

Query Settings

You can view and modify the following engine-level settings for queries.

Language Settings

- ▶ `query.language.mdx.dimquery.result.limit` : When you are using very large dimensions, you can use this setting to set the number of data rows (for example, 10000) to be returned for Dimension-only queries (if enabled). When set to -1 it is disabled, and the limitation from the `query.result.max_rows` setting is applied.
- ▶ `query.language.mdx.vba.format.firstdayofweek` : Determines the default value used for the `First_Day_Of_Week` parameter of the `Format` VBA function. The default value is `vbSunday` . For more information, including supported values, see [Format](#).
- ▶ `query.language.mdx.vba.format.firstweekofyear` : Determines the default value used for the `First_Week_Of_Year` parameter of the `Format` VBA function. The default value is `vbFirstJan1` . For more information, including supported values, see [Format](#).

Including Comments In Queries

When you need to track and analyze queries, you can include dedicated comments in the inbound queries.

To configure AtScale to propagate these comments to the outbound queries, enable (set to true) the following engine settings:

- ▶ `query.includeComments.general.enabled` : Whether to include comments in the outbound query from system-generated queries. Default value is true. Should be set in Custom Settings.
- ▶ `query.includeComments.general.userComments` : Whether to include the user comments in the outbound query from inbound queries. Default value is true.
- ▶ `query.includeComments.general.queryId` : Whether to include the query id in the outbound query from system-generated queries. Default value is false. Should be set in Custom Settings.
- ▶ `query.includeComments.general.userId` : Whether to include the user id in the outbound query from system-generated queries. Default value is false. Should be set in Custom Settings.

For SQL language queries, the comments must be located in the body of the query, right after the `SELECT` keyword. They should be wrapped like this:

```
/* user_comment(` comments here `) */
```

Here is an example:

```
SELECT /* user_comment(` comments here `) */
  CAST('Can you see the comment?' AS BOOLEAN)
FROM
  asadventuresmall
LIMIT
  1
```

For MDX language queries, the comments must be located in the body of the query, in the PropertyList > UserComments tag. Here is an example:

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <Execute xmlns="urn:schemas-microsoft-com:xml-analysis">
      <Command>
        <Statement>
          (statement ...)
        </Statement>
      </Command>
      <Properties>
        <PropertyList>
          <UserComments>This is a user comments space</UserComments>
          <Catalog>EngineDemo</Catalog>
          (other tags ...)
        </PropertyList>
      </Properties>
    </Execute>
  </soap:Body>
</soap:Envelope>
```

Semi-Additive Metric Settings

- ▲ `query.planning.semiAdditive.defaultAggregationWhenOutOfScope` : Enables/disables standard aggregation of semi-additive metric across standard (non-semi-additive) dimensions. The default value is `true` . You can set this value to `false` to regain the pre-12023.4.1 behavior of not aggregating results across additive dimensions; this is equivalent to a global First/Last Value Not Empty aggregation function. You do not need to restart the engine after updating this setting.