Dimensions

Dimension files define the dimensions used in the model. A dimension is a logical collection of attributes that are bound to specific columns in a source dataset. These attributes are in turn used to group and filter metric data at query time.

AtScale supports the following types of dimensions:

- **Normal:** Dimensions that are based on a dataset. All data for a normal dimension is normalized into a single table or view. There are two types of normal dimensions:
  - **Standard:** Can have any type of hierarchy.
  - **Time:** Must have a time hierarchy.
- **Degenerate:** A dimension that is based on one or more columns in a fact dataset.
- **Shared degenerate:** A dimension that is based on one or more columns that are common to two or more fact datasets.
- **Snowflake:** A logical dimension that is composed of multiple underlying physical datasets.
- **Many-to-many:** Also called multi-valued. This is when a fact dataset row refers to more than one row in a dimension dataset. In AtScale, this is modeled by defining a dimensional bridge or junction table to resolve the many-to-many relationship.

For more information on using dimensions in AtScale, see Modeling Dimensions.

Dimension files support the following properties.

Unique_name

- **Type:** string
- **Required:** Y

The unique name of the dimension. This must be unique across all repositories and subrepositories.

Object_type

- **Type:** const
- **Required:** Y

The type of object defined by the file. For dimensions, this value must be `dimension`.

Label

- **Type:** string
- **Required:** Y

The name of the dimension, as it appears in AtScale. This value does not need to be unique.

Description

- **Type:** string
- **Required:** N

A description of the dimension.

Type

- **Type:** enum
- **Required:** N

The type of dimension defined by this file.

Supported values:

- `standard` : Can have any type of hierarchy.
- `time` : Must have a time hierarchy.

Hierarchies

- **Type:** array
- **Required:** Y

Defines the hierarchies within the dimension.

Hierarchies organize the dimension attributes into categories or levels, where each level is a subdivision of the level above. Every logical dimension you create has at least one hierarchy with at least one level.

The `hierarchies` property within a dimension file supports the following properties.

- `unique_name` : String, required. The unique name of the hierarchy. This must be unique within the dimension.
- `label` : String, required. The name of the hierarchy, as it appears in AtScale. This value does not need to be unique.

- `levels` : Array, required. Defines the levels within the hierarchy. You can include as many levels as needed in the list.
  Supported properties:

  - `unique_name` : String, required. Specifies the unique name of the level. This must be unique within the dimension.
  - `secondary_attributes` : Array, optional. Defines secondary attributes for the dimension level. For more information, see secondary_attributes below.
  - `aliases` : Array, optional. Defines secondary attributes that can be used as aliases for specific hierarchy levels within BI tools. For more information, see aliases below.
  - `metrics` : Array, optional. Defines metrics for the level. For more information, see metrics below.

- `description` : String, optional. A description of the hierarchy.
- `folder` : String, optional. The name of the folder in which to display this hierarchy in BI tools. If your model has a lot of dimensional hierarchies, folders are a good way to organize them.
- `filter_empty` : String, optional. Configures the join behavior for the hierarchy, which determines how empty values are handled in client BI tools. The value you specify must be in quotes.
  Supported values:

  - `yes` : Query results in BI tools only include members that join to the fact dataset (inner join behavior). Members with no matching entries in the fact dataset are still included if the client BI tool requests them.
  - `no` : Query results include all members of the dimension, even those that have no matching entries in the fact dataset (outer join behavior). This occurs unless the client BI tool specifically requests to have these values filtered out.
  - `always` : Query results only include members that join to the fact dataset (inner join behavior). This typically provides the best performance.

- `default_member` : String, optional. Defines a member of the hierarchy to use as the default filter for MDX queries on the hierarchy. The value must be formatted as an MDX expression and must be in quotes. For more information, see About Default Hierarchical Members.

Secondary_attributes

- **Type:** Array
- **Required:** N

Secondary attributes are dimensional attributes that are not the dimension's key, and are not part of a hierarchy.

AtScale supports the following types of secondary attributes:

- **Dimensional:** Provides an independent "dimensional" attribute for grouping metric data. This is the default type

of secondary attribute.

