <p>Default is 1433.</p>
Instance Name	Instance name of the Microsoft SQL Server database.
Database Name	<p>Database name for the Microsoft SQL Server target connection.</p> <p>Database name is case-sensitive if the database is case-sensitive. Maximum length is 100 characters.</p> <p>Database names can include alphanumeric and underscore characters.</p>
Schema	<p>The schema name to select tables during object selection in a mapping.</p> <p>When you select objects in a mapping, all schemas you have access to are displayed, and you can select tables from the available schemas.</p>
Code Page	The code page of the database server.
Domain	The domain name of the Windows user.

The following table describes the basic connection properties for Kerberos authentication:

Property	Description
Host	Name of the machine hosting the database server. To connect to Microsoft Azure SQL Database, specify the fully qualified host name. For example, <code>vmjcmwxsfbheng.westus.cloudapp.azure.com</code> .
Port	Network port number used to connect to the database server. Default is 1433.
Instance Name	Instance name of the Microsoft SQL Server database.
Database Name	Database name for the Microsoft SQL Server target connection. Database name is case-sensitive if the database is case-sensitive. Maximum length is 100 characters. Database names can include alphanumeric and underscore characters.
Schema	The schema name to select tables during object selection in a mapping. When you select objects in a mapping, all schemas you have access to are displayed, and you can select tables from the available schemas.
Code Page	The code page of the database server.
Domain	This property is not applicable for Kerberos authentication.

6. Click **Test Connection**.

CHAPTER 3

Create a catalog source in Metadata Command Center

Use Metadata Command Center to configure a catalog source for Microsoft Azure SQL Server Script and extract metadata.

When you configure a catalog source, you define the source system that you want to extract metadata from. Configure filters to include or exclude source system metadata before you run the job. Optionally, configure other capabilities, such as lineage discovery, data profiling and quality, data classification, relationship discovery, and glossary association.

To provide stakeholders access to technical assets, you can assign access through stakeholder roles. You can also associate technical assets extracted from the catalog source to asset groups. If your catalog source references other source systems, you can create a connection assignment to the endpoint catalog source to view complete lineage.

Step 1. Register a catalog source

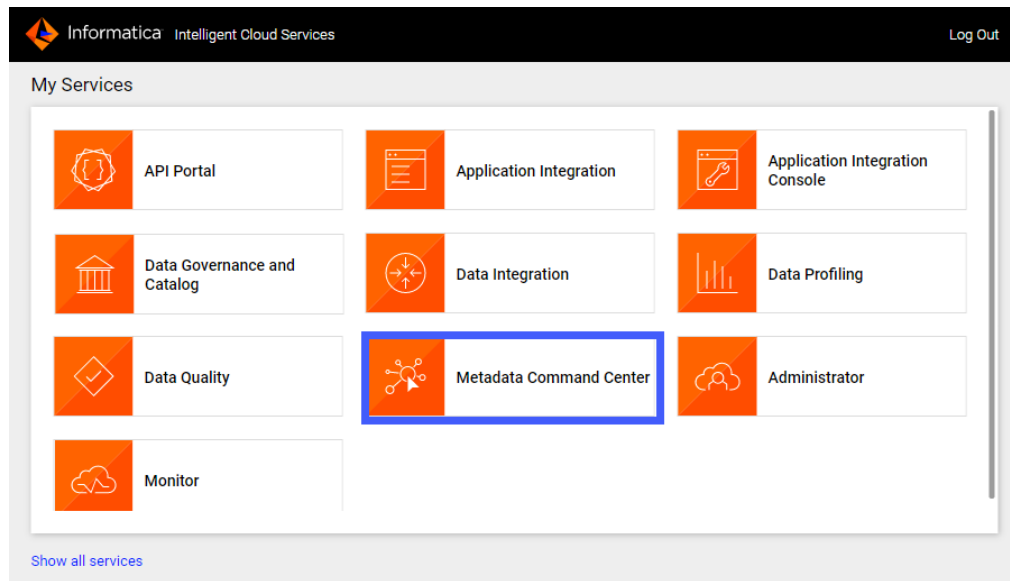
When you register a catalog source, provide general information and connection values.

1. Log in to Informatica Intelligent Cloud Services.

The **My Services** page appears.

