

Import From SSAS (SQL Server Analysis Services)

AtScale supports the limited conversion of SQL Server Analysis Services (SSAS) models to AtScale models. This is a convenient way to bootstrap a SSAS migration project because it eliminates a large amount of human error from the migration process. The AtScale SSAS importer will apply best practices to convert SSAS semantics to AtScale semantics. However, the imported model must be reviewed by a data architect to ensure that all modeling use cases are satisfied.

Although the semantics of the two model types are similar, they do not match exactly. To bridge the gap between AtScale and SSAS semantics, AtScale may insert hierarchy levels or attributes to create a valid AtScale model. AtScale will not convert SSAS model components that are not supported in AtScale.

Additionally, some SSAS modeling concepts are supported in a different way in AtScale. In these cases, the importer will create the equivalent AtScale model structure. The import report contains a comprehensive record of the conversion process including items omitted because they are unsupported as well as items that are converted to AtScale concepts. See [Limitations of the AtScale SSAS Importer](#) below.

Before You Begin

- ▶ Ensure that your user ID is assigned to a role that has the Create Projects permission for the current organization. By default, the Organization Admin role has this permission.
- ▶ Export the SSAS database definition in XMLA format and save as an .xml file. You must export the entire database, not just a single cube.

Procedure

1. On the AtScale home page's Quick Start section, click the right shuffle control ">" to scroll to the **Import from SSAS** quick start option and click it.

The Project Wizard appears. Use the **Next** button to step through the wizard.

2. Click the **Browse** button and choose an XMLA .xml file from your local file system.

If AtScale detects errors in the file, you will see an error message. If this happens, you have to correct any errors in the file and re-upload it before proceeding.

3. Enter the project name. By default, the project name is pre-populated from the XMLA file.
4. Enter a name for each cube in the project. By default, the cube name is pre-populated from the XMLA file.

After clicking **Next**, you may have to wait while the system processes the XMLA file. The wait time depends on the size of the model and network bandwidth. Do not click any buttons while waiting.

5. Review the import summary. Click the **Back** button to change the project or cube names. Click the **Next** button to continue.
6. If necessary, review on-screen warnings.
7. Download and review the conversion report. The conversion report contains more detail about the conversion process in the form of info, warning, and error messages.
8. Click the **Next** button to advance.
9. Set the initial permissions for your project. Projects and cubes are created with the system-configured default **design-time** and **run-time** security values. The default system values for design-time and run-time access are **open** to all users. A Super User Administrator can change this behavior to **Exclusive Access** by changing the **Default Project/Cube Security** from the **Settings > Organization Settings: Options** page.



Note: A user must have the **Administer Organization** permission, or be the installation Administrator to change run-time and design-time security settings. When creating a project with a Wizard, a non-administrator can see the security settings of the project, but cannot change them.

- ▲ **Yes, allow all:** Allow all users in the organization to read, update, delete, and publish the project. The initial cube in the project is created and all users in the organization have read, update, and delete permissions on it.
 - ▲ **No, just me:** Allow only yourself, organization admins for the current organization, and super users to access the project. The initial cube in the project is created with the same restriction.
10. Click **Finish** to exit the wizard and load the cube.

Post-Procedure

1. If the data warehouse does not contain the same schemas and tables as the SSAS database you must either:
 - ▲ Migrate the SSAS schema and tables to the Big Data warehouse, or
 - ▲ Update each dataset in the AtScale model with the desired schema and table names.
2. Review the import report and make any additional edits to the model to finish your SSAS migration. Common activities include:
 - ▲ Manually add supported features to the model that are not supported by the importer. See [Limitations of the AtScale SSAS Importer](#) below.
 - ▲ Migrate MDX cube build script calculations to either the AtScale cube as calculated measures, calculated columns, and query datasets, or to workbook calculations.
 - ▲ Make non-unique captions unique. SSAS allows for duplicate attribute caption names, however AtScale discourages this practice. AtScale will display the following warning message on attribute dialogs if duplicate attribute names are detected, "This display name is already in use". To avoid accidentally using the wrong attribute in reports and calculations you should only use globally unique attribute captions.

3. Test the model with [Cube Data Preview](#).
4. Publish the model and query it with a supported Business Intelligence Tool.

Limitations Of The AtScale SSAS Importer

SSAS Features that Require Manual Creation in AtScale: Create the following features in AtScale with Design Center.

- ▲ [Shared degenerate dimensions](#)
- ▲ [Perspectives](#)
- ▲ [Cube and Project Security](#)
- ▲ [Runtime Dimension Security with `Security Dimensions](#)
- ▲ [Aggregation Designs](#)
- ▲ [Partitioned System Defined Aggregate Tables](#)

Differently Supported: The following SSAS semantics are converted to AtScale semantics by the importer.

- ▲ SSAS referenced relationships are modeled in AtScale as secondary (aka, "embedded") dimensions.
- ▲ SSAS many-to-many relationships typically have four underlying tables. This design is converted to the following AtScale dimension design: A) Fact dataset --> B) Intermediate dimension --> C) Embedded dimension (from SSAS measure group) --(M2M)--> (D) Embedded dimension.
- ▲ AtScale requires that all attributes exist in hierarchies, while SSAS does not. The AtScale importer creates AtScale hierarchies using the attribute relationships defined in SSAS.
- ▲ Non-additive measures without a corresponding dimension on which to aggregate are created using the Max aggregation function.
- ▲ AtScale dimensions cannot be overridden per cube. Therefore, SSAS dimensions that override dimension definitions with subsets and joins have dedicated dimension definitions in AtScale.

SSAS features NOT supported by AtScale: The following SSAS features have no AtScale corollary.

- ▲ Ragged hierarchy levels
- ▲ Parent-child relationships
- ▲ Data mining structures
- ▲ Metrical attributes
- ▲ Calculated dimension members
- ▲ KPIs
- ▲ Proactive caching of dimensions, cubes, and measure groups
- ▲ MDX scripting language, (aka, MDX scripts for cube building)
- ▲ Dimension-only design-time security