- **Level alias:** Enables the creation of tabular reports that select hierarchical expressions without forcing the user to drill down a hierarchy.

Note: Secondary attributes cannot be used to create relationships between datasets and dimensions.

For more information on secondary attributes, see About Dimension Attributes.

The `secondary_attributes` property supports the following properties:

- `unique_name`: String, required. The unique name of the secondary attribute. This must be unique within the dimension.
- `label`: String, required. The name of the secondary attribute, as it appears in AtScale. This value does not need to be unique.
- `dataset`: String, required. The dataset that contains the `key_columns` the secondary attribute is based on.
- `name_column`: String, required. The dataset column that the attribute is based on.
- `key_columns`: Array, required. A list of the key columns that the attribute is based on. If the attribute has a compound key, you should specify all columns that make up the key as a list.
- `sort_column`: String, optional. The column used to sort the attribute's values in result sets. (This only applies to MDX queries.)
- `allowed_calcs_for_dma`: Array, optional. A list of the calculation types that can be used to create dimensionally modified aggregates for the secondary attribute. Note that when working with a time dimension, you can only define calculation types if the `time_unit` property for the level is set to `day` or longer. For more information on dimensionally modified aggregates, see Dimensionally Modified Aggregates.
- `exclude_from_dim_agg`: Boolean, optional. Excludes this attribute from system generated dimension-only aggregates. This is useful if the attribute contains a large number (millions) of distinct values that you don't want to aggregate.
- `exclude_from_fact_agg`: Boolean, optional. Excludes this attribute from system generated fact-based aggregates. This is useful if the attribute contains a large number (millions) of distinct values that you don't want to aggregate.
- `custom_empty_member`: Array, optional. Defines a custom empty member for the attribute.
  This feature allows fact data with missing or invalid foreign key values to be isolated and independently aggregated from those with valid foreign key values. Because fact records with invalid foreign keys are aggregated separately from records referencing valid dimension members, analysts can easily spot data integrity problems and further investigate them.
  Use this feature to ensure that un-joinable values are included in query results and aggregated under a specially designated dimension member called the Custom Empty Member.
  Supported properties:

- `key`: Array, required. A list of the empty member values to use for key fields.
- `name`: String, required. The empty member value to use for name fields.
- `sort`: String, optional. The empty member value to use for the attribute's sort column, if one is specified.

- `description`: String, optional. A description of the secondary attribute.
- `is_hidden`: Boolean, optional. Determines whether the attribute is visible in BI tools.
  Supported values:

  - `false` (default)
  - `true`

- `folder`: String, optional. The name of the folder in which the attribute is displayed in BI tools.
- `contains_unique_names`: Boolean, optional. Determines whether each member of this attribute has a unique name. Do not enable this functionality if two members have different keys but the same name.
  Supported values:

  - `true`
  - `false`

Aliases

- **Type:** array
- **Required:** N

The `aliases` property defines secondary attributes to use as aliases for specific levels within a hierarchy. These are useful in BI tools, as they enable the user to select a specific level without having to navigate through the hierarchy it belongs to. You can include as many aliases as needed in the list.

For more information on aliases in AtScale, see About Dimension Attributes.

The `aliases` property within a dimension hierarchy level supports the following properties:

- `unique_name`: String, required. The unique name of the alias. This must be unique within the dimension.
- `label`: String, required. The name of the alias, as it appears in AtScale and BI tools. This value does not need to be unique.
- `dataset`: String, required. The source dataset that contains the column that the alias is based on.
- `name_column`: String, required. The dataset column that the alias is based on.
- `sort_column`: String, required. The column to use to sort the values in result sets. This applies to MDX queries only (queries received through the XMLA interface).
- `description`: String, optional. A description of the alias.