2. Click **Metadata Command Center**.

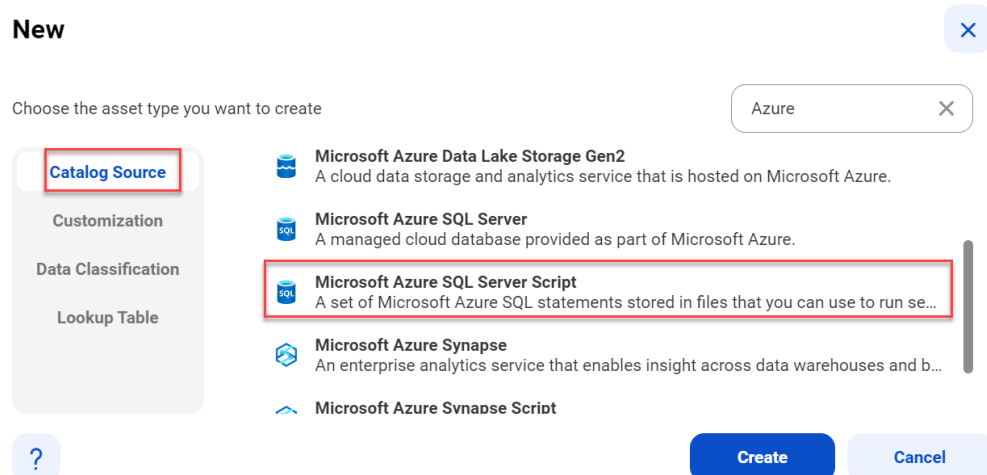
The following image shows the Metadata Command Center box on the **My Services** page:



The Metadata Command Center home page appears.

3. Click **New**.
4. Select **Catalog Source** from the list of asset types.
5. Select **Microsoft Azure SQL Server Script** from the list of catalog source types.
6. Click **Create**.

The following image shows where you choose the catalog source:



The **New Catalog Source** page opens.

7. In the **General Information** section, enter a name and an optional description for the catalog source.

Note: You can rename a catalog source after you create it, but to apply the change to all associated objects you must rerun the metadata extraction job.

After you save the catalog source, you can update the description in Metadata Command Center and Data Governance and Catalog. The update appears only in the service in which you update it.

8. In the **Connection Information** area, select the connection that you created in Administrator.

Note: To create or edit a catalog source, you need permissions on the connection to the source system. Select a connection that you have access to, or ask the administrator to grant the necessary permissions to the connection that you want to use.

9. Click **Connection Properties** to expand and view the connection properties for the selected connection.
10. Click **Test Connection** to test your connection to the source system.
11. Click **Next**.

The **Configuration** page appears.

Step 2. Configure capabilities

When you configure the Microsoft Azure SQL Server Script catalog source, you define the settings for the metadata extraction capability.

The metadata extraction capability extracts source metadata from external source systems. You can also configure other capabilities that the catalog source includes.

You can save the catalog source configuration at any point after you enter the connection information. After you save the catalog source, you can choose to run the catalog source job. To run the job once, click **Run**. To run metadata extraction and other capabilities on a recurring schedule, configure schedules on the **Schedule** tab.

Configure metadata extraction

When you configure the Microsoft Azure SQL Server Script catalog source, you choose a runtime environment, define filters, and enter configuration parameters for metadata extraction.

1. In the **Connection and Runtime** area, choose a serverless runtime environment or the Secure Agent group where you want to run catalog source jobs.

Note: Serverless runtime environment options are available if the catalog source works with a serverless runtime environment.

2. Choose to retain, delete, or deprecate objects that are deleted from the source system in the catalog with the **Metadata Change Option**.
 - **Retain.** Retains objects that are deleted from the source system in the catalog. If you update or add a filter, the catalog retains objects extracted from the previous job and extracts additional objects that match the current filter. Objects deleted from the source system are not deleted from the catalog. Enrichments added on deleted objects and relationships are retained.
 - **Delete.** Deletes metadata from the catalog based on objects deleted from the source system and changes you make to the filter. Enrichments added on deleted objects and relationships are also permanently lost. Objects renamed in the source system are removed and recreated in the catalog.
 - **Deprecate.** The lifecycle of objects imported into the catalog moves to Obsolete based on objects deleted from the source system and changes you make to the filter. This does not impact enrichments added on deprecated objects and relationships. Objects renamed in the source system are removed and recreated in the catalog. When you run the catalog source job again for other capabilities such as data classification, relationship discovery, or glossary association, the job doesn't consider obsolete objects. Obsolete objects remain in the catalog until they are purged when you run a **Purge Obsolete Objects** job on the **Explore** page.

Note: You can also change the configured metadata change option when you run a catalog source.

3. In the **Filters** area, define one or more filter conditions to extract metadata:
 - a. Select **Yes** to view filter options.
 - b. From the Include/Exclude list, choose to include or exclude metadata based on the filter parameters.
 - c. From the object type list, select **Script Path**.
 - d. Enter the script path as the filter value.

Filter values can contain wildcards. Use the following rules when you enter filter values with wildcards:

- Use a question mark to represent a single character.
- Use an asterisk to represent multiple characters.
- If an object contains an asterisk or a question mark, enclose the symbol in double quotes.
- If a filter value contains spaces before or after the string value, enclose the value in double quotes.
- Don't use wildcards in file paths. To enter a path hierarchy, use separators that the source system allows, such as a period or a slash.

