Cloudera Data Visualization 7.2.1

Data

Date published: 2020-10-30
Date modified: 2024-02-29

https://docs.cloudera.com/

Legal Notice

Contents

Datasets in Data Visualization

Datasets are the foundation and starting point for visualizing your data. They are defined on the connections to your data and provide access to the specific tables in the data store.

A dataset is the logical representation of the data you want to use to build visuals. It is a logical pointer to a physical table or a defined structure in your data source. Datasets may represent the contents of a single data table or a data matrix from several tables that may be in different data stores on the same connection.

Other than providing access to data, datasets enhance data access and use in many ways, including (but not limited to):

- Table joins allow you to supplement the primary data with information from various other data sources. For more information, see *Data modeling*.
- Derived fields/attributes support flexible expressions, both for dimensions and for aggregates. For more information, see *Creating calculated fields*.
- Hiding fields enables you to eliminate the fields that are unnecessary to the business use case or to obscure sensitive data without affecting the base tables. For more information, see *Hiding dataset fields from applications*.
- Changing data types of the field attributes often helps you to deal with data types, or to ensure that numeric codes (like event ids) are processed correctly. For more information, see *Changing data type*.
- Changing the default aggregation of fields at the dataset level prevents common mistakes when building visuals. For more information, see *Changing field aggregation*.
- Providing user-friendly names for columns or derived attributes often makes the visuals more accessible and saves some of the efforts of applying aliases to each field of the visual. For more information, see *Automatically renaming dataset fields* and *Custom renaming dataset fields*.

Related Information

Data modeling

Creating calculated fields

Hiding dataset fields from applications

Changing data type

Changing field aggregation

Automatically renaming dataset fields

Custom renaming dataset fields

Data Extracts in Data Visualization

Data Extracts are saved subsets of data that you can use for data discovery and analytics.

Key concepts

When you create a Data Extract, you reduce the total amount of data you work with by selecting certain dimensions and measures. After you create an extract, you can refresh it with data from the original dataset. With the help of data extracts you can manage the analytical capabilities, performance, concurrency, and security of data access in your system.

Data Extracts are created from a single dataset but a dataset can have multiple extracts associated with it. You can create a Data Extract in Cloudera Data Visualization on any target connection that supports it. Since the target of an extract is a simple representation of data on the target data connection, you can use it independently as part of a different dataset, or even use it as the source for a further transformation into a second Data Extract.

When a Data Extract is created, it targets a data connection as the new location of the data. This data connection can be the same or a different connection that you have used to build the extract. A second dataset built on the target data will be used as the dataset for building visuals.

You can refresh Data Extracts on a schedule. This functionality creates a new kind of job and uses the standard job scheduling mechanism to schedule and monitor ongoing refreshes. Alternatively, you can do the refresh manually when you need it. For more information on automated refreshes, see Managing schedule intervals.

You can use extracts to model data within or across data connections. You can also use extracts to materialize the data models on the same or different data connections. Data modeling is done in the context of a dataset built on the source connection. The modeled form of the data can be materialized on that same or on a different data connection. Modeling includes everything that is possible within a dataset, including joins, scalar transformations, hidden columns and complex aggregations.

Note: Data Extracts are run on a data connection with the permissions of the user who sets up the Data Extract. This user needs to have all the permissions associated with running a Data Extract, including Manage AVs/Data Extracts permissions on both the source and target data connections, and the permissions in the backend data connection to drop and create the target table, select data from the source table, and insert data into the target table.

Key features

Improving analytical capabilities

Data Extracts support large data sets. You can create extracts from data sets that contain billions of rows of data. Extracts permit joins across data connections. You can move data first and then join the data on the target system.

Data Extracts enable you to use different data connection capabilities for analysis. You can move data from an analytical system to an event-optimized or search-optimized store.

Improving performance

Data extraction offers increased performance when large tables or datasets are slow to respond to queries. When you work with views that use extracted data sources, you experience better performance than when interacting with views based on the original dataset.

With Data Extracts, you can do the following:

- Model storage of data into columnar forms
- Move data to a different query system that performs better

When source data requires expensive join and transformation operations for reporting use cases, you can pre-compute join and transformation operations.

When base tables are large but reporting requires a subset of data, or is only needed on well understood roll-ups, you can pre-compute filters and aggregations as Data Extracts.