- `is_hidden`: Boolean, optional. Determines whether the alias is visible in BI tools.
  Supported values:
  - `true`
  - `false`
- `exclude_from_dim_agg`: Boolean, optional. Excludes this alias from system generated dimension-only aggregates. This is useful if the alias contains a large number (millions) of distinct values that you don't want to aggregate.
  Supported values:
  - `true`
  - `false`
- `exclude_from_fact_agg`: Boolean, optional. Excludes this alias from system generated fact-based aggregates. This is useful if the alias contains a large number (millions) of distinct values that you don't want to aggregate.
  Supported values:
  - `true`
  - `false`
- `custom_empty_member`: Object, optional. Defines custom empty member values for the alias.
  This feature allows fact data with missing or invalid foreign key values to be isolated and independently aggregated from those with valid foreign key values. Because fact records with invalid foreign keys are aggregated separately from records referencing valid dimension members, analysts can easily spot data integrity problems and further investigate them.
  Use this feature to ensure that un-joinable values are included in query results and aggregated under a specially designated dimension member called the Custom Empty Member.
  Supported properties:
  - `key`: Array, required. A list of the empty member values to use for key fields.
  - `name`: String, required. The empty member value to use for name fields.
  - `sort_name`: String, optional. The empty member value to use for the alias's `sort_column`, if one is specified.
- `format`: String, optional. The format in which query results are returned. You can use one of AtScale's built-in named formats or a custom format string.
  Supported named formats: `fixed`, `general number`, `none`, `percent`, `scientific`, `standard`
  Custom format strings should be in quotes and contain one to four sections, separated by semicolons. For more information on defining custom format strings, see Number Format Strings.
- `folder`: String, optional. The name of the folder in which the alias appears in BI tools.

Metrics

- **Type:** array
- **Required:** N

The `metrics` property of a dimension level defines secondary metrical attributes for the dimension, which behave like metrics in a very limited context of the data model.

Note: This feature is experimental and must be enabled by an AtScale admin.

For more information on secondary metrical attributes, see Add Or Edit A Metric Within A Dimension.

The `metrics` property within a dimension hierarchy level supports the following properties:

- `label` : String, required. The name of the secondary metrical attribute, as it appears in AtScale. This value does not need to be unique.
- `unique_name` : String, required. The unique name of the secondary metrical attribute. This must be unique within the dimension.
- `dataset` : String, required. The source dataset that contains the column that the secondary metrical attribute is based on. This should be the dimension dataset name.
- `column` : String, required. The column within the `dataset` that the secondary metrical attribute is based on.
- `calculation_method` : String, required. The calculation to apply to the data.
  Supported values: `average` , `count distinct` , `count non-null` , `estimated count distinct` , `maximum` , `minimum` , `percentile` , `stddev_pop` , `stddev_samp` , `sum` , `var_pop` , `var_samp`
- `description` : String, optional. A description of the secondary metrical attribute.
- `is_hidden` : Boolean, optional. Determines whether the secondary metrical attribute is visible in BI tools.
  Supported values:
  - `true`
  - `false`
- `folder` : String, optional. The name of the folder in which this secondary metrical attribute appears in BI tools.
- `format` : String, optional. The format of the data in the column that the measure is based on. You can use one of AtScale's built-in named formats or a custom format string.
  Supported values: `fixed` , `general number` , `none` , `percent` , `scientific` , `standard`
  Custom format strings should be in quotes and contain one to four sections, separated by semicolons. For more information on defining custom format strings, see Number Format Strings.
- `exclude_from_dim_agg` : Boolean, optional. Excludes this secondary metrical attribute from system generated dimension-only aggregates. This is useful if the secondary metrical attribute contains a large number (millions) of distinct values that you don't want to aggregate.

Supported values:

- `true`
- `false`

- `exclude_from_fact_agg`: Boolean, optional. Excludes this secondary metrical attribute from system generated fact-based aggregates. This is useful if the secondary metrical attribute contains a large number (millions) of distinct values that you don't want to aggregate.
  Supported values:

  - `true`
  - `false`

- `custom_empty_member`: Object, optional. Defines custom empty member values for the secondary metrical attribute.
  This feature allows fact data with missing or invalid foreign key values to be isolated and independently aggregated from those with valid foreign key values. Because fact records with invalid foreign keys are aggregated separately from records referencing valid dimension members, analysts can easily spot data integrity problems and further investigate them.
  Use this feature to ensure that un-joinable values are included in query results and aggregated under a specially designated dimension member called the Custom Empty Member.