The following image shows the filter condition:

Filters

Specify metadata filters: ☐ No ☒ Yes

[Show supported wildcards and examples](#)

Include or exclude m...

Select the object type

Enter a value to specify the object location

If the scripts root directory path is `/users/opt/input`, use the following examples to create filter conditions:

- To include or exclude metadata from the script named `script1.sql` located in the path `/users/opt/input/folder1/`, enter: `folder1/script1.sql`
- To include or exclude metadata from all scripts with SQL extension stored in the path `/users/opt/input/folder1/`, enter: `folder1/*.sql`
- To include or exclude metadata from all scripts stored in the path `/users/opt/input/`, enter: `*`
- To include or exclude metadata from all scripts with SQL extension and names that start with 'script' followed by a single character, stored in the path `/users/opt/input/folder1/`, enter: `folder1/script?.sql`

- e. Optionally, to define an additional filter with an OR condition, click the **Add** icon.

4. In the **Configuration Parameters** area, enter configuration parameters.

The following table describes the properties that you can enter:

Property	Description
Scripts Root Directory Path	Path to the remote SQL script root directory.
Default Database	Default database for the SQL script processing.

Property	Description
Default Schema	Default schema for the SQL script processing.
MetaTables Include Filter	<p>Advanced parameter. When you process PL/SQL statements, Metadata Command Center does not read tables or view content by default. If you want to use the content, for example, to process dynamic SQL statements, use the MetaTables Include Filter parameter. This parameter prompts the database for the required metadata. Verify that the user has SELECT permissions for metatables.</p> <p>Note: This parameter appears when you click Show Advanced.</p> <p>Note: Don't use this option to specify filters for tables that you want to include or exclude during the metadata extraction run.</p>
Additional Settings	<p>Configure expert parameters to specify additional configuration options to be passed at runtime. Required if you need to troubleshoot the catalog source job.</p> <p>Caution: Use expert parameters when it is recommended by Informatica Global Customer Support.</p> <p>Note: This parameter appears when you click Show Advanced.</p>

5. Configure additional capabilities for the catalog source by clicking on the tabs.

Configure lineage discovery

Enable the lineage discovery capability and use CLAIRE to build complete lineage by recommending endpoint catalog source objects to assign to reference catalog source connections.

1. Click the **Lineage Discovery** tab.
2. Select **Enable Lineage Discovery**.
3. In the **Filters** area, define one or more filter conditions to apply for lineage discovery.

To define filters, you can choose to select catalog source types, asset groups, or enter a catalog source name or search from a list of catalog sources.

- a. Select **Yes** to view filter options.
- b. From the Include/Exclude list, choose to include or exclude catalog sources for lineage discovery based on the filter parameters.
- c. From the filter type list, select catalog source type, catalog source name, or asset group.
- d. In the filter value field, select the required catalog source types, or click the Search button and select catalog sources or asset groups.

Filters can contain the asterisk wildcard to represent multiple characters or empty text.

The following image shows the filter condition options:

Enable Lineage Discovery: ☒

Filters

Specify lineage discovery filters: ☐ No ☒ Yes

> Show supported wildcards and examples

Include	Catalog Source Type	Select Catalog Source Types	+	🗑️
Exclude	Catalog Source Name	Select Catalog Sources	+	🗑️
Exclude	Asset Group	Select Asset Groups	+	🗑️

Examples:

- To include or exclude all Oracle catalog sources, select **Catalog Source Type** as the filter type and select `Oracle` in the filter value field.
- To include or exclude the 'Oracle_Retail' catalog source, select **Catalog Source Name** as the filter type and search for the catalog source or enter `Oracle_Retail` in the filter value field.
- To include or exclude all catalog sources with names that start with 'Oracle', select **Catalog Source Name** as the filter type and search for the catalog source or enter `Oracle*` in the filter value field.
- To include or exclude all catalog sources with names that end with 'Retail', select **Catalog Source Name** as the filter type and search for the catalog source or enter `*Retail` in the filter value field.
- To include or exclude all catalog sources with names that contain 'Ret', select **Catalog Source Name** as the filter type and search for the catalog source or enter `*Ret*` in the filter value field.
- To include or exclude all catalog sources that are part of the 'Financial Group' asset group, select **Asset Group** as the filter type and search `Financial Group` in the filter value field.

Note: You can't add more than one include or exclude filter for the same filter type.

- e. Optionally, to define an additional filter with an AND condition, click the **Add** icon.

For more information about lineage discovery, see *Lineage discovery* in the *Administration* help.