Improving workload management

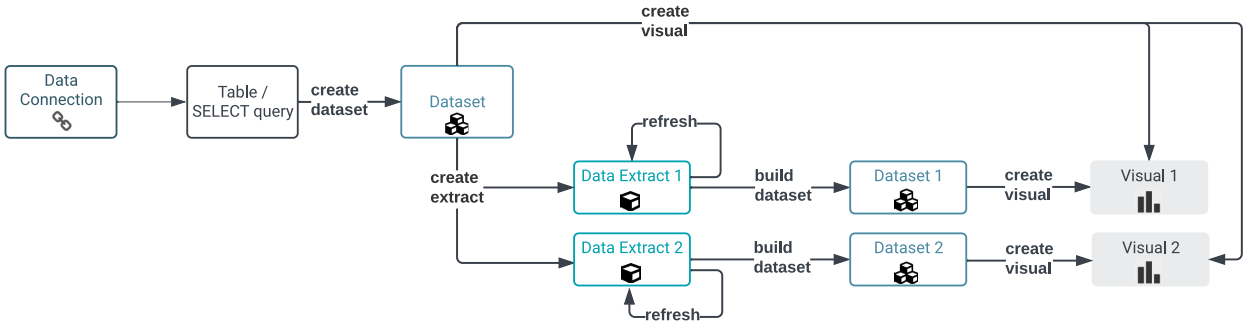
You can use isolated systems for performance characterization on particular workloads by moving data from a busy system to another isolated system.

Improving security

You can provide access only to the relevant reporting data by masking column names and schemas from end users through building dashboards off a derived copy of the data with masking enforced. You can move data that is relevant for reporting from one system to another and ensure that end users only have access to the target system. This ensures that only the data that is relevant for reporting is accessible to end users.

Workflow

The following diagram shows you the workflow of building visualizations on top of data extracts:



Supported sources

Cloudera Data Visualization supports the following sources for Data Extracts:

- Hive
- Impala
- MariaDB
- MySQL
- PostgreSQL
- Spark SQL
- SQLite (not supported in Cloudera Data Warehouse (CDW))
- Snowflake (Technical Preview)

Supported targets

Cloudera Data Visualization supports the following targets for Data Extracts:

- Hive
- Impala
- MariaDB
- MySQL
- PostgreSQL
- Spark SQL
- SQLite (not supported in Cloudera Data Warehouse (CDW))
- Snowflake (Technical Preview)

Data modeling in Data Visualization

With Cloudera Data Visualization, you can create logical datasets that model semantic relationships across multiple data sources. These datasets enable you to build blended visuals, dashboards, and applications.

You can expand a basic, one-table dataset by creating joins with other relevant tables from the same source or from other data stores. The joins created in Data Visualization are not materialized. They are calculated during run-time.

Combining data across multiple tables enriches the dataset considerably; it enables you to conduct much more meaningful research and produce insightful visual analytics.

There are two distinct approaches to create data joins for visualization:

- Defining in UI is ideal for datasets that include star-type schemas.
- Defining on back-end ETL is preferable for fact-fact joins, so they may be pre-materialized.

See *Working with data models* for instructions on how to create and manage joins.

See topics in *How To: Advanced Analytics* for more advanced data modeling details, such as dimension hierarchies, segments, and events.

See *Setting filter associations* for information about filter associations related to data modeling and management.

Related Information

Working with data models

How To: Advanced Analytics

Setting filter associations

Data joins in Data Visualization

With Cloudera Data Visualization, you can create joins between different table columns.

Data Visualization supports five types of column connections, also known as joins.

Note: The types of joins available depend on the underlying database. For example: MySQL connections do not support FULL OUTER JOIN operations. SQLite connections do not support FULL OUTER JOIN and RIGHT OUTER JOIN operations.

Inner join

It is the most common join type. Rows in the result set contain the requested columns from both tables for all combinations of rows where the selected columns of the tables have identical values.

Left join

The result set contains all rows from the left table and the matching data from the right table. Whenever no matching data is available on the right side of the join, the corresponding columns in the result set have the value NULL.

Right join

The result set contains all rows from the right table and the matching data from the left table. Whenever no matching data is available on the left side of the join, the corresponding columns in the result set have the value NULL.

Outer join

An outer join returns all rows from both tables. Whenever no matching data is available on one of the sides of the join, the corresponding columns in the result set have the value NULL.

Left outer join

It supports join elimination in datasets that contain more than one table. By default, this setting is on, but it can be disabled.

Related Information

Working with data models