  Supported properties:

  - `key`: Array, required. A list of the empty member values to use for key fields.
  - `name`: String, required. The empty member value to use for name fields.
  - `sort`: String, optional. The empty member value to use for the secondary metrical attribute's sort column, if one is specified.

- `unrelated_dimensions_handling`: Enum, optional. Determines how the AtScale engine behaves when all of the following conditions are true:

  - A client queries a model that contains multiple fact datasets.
  - The data in each fact dataset are at a different level of granularity than the data in the other fact datasets.
  - The query references dimensions that are not related to the metrics being queried.

  Supported values:

  - `error`: AtScale rejects the query and returns an error message.
  - `empty`: AtScale displays empty cells in the query results.
  - `repeat`: In the query results, AtScale repeats the values for the secondary metrical attribute at a level of aggregation that is determined from the shared dimensions in the query.

Level_attributes

- **Type:** array
- **Required:** Y

Level attributes are attributes associated with a particular dimension hierarchy. Every hierarchy has a key level attribute, which is the most granular representation of the dimension's data. Only level attributes can be used to define relationships between datasets and other dimensions. For more information on level attributes within AtScale, see About Dimension Attributes.

The `level_attributes` property of a dimension file supports the following properties.

Unique_name

- **Type:** string
- **Required:** Y

The unique name of the level attribute. This must be unique within the dimension.

Label

- **Type:** string
- **Required:** Y

The name of the level attribute, as it appears in AtScale. This value does not need to be unique.

Dataset

- **Type:** string
- **Required:** Y

The source dataset that contains the columns that this level attribute is based on.

Name_column

- **Type:** string
- **Required:** Y

The column whose values appear for this level in BI tools. For example, the key may be a product ID number, but you want users to see product names instead.

Key_columns

- **Type:** array
- **Required:** Y

The dataset column that the level attribute is based on. If the level has a compound key, list all columns that make up the key.

If the key consists of one column, the values in that column must be unique. If the key is a compound key, the columns together must provide unique values.

Description

- **Type:** string
- **Required:** N

A description of the level attribute.

Is_hidden

- **Type:** boolean
- **Required:** N

Determines whether the level attribute is visible in BI tools.

Supported values:

- `true`
- `false`

Is_unique_key

- **Type:** boolean
- **Required:** N

Determines whether the `key_columns` values are unique for each row.

Supported values:

- `true`: The key column values are unique for each row. The join behavior considers the first matching row at query runtime.

- `false` : The key column values are multi-valued. The join behavior considers all matching rows at query runtime.

Note: Setting this value to `true` is equivalent to declaring the key to be a primary key. The AtScale engine uses this property as input when joining rows from this dimension level to other datasets in the model.

Contains_unique_names

- **Type:** boolean
- **Required:** N

Determines whether each member of this level attribute has a unique name. Do not enable this functionality if two members have different keys but the same name.

Supported values:

- `true`
- `false`

Exclude_from_dim_agg

- **Type:** boolean
- **Required:** N

Excludes this level attribute from system generated dimension-only aggregates. This is useful if the attribute contains a large number (millions) of distinct values that you don't want to aggregate.

Supported values:

- `true`
- `false`

Exclude_from_fact_agg

- **Type:** boolean
- **Required:** N

Excludes this level attribute from system generated fact-based aggregates. This is useful if the attribute contains a large number (millions) of distinct values that you don't want to aggregate.

Supported values:

- `true`
- `false`

Sort_column

- **Type:** string
- **Required:** N

Defines the column to sort query results on. By default, this is the `name_column`; however, you can optionally use this property to specify a different column.

Note: This only applies to MDX queries (queries received through the XMLA interface).

Allowed_calcs_for_dma

- **Type:** array
- **Required:** N

A list of the calculations that can be used when creating dimensionally modified aggregates for the level attribute.

Folder

- **Type:** string
- **Required:** N

The name of the folder in which this level attribute appears in BI tools.

Time_unit

- **Type:** string
- **Required:** N

For time dimensions only. The unit of time to use.