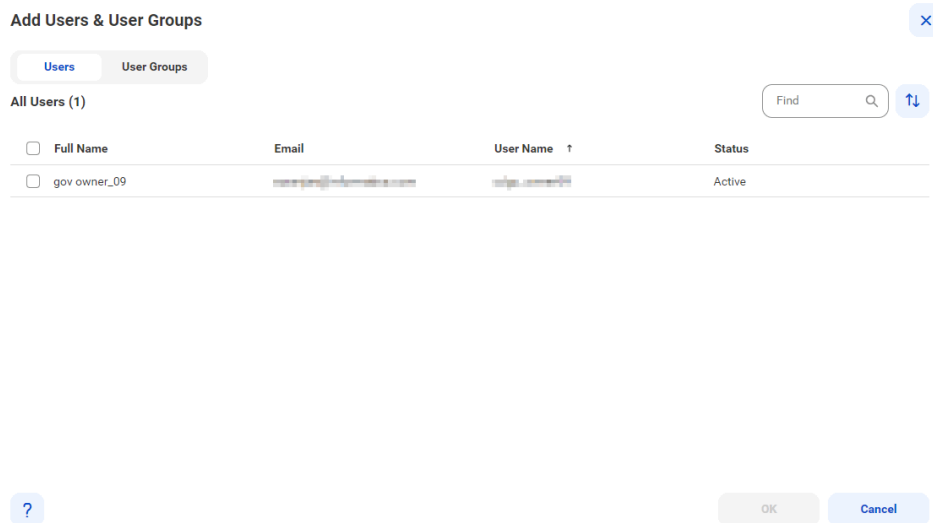
Step 3. Associate stakeholders and asset groups

Associate users or user groups within a stakeholder role as stakeholders for technical assets in Data Governance and Catalog. Also, you can choose to assign technical assets extracted from the catalog source to asset groups. You can then use access policies to control permissions on assets that are assigned to asset groups.

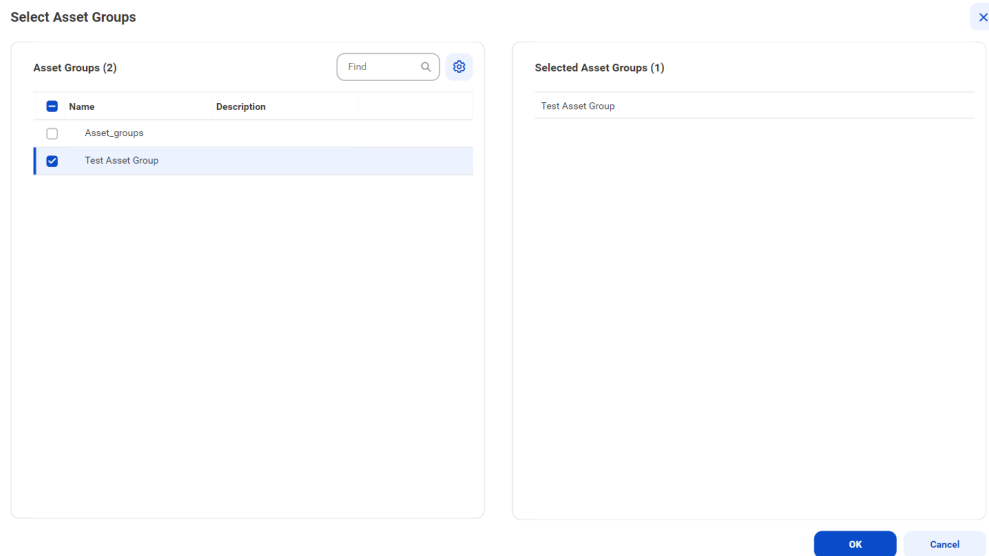
Verify that the administrator assigned users and user groups to the stakeholder role that you want to associate with technical assets.

1. To associate users or user groups as stakeholders with technical assets extracted from the catalog source, perform the following steps:
 - a. On the **Associations** page, click **Stakeholders**.
 - b. Select **Assign Stakeholders**.
 - c. Select a stakeholder role.
 - d. Click **Select** to add users and user groups from the stakeholder role as stakeholders for the technical assets.

The **Add Users & User Groups** dialog box displays a list of users and user groups assigned to the selected stakeholder role.



- e. Select one or more users or user groups to assign as stakeholders for the technical assets, and click **OK**.
Only the selected users and user groups belonging to the specified stakeholder role are granted the permissions to technical assets.
- f. To assign users or user groups from another stakeholder role, click **Add** and then repeat the steps.
2. To assign asset groups to technical assets extracted from the catalog source, perform the following steps:
 - a. On the **Associations** page, click **Asset Groups**.
 - b. Select **Assign Asset Groups**.
 - c. Click **Select**.
The **Select Asset Groups** dialog box displays the list of asset groups.
If you enabled an access policy that includes an asset group, you can only view assets that belong to that asset group.
3. Select the asset groups to which you want to assign technical assets extracted from the catalog source, and click **OK**.



