Managing Aggregates

This section explains how AtScale creates and uses aggregate tables (or aggregates for short) to optimize query performance, how you can define your own aggregates, and how administrators can monitor and configure aggregates.

- ▲ About Aggregates AtScale incorporates the data-warehousing concept of aggregate tables. Such tables most often contain measures from one or more fact datasets and include aggregated values for these measures. (There are dimension-only aggregate tables.) The aggregation of the data is at the level of one or more dimensional attributes or, if no dimensional attributes are included, the aggregated data is a total of the values for the included measures.

- ▲ Changing the Schema for Aggregate Tables AtScale creates its aggregate tables in your data warehouse in a schema that a system administrator creates before deploying AtScale. It is possible to change the schema after AtScale is deployed.

- ▲ Engine Settings for System-Defined Aggregates Only There settings that you can use to influence the behavior of the AtScale engine with regard to system-defined aggregates. The settings apply to all of the aggregates that are in the organization for which an AtScale engine is registered.

- ▲ Engine Settings for User-Defined Aggregates Only Use these settings to control partitioning of user-defined aggregates.

- ▲ Engine Settings for Both System-Defined and User-Defined Aggregates There settings that you can use to influence the behavior of the AtScale engine with regard to system-defined and user-defined aggregates. The settings apply to all of the aggregates that are in the organization for which an AtScale engine is registered.

- ▲ Configuring Aggregate Life Cycle Settings You can configure aggregate life cycle settings to control the number of system-defined aggregates (both demand-defined and prediction-defined).

- ▲ Priming the Aggregate System with Demand-Defined Aggregates You can cause the AtScale engine to generate demand-defined aggregates before any analysts start querying your data. By priming your cubes with aggregates, you might be able to improve the performance of the initial queries that analysts issue from their data-analytics software. Such priming is done by enabling Training Mode on the User Profile page and running queries representative of the queries that analysts will issue.

- ▲ Exporting and Importing System-Defined Aggregate Definitions Primed aggregate sets can be exported from a Staging system and imported into a Production System. Alternatively, Production aggregate sets can be migrated to Staging systems to support performance tuning activities.

- ▲ Disabling the Creation of System-Defined Aggregates for a Dataset For some fact datasets, you might want to disable the creation of system-defined aggregates entirely.

- ▲ Defining Aggregates Yourself You can define your own aggregates for use cases that fall outside of those covered by system-defined aggregates.

- ▲ Monitoring Aggregate Usage You can view what aggregates have been created, when they were last updated, and how often they have been used by a query. You can also see the status of instances and a history of instances for each aggregate.

- ▲ Rebuilding Aggregates Using the REST API You can use the `aggregate-batch` endpoint of the AtScale engine

REST API to trigger an initial build or rebuild for all aggregates of a published cube. The Design Center provides a sample Bash script that runs curl commands to authenticate with and post a request to the `aggregate-batch` endpoint for the cube.

▲ Rebuilding Aggregates Manually You can manually perform an initial build or a rebuild all aggregates for a published cube.

▲ Triggering Aggregate Rebuilds You can trigger a rebuilding of aggregate instances whenever a specified file in the distributed file system of your data warehouse is updated. Aggregate rebuilds are triggered per cube.

▲ Scheduling Recurring Builds You can schedule builds of the aggregate tables for individual cubes to run at specific times on specific days of the week. The schedules repeat every week until you delete them.

▲ Setting Properties to Allow Incremental Rebuilds of Aggregates If you want to allow the AtScale engine the option of performing incremental rebuilds of the aggregates for a cube, you must change settings in the properties of the fact dataset and dimensions of the cube

▲ Performing Full Rebuilds of Incremental Aggregates Incremental aggregates process only new windows of fact data rows, rather than processing all of the data. In some cases, you may want to reprocess all of the data to ensure that aggregates are accurate. For example, if the data of a dimension dataset has changed significantly or if you have older data that missed a processing window.

▲ Handling NULL Values to Prevent Incomplete Aggregate Tables and Unexpected Query Results

▲ Configuring Aggregate Maintenance The AtScale engine can check the health of the aggregate instances using a dedicated maintenance job. You can choose when and how this job would be run.