

# Modeling Relationships

This section explains how to model relationships in your AtScale cube to get the desired matching and filtering behavior when the cube is queried. In the AtScale cube model, a relationship links a logical dimension to a physical dataset. Relationships are not modeled between two datasets directly.

When you model a relationship in an AtScale cube, you are providing information that the AtScale query engine can use to join the underlying tables at query time.

You will need to have your dimensions modeled before you can model the relationships of a cube. See [Add Dimensions to Projects and Cubes](#) for more information about defining dimensions.

- ▶ [Types of Relationships in an AtScale Cube](#) This section explains the types of relationships you can model in an AtScale cube. A relationship creates a link between a physical dataset and a logical dimension, and adds an instance of the dimension to the cube.
- ▶ [Model a Relationship](#) In an AtScale cube, you don't model relationships directly between two datasets - you create relationships between a dataset and a logical dimension. A dimension is not considered part of a cube until it has a relationship to the fact data of the cube (either directly or indirectly).
- ▶ [About Keys and Foreign Keys](#) Two datasets can be joined together on columns that they have in common. The column or columns in the dimension dataset are modeled as the key level of a dimension hierarchy. This key level can then be linked to the corresponding columns in the fact dataset (known as the foreign key).
- ▶ [About Join Behavior](#) When a query involves data from two different datasets, AtScale uses the relationships defined in the cube to plan how to join the underlying tables in the connected data warehouse. This section explains the join behavior considered by the AtScale query engine.
- ▶ [Configure Join Behavior on a Dimension](#) You can configure a dimension hierarchy to specify the default matching and filtering behavior used by the AtScale query engine.
- ▶ [About Factless Queries](#) This section explains factless, dimension-only queries and use cases associated with them.