4. Choose to save and run the job or to schedule a recurring job.
 - To save and run the job, click **Save** and then **Run**.
 - To schedule a recurring job, click **Next** to open the **Schedule** page.

Step 4. Run or schedule the job

Choose to run a catalog source job manually, or configure it to run on schedule.

Note: You can't run multiple jobs simultaneously.

You can choose to perform a full or an incremental metadata extraction. A full metadata extraction extracts all objects from the source to the catalog. An incremental metadata extraction extracts only the changed and new objects since the last successful catalog source job run. Incremental metadata extraction doesn't remove deleted objects from the catalog and doesn't extract metadata of code-based objects if applicable.

When you run an incremental metadata extraction job with a filter to include metadata from objects, the job extracts only the objects that have the latest timestamp since the last successful job.

Note: The incremental extraction option appears if it is available for the catalog source.

Run the job manually

Click **Save** to save the catalog source and click **Run**. On the **Run Catalog Source Job** window, click **Run** to run the job.

You can override the capabilities that you selected while configuring your catalog source on the **Configuration** page. The first time you run the catalog source job, the metadata extraction capability is mandatory. From the second run onwards, you can choose to override the configured metadata change option. You can retain, delete, or deprecate objects that are deleted from the source in the catalog. For subsequent runs of the catalog source job, the metadata extraction capability is optional.

Note: You can choose incremental metadata extraction for subsequent runs only after one full metadata extraction job completes successfully. Incremental metadata extraction jobs run with the **Retain** metadata change option even if you set the option to **Delete** or **Deprecate** in the catalog source.

Note: To run a catalog source job, you need permissions on the connection to the source system. To run a catalog source job for catalog sources that reference other source systems, you need permissions on the connections for all the reference source systems.

Run the job on a schedule

You can choose to run metadata extraction and other capabilities on a recurring schedule. You can't choose incremental metadata extraction and full metadata extraction in the same schedule. To create a schedule for incremental metadata extraction, you must have completed at least one full metadata extraction job successfully. If not, first create a schedule for a full metadata extraction.

If an incremental metadata extraction is scheduled to run when the last run details aren't available, the job first performs a full metadata extraction, followed by incremental metadata extraction on subsequent runs.

For example, this can happen in the following scenarios:

- You create schedules for both incremental metadata extraction and full metadata extraction, but schedule the incremental extraction to run before the first full metadata extraction job.
- You create schedules for both incremental metadata extraction and full metadata extraction, but delete the full metadata extraction schedule before its first run.

1. On the **Schedule** tab, select **Run on Schedule**.
The **Schedule** configuration page opens.
2. Click the checkbox corresponding to each capability that you want to include in the schedule.
3. Enter the start date, time zone, and the interval at which you want to run the job.
4. You can manage additional schedules using the following options:
 - To create a new schedule, click the **Add** button.
 - To delete a schedule, click the **Delete** button.
 - To enable or disable a schedule, click the **Enable Schedule** toggle button.

Note: You can create a maximum of one schedule per capability that you enable. If you purged a catalog source or did not run the metadata extraction job, the catalog source job runs metadata extraction before running other scheduled capabilities.

Note: To create a schedule, you need permissions on the connection to the source system. If you lose permissions on the connection after you create a schedule, the scheduled jobs continue to run.

5. Click **Save** to save the schedule.

Monitor job status

After the job runs, you can monitor the status of the job on the **Overview** page of the job.

For more information about job monitoring, see *Administration*.

Step 5. Assign reference catalog source connections to endpoint catalog source objects

When you run the catalog source job, if the catalog source references another source system, a reference catalog source and connection get created that point to the reference source system. To view the complete lineage for your catalog source, you can perform connection assignment from the reference catalog source connection to the objects in the reference source system. A reference source system might be a database,

such as Microsoft Azure SQL Server. You must first create and run an endpoint catalog source that connects to the reference source system.

Before you assign a connection, ensure that you have created and run an endpoint catalog source for each reference source system.

Note: If the source schema contains case-sensitive tables or if the reference objects contain multiple objects with the same name in different cases, perform case-sensitive connection assignment to get correct lineage.

If you enabled the lineage discovery capability for your catalog source, you can either curate the CLAIRE recommended endpoint objects on the **Related Catalog Sources** tab or assign connections manually.

For more information about related catalog sources and lineage discovery, see *Lineage discovery* in the *Administration* help.

