

Adding Microsoft SQL Server Data Warehouses

Microsoft SQL Server is a data warehouse that contains the tables and views that you want to access as cube facts and dimensions. It also contains aggregate-table instances that either you or the AtScale engine creates in a schema that you specify.



- ▶ To configure a MSSQL Data Warehouse, you must have a special AtScale license. Ask your sales representative about obtaining a MSSQL-enabled license.
- ▶ Microsoft supports database compatibility levels. The compatibility level is the version of SQL Server that your database uses. AtScale supports SQL Server Version 2016 and above which starts at compatibility level 130. Support for Azure Database starts at compatibility level 130 as well.
- ▶ Microsoft SQL Server does not support POSIX Regular Expressions. Inbound queries that specify POSIX Regular Expressions, such as the Hive RLIKE function generated by some Tableau text filters, are not supported when connected to MS SQL Server.

Before You Begin

- ▶ Ensure that your user ID in the Design Center is assigned the Super User role or is assigned the Manage Data Warehouses role permission.
- ▶ **Aggregate Schema:** Ensure that you know the schema to use for aggregate tables to be built in the data warehouse. AtScale creates tables, drops tables, and reads and writes aggregate data to this schema.

Add A Microsoft SQL Server Data Warehouse

1. Go to **Settings > Data Warehouses** and click **Create Data Warehouse**.
2. In the Add a Data Warehouse dialog box, under **Type of Data Warehouse**, select **Microsoft SQL Server**.
3. Enter a unique name for the data warehouse. AtScale displays this name in the Design Center and uses it in log files.
4. Enter the name of the **External Connection ID**. This input will be inherited by the **Name** value, but can be overridden by enabling the **Override generated value** slider.
5. Enter the name of the **Database** for AtScale to use when creating aggregate tables. (You must create this database; AtScale does not automatically create a database if one does not exist.)
6. Enter the name of the **Aggregate Schema** for AtScale to use when creating aggregate tables. (You must create a schema; AtScale does not create a schema if one does not exist.)
7. Specify the Data Warehouse as a **Read-only source**.

- ▶ Select this option if AtScale will be configured to access this database with credentials that do not have permission to create new tables, or if you need to prevent AtScale from placing any aggregate tables in this data warehouse.
 - ▶ If **Read-only source** is enabled upon editing an existing data warehouse with aggregates present, all existing aggregates stored in this data warehouse will be deactivated.
8. If desired, enable **Impersonation** to allow AtScale to communicate with the data platform using the BI tool end-user user credentials. This allows the data warehouse administrator to apply more detailed security policies when securing data.
 9. If Impersonation is enabled, toggle the settings as needed:
 - ▶ **Always Use Canary Queries:** Whether canary queries are required in light of aggregate misses.
 - ▶ **Allow Partial Aggregate Usage:** Allows mixed aggregate and raw data queries.



AtScale cannot import data-loader bundles while impersonation is enabled. It is suggested that you import your desired data-loader bundles, commonly used for training, before enabling impersonation or use a separate AtScale installation for training purposes.

10. Select the Custom Function Installation Mode:
 - ▶ **Customer Managed** - If selected, the MSSQL administrator installs the custom functions in a designated schema. See [Customer-Managed UDAF Installation on a MSSQL Cluster](#).
 - ▶ **None** - Available for certain management edge-cases. Do not use this settings when servicing end-user queries because system performance will be degraded.

Add A MSSQL Connection

After setting up your data warehouse, you must add a connection(s) to the data warehouse before you can run queries. Expand the Microsoft SQL Server Data Warehouse, and select **Create Connection**.

1. Enter the unique name of the Microsoft SQL Server connection.
2. For **Host**, enter the Host for which your MSSQL Server resides.
3. For **Port**, enter the port number to route transmitted data. The default port number is 1433.
4. If necessary, enter the JDBC flags that are needed for the connection in the Extra JDBC Flags field:
 - ▶ For TLS based connections, set `;trustServerCertificate=true` . For additional details please refer to [Connecting with encryption](#).
 - ▶ For Kerberos integrated authentication, set `;integratedSecurity=true;authenticationScheme=JavaKerberos;` . For additional details please refer to [Using Kerberos integrated authentication to connect to SQL Server](#).

5. Select the **Authorization** type:

- ▲ If Username and Password authentication is used, select **Password**.

If you have [external secret manager](#) enabled and configured, enter the corresponding credentials.

- ▲ Select **Kerberos** if [Kerberos authentication](#) is configured.

6. Test the connection.

7. Click **Save** to complete the setup.