Supported values: `year`, `halfyear`, `trimester`, `quarter`, `month`, `week`, `day`, `hour`, `minute`, `second`, `undefined`

Relationships

- **Type:** array
- **Required:** N

The `relationships` property in a dimension file defines the relationships to embedded and snowflake dimensions.

Note: The relationships between the model's fact datasets and first order dimensions (fact relationships) are defined in model files.

For more information on relationships in AtScale, see Modeling Relationships.

The `relationships` property supports the following properties.

From

- **Type:** object
- **Required:** Y

Defines the side of the relationship that contains the physical dataset that you want to connect to another dimension.

Supported properties:

- `dataset`: String, required. The physical dataset you want to link to a dimension.
- `join_columns`: Array, required. The column(s) within the `dataset` that you want to use for the join.
- `hierarchy`: String, optional. The hierarchy within the dimension from which the relationship should originate.
- `level`: String, optional. The level within the `hierarchy` from which the relationship should originate.

For snowflake relationships (as defined by the `type` property), you only need to define `dataset` and `join_columns`.

To

- **Type:** object
- **Required:** Y

Defines the dimension that the `from` dataset is linked to.

Supported properties:

- `dimension`: String. The name of the dimension the `from` dataset is linked to.
- `level`: String, required if `row_security` is undefined. The key level within the dimension to use for the relationship.
- `row_security`: String, required if `level` is undefined. For security relationships, the row security object that the `from` dataset is linked to.

For snowflake relationships (as defined by the `type` property), you only need to define `level`.

Type

- **Type:** string
- **Required:** Y

Defines the relationship as either embedded or snowflake.

Supported values:

- `embedded`: A secondary relationship, or one that connects a primary dimension to a secondary dimension.
- `snowflake`: A relationship that connects one of several underlying physical datasets together to create a snowflake dimension.

Role_play

- **Type:** string
- **Required:** N

For role-playing relationships only. Defines the role-playing template for the relationship.

The role-playing template is the prefix or suffix that is added to every attribute in the role-played dimension. You can also specify both a prefix and a suffix.

This value must be in one of the following formats (including quotation marks):

- **Prefix:** `"<prefix> {0}"`
- **Suffix:** `"{0} <suffix>"`
- **Prefix and suffix:** `"<prefix> {0} <suffix>"`

For example, if you wanted to use the prefix **Order**, you would set `role_play` to `"Order {0}"`.

For more information on role-playing relationships in AtScale, see Role-Playing Relationships.

Unique_name

- **Type:** string
- **Required:** N

The unique name of the relationship. This must be unique within the dimension.

Calculation_groups

- **Type:** array
- **Required:** N

The `calculation_groups` property in a dimension file defines calculation groups to use in the dimension.

Dimension calculation groups offer a simplifying alternative to calculated metrics by enabling the expression of boiler-plate calculations across multiple metrics. This feature defines calculations as dimension members and removes static references to individual measures.

For more information on calculation groups, see Using Dimension Calculation Groups.

The `calculation_groups` property in a dimension file supports the following properties.

Unique_name

- **Type:** string
- **Required:** Y

The name of the calculation group. This must be unique within the dimension.

Description

- **Type:** string
- **Required:** Y

A description of the calculation group.

Calculated_members

- **Type:** array
- **Required:** Y

Defines the individual calculations in the group.

Supported properties:

- `unique_name`: String, required. The name of the calculation. This must be unique within the dimension.
- `description`: String, required. A description of the calculation.
- `expression`: String, required. The MDX expression for the calculation. This must be in quotes.

If you plan on referencing a calculated metric via the `Aggregate` MDX function in your calculation, ensure that the calculated metric has an aggregation function set. You can do this by including the `mdx_aggregation_function` property in the calculation file. If you do not set an aggregation function, you may encounter errors at query time. For more information, see Calculations.

- `format`: String, optional. The format for the calculation results. You can use one of AtScale's built-in named formats or a custom format string:
  - Supported named formats: `fixed`, `general number`, `none`, `percent`, `scientific`, `standard`
  - Custom format strings should be in one of the formats described in Formats for Data Values.

Folder

- **Type:** string
- **Required:** N

The name of the folder in which the calculation group appears in BI tools.