1. On the **Configure** page, select the **Lineage** tab, and then select the **Lineage Discovery** tab. On the **Catalog Sources** panel, select the required catalog source and click the **Assign Connections** tab.
The **Assign Connections** tab displays a list of assigned and unassigned connections along with details for each connection. Use filters to view the connections based on the connection names. Click the **Add Filter** menu to add filters.
2. Select the connection to the reference source system and click **Assign**.
The connection name appears prefixed to the reference catalog source name on the **Hierarchy** tab of your catalog source in Data Governance and Catalog.
The **Assign Connection** dialog box appears with a list of recommended objects from the endpoint catalog sources. Click **All** to view all endpoint catalog source objects.
3. In the **Assign Connection** dialog box, select one or more catalog sources to assign to the selected connection and click **Assign**.
You can assign a Microsoft Azure SQL Server as referenced source system. To create a connection assignment to Microsoft Azure SQL Server Script catalog sources, the referenced catalog source must belong to the Schema class type.
4. Run the catalog source job again. If you configured the catalog source job to run on a regular schedule, the next scheduled run picks up the updated details. If you didn't configure a schedule, run the catalog source job again to view complete lineage.

CHAPTER 4

View results in Data Governance and Catalog

After Metadata Command Center runs a job, you can view the results in Data Governance and Catalog where the catalog source and its elements are called technical assets. You can view a catalog source as a hierarchy. Expand each technical asset to see its components.

When referenced source systems are connected to a catalog source, you can expand the hierarchy to see details about the technical asset's component elements.

You can view the data lineage of an asset contained within a catalog source to see individual elements such as data sources, calculations, and filters. When you view data lineage, you can see the individual upstream elements that contribute data or expressions to each component of a data flow or catalog source.

View metadata extraction results

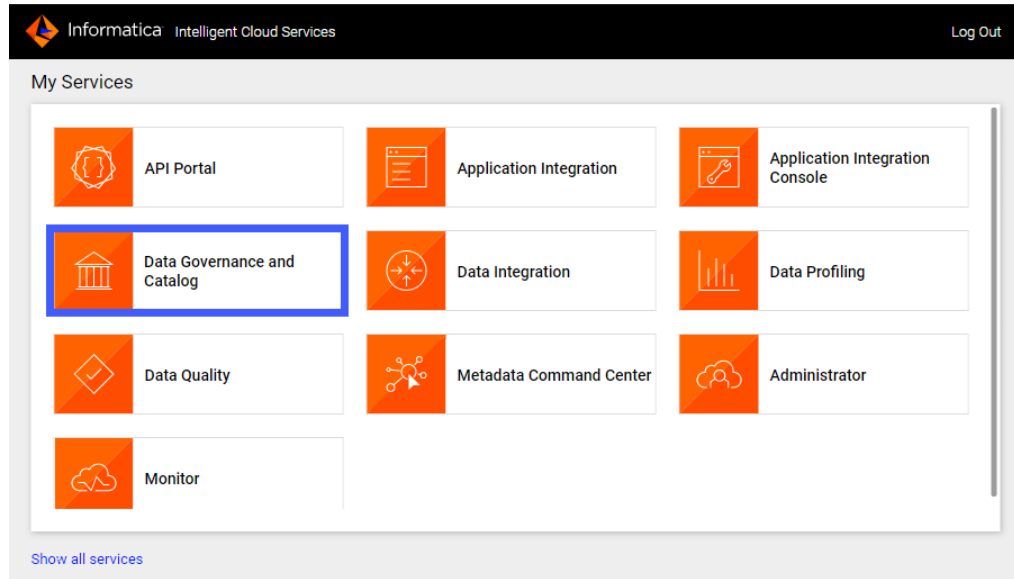
After a job runs in Metadata Command Center, view the results in Data Governance and Catalog. You can view details about source system contents as hierarchical displays and trace data lineage.

1. Log in to Informatica Intelligent Cloud Services.

The **My Services** page appears.

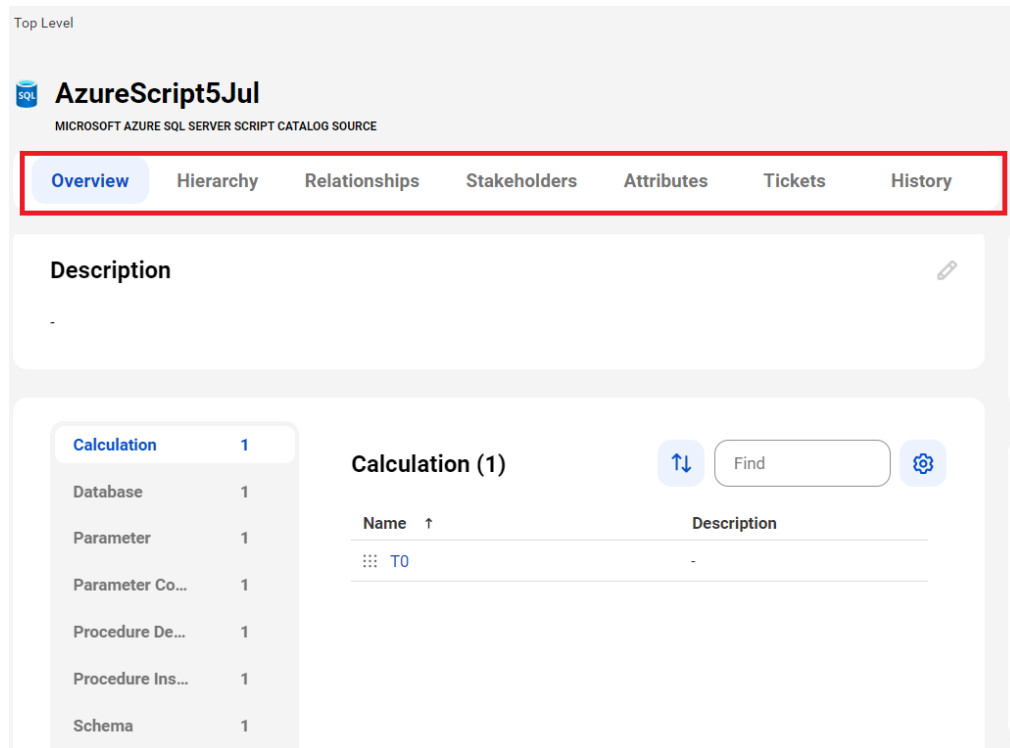
2. Click Data Governance and Catalog.

The following image shows the Data Governance and Catalog box on the **My Services** page:



3. On the Data Governance and Catalog home page, click the number in the **Technical Assets** panel.
The **Technical Assets** page opens.
4. Select **Catalog Source** in the **Filter** list.
The list of catalog sources opens.
5. Search for the catalog source from which you extracted metadata, and click the name.
The **Overview** tab of the asset opens.

The following image shows a sample asset page:



6. View the asset from different perspectives by clicking on the tabs.

For more information about working with assets, see *Cloud Data Governance and Catalog* help.

View data lineage

Data lineage is a visual representation of the flow of data across the systems in your organization. Lineage depicts how the data flows from the system of its origin to the system of its destination.

Data lineage views are available for technical assets in the catalog source. You can view lineage at the catalog source, data set, or data element level.

The lineage at the catalog source level shows how data flows from one catalog source to another. The lineage at the data set and the data element levels show how other technical assets such as files or tables contribute to the selected asset.

If linking catalog sources is available for your catalog source, you can use Metadata Command Center to generate data lineage based on rules or by generating automated lineage with CLAIRE. You can choose source and target catalog sources and objects to link and generate lineage.

To determine whether linking catalog sources is available for your catalog source, navigate to the **Configuration** tab of the **Link Catalog Sources** page. The catalog source must appear in the list of source and target catalog sources.

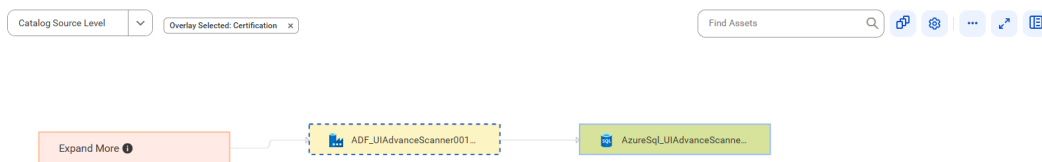
For information about linking catalog sources, see *Link catalog sources* in the Administration help.

View lineage at the catalog source level

The catalog source level shows how data flows from one catalog source to another with the lineage aggregating data from the data set and data element levels.

To view data lineage at the catalog source level, open a technical asset, click the **Lineage** tab, and then verify that the level is set to **Catalog Source Level**.

The following image shows how the AzureSql_UIAdvanceScanner catalog source gets data from the ADF_UIAdvanceScanner catalog source after connection assignment:



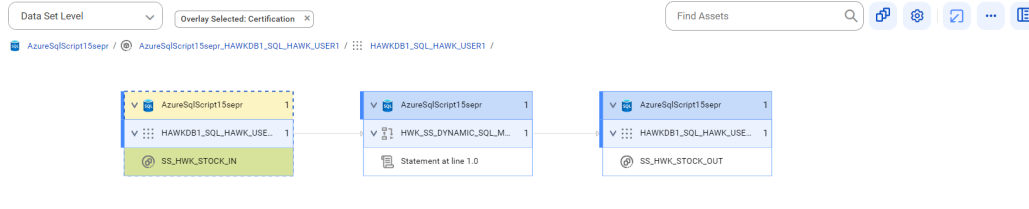
After connection assignment, the referenced object icons change to specific object icons.

View lineage at the data set level

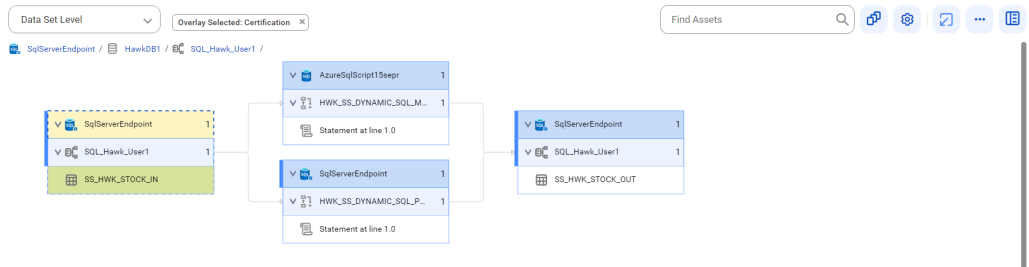
The data set level displays individual sets of data in the data flow.

To view lineage at the data set level, open a technical asset, click the **Lineage** tab, and then verify that the level is set to **Data Set Level**.

The following image shows data set level lineage where the SS_HWK_STOCK_OUT reference table gets data from the SS_HWK_STOCK_IN referenced table before connection assignment:



The following image shows data set level lineage where the SS_HWK_STOCK_OUT table gets data from the SS_HWK_STOCK_IN table after connection assignment:



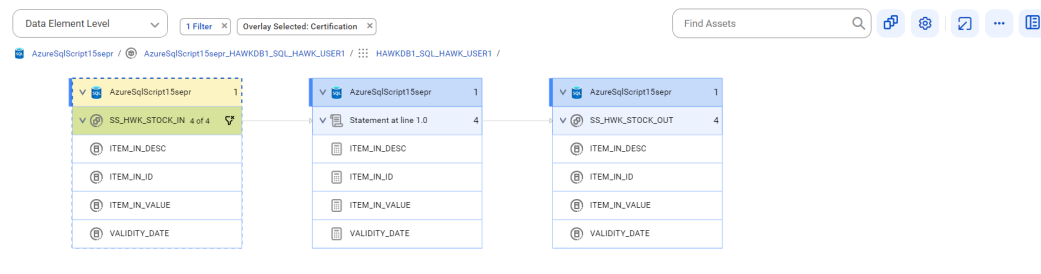
After connection assignment, the referenced object icons change to specific object icons.

View lineage at the data element level

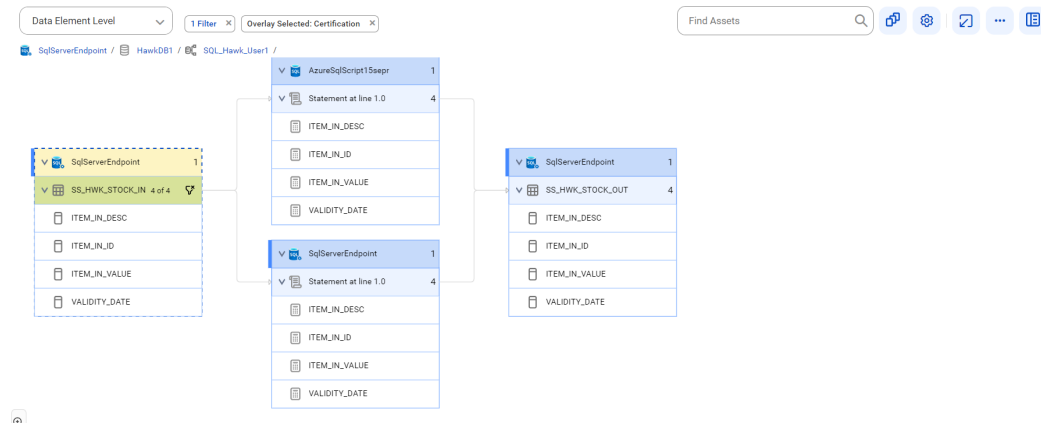
The data element level displays details of the data set level. At the data element level, you can see the input sources for expressions or commands and calculations or transformations on the data.

To view data lineage at the data element level, open a technical asset, click the **Lineage** tab, and then verify that the level is set to **Data Element Level**.

The following image shows data element level lineage where the ITEM_IN_DESC reference column gets data from the SS_HWK_STOCK_IN referenced table before connection assignment:



The following image shows data element level lineage where the ITEM_IN_DESC column gets data from the SS_HWK_STOCK_IN table after connection assignment:



After connection assignment, the referenced object icons change to specific object icons.