Contents

[Part 1 1](#_Toc201738511)

[Part 2 2](#_Toc201738512)

## Part 1

As a seasoned data engineer, here's a breakdown of Apache Superset's basics and core concepts:

### What is Apache Superset?

* + It's a data visualization tool that enables you to create charts and dashboards.
  + It's designed to be user-friendly ("extremely easy to create a dashboard") and "enterprise-ready."
  + It's "cloud-native" and lightweight, meaning it doesn't consume a lot of memory.
  + It's comparable to tools like Tableau, Power BI, or MicroStrategy, focusing on data analytics and exploration with a "data-centric methodology."

### Key Functionality and Features:

* + **Data Visualization:** Offers "more than 50" different chart types for diverse visualization needs.
  + **Dashboard Creation:** Allows for interactive and dynamic dashboard building with drag-and-drop functionality for arranging charts and adjusting layouts.
  + **Security (RBAC):** Implements Role-Based Access Control (RBAC) for authorization, allowing administrators to create user groups and grant or deny access to specific resources.
  + **Data Source Connectivity:** Can connect to "many different databases or data sources."
  + **Interactive Filtering:** Supports interactive dashboard filtering for dynamic data exploration.
  + **Sharing:** Dashboards and visualizations can be shared over a local network.

### Core Concepts:

* + **Data Source:** The original database or data system you connect to (e.g., MySQL).
  + **Dataset:** A crucial concept in Superset. Instead of directly connecting to a table each time, you create a "local data set" within Superset. This dataset maps columns to the original data source table. You can apply "light transformation," define queries, and even change data types (e.g., set a timestamp column) within a dataset. Charts are built on top of these datasets.
  + **Chart:** A graphical representation of data derived from a dataset.
  + **Dashboard:** A collection of one or more charts, organized for comprehensive data storytelling. Dashboards are highly customizable with drag-and-drop features.

### Architecture (Two-Layered):

* + **Front-end:** Built with JavaScript, making it lightweight and runnable directly in a web browser.
  + **Back-end:** Developed using Python and Flask, operating independently of Hadoop or Spark ecosystems.
  + **Components (Mandatory):**
    - **Web Server:** Handles user requests and serves the front-end.
    - **Metastore Database:** An external database (e.g., PostgreSQL or MySQL, defaulting to SQLite for local installs) that stores all Superset metadata. This includes definitions of datasets, column mappings, dashboard layouts, charts, and saved queries. In case of a failure, reconnecting to this external metastore database can restore all configurations.
  + **Components (Optional):**
    - **Caching Mechanism:** Utilizes NoSQL tables for caching to "enhance and improve performance."

### Deployment Modes:

* **Sequential Mode (Default):** Suitable for small tables and quick queries (less than 60 seconds). Queries exceeding this time will time out.
* **Distributed Mode:** Supports concurrent access and longer-running queries (though a 60-second timeout is still recommended for optimal performance in production).

### Main Components/Interfaces:

* + **Visual Dashboard Building:** The primary interface for creating and arranging dashboards using a drag-and-drop approach.
  + **Data Exploration:** Used to manage and modify datasets. This includes mapping columns, applying transformations, defining queries for dataset creation, and changing column types.
  + **SQL Lab (SQL Powerhouse):** An integrated development environment (IDE) for writing, testing, and running SQL queries against connected data sources. Results of queries can be saved as datasets and then used to build charts. It's particularly useful for complex queries like federated queries.
* **Creating a Dashboard (Three Main Steps):**
  + **Create a Dataset:** Can be done explicitly from a query in SQL Lab, or implicitly by selecting a table when creating a chart.
  + **Create a Chart:** Based on the newly created or existing dataset.
  + **Add to Dashboard:** Place the created chart onto an existing or new dashboard.
* **Export and Import:**
  + Superset allows you to export various components (datasets, charts, dashboards, saved queries) as .zip files.
  + These .zip files can be directly imported into another Superset instance for sharing or backup purposes.
  + Dashboards can also be exported as JPEG or PDF files for reporting or documentation.

## Part 2

As a seasoned data engineer, here's a step-by-step guide to working with Apache Superset, as demonstrated:

* **Accessing Superset:**
  + Open your web browser and navigate to the Apache Superset instance (e.g., from your quick links page on the virtual machine).
  + Log in using your credentials (e.g., username: admin, password: training).
* **Navigating the Interface:**
  + Upon successful login, you'll land on the home page, which displays your most recent dashboards.
  + The main navigation includes tabs for "Dashboards," "Charts," "Datasets," and "SQL Lab."
  + A prominent "+" (plus) sign provides quick access to create new "Data," "SQL query," "Chart," or "Dashboard."
* **Understanding Datasets:**
  + Datasets in Superset act as an abstraction layer over your raw data sources (databases and tables). They store metadata, not the actual data.
  + You can view existing datasets under the "Datasets" tab.
  + Clicking on an existing dataset will take you to the "Chart" creation interface, pre-selecting that dataset as your data source.
* **Creating a Chart:**
  + **Choose Data Source:**
    - Click the "+" sign and select "Chart."
    - The first step is to "Choose data set." Select an existing dataset (e.g., tutorial\_flight) or implicitly create one by picking a database and table from the "Data" option.
  + **Select Visualization Type:** Choose a visualization type (e.g., "Bar Chart").
  + **Configure Chart Properties:**
    - On the left, you'll see the "Chart source" (your chosen dataset).
    - Superset often provides default metrics (e.g., "count").
    - Define the "x-axis" and "metrics" for your chart. You can select existing columns (e.g., "Cost" for x-axis) or create custom metrics (e.g., AVG(cost)).
    - Click "Update Chart" to preview your visualization.
  + **Save the Chart:**
    - Click "Save."
    - Give your chart a meaningful "name."
    - Optionally, select an existing dashboard to add it to, or choose to create a new dashboard simultaneously.
* **Creating and Managing Dashboards:**
  + **Add Chart to Dashboard:**
    - When saving a chart, you can choose to add it to a new or existing dashboard.
    - Alternatively, from the "Dashboards" tab, you can click the "+" sign to create a "New Dashboard" or "Add a Chart."
  + **Edit Dashboard Layout:**
    - Once a dashboard is created, click "Edit Dashboard."
    - Use the drag-and-drop functionality to arrange charts, add tabs, rows, or columns to customize the layout.
    - Change the dashboard's name if desired.
  + **Save and Publish:**
    - Click "Save" to save your dashboard layout.
    - Dashboards are initially in "Draft" status. To make them visible to others on your network, click "Publish."
  + **Dashboard Actions:**
    - **Refresh:** Updates the data displayed on the dashboard. You can also set an "Auto Refresh Interval."
    - **Full Screen:** Displays the dashboard in full-screen mode.
    - **Save As:** Creates a copy of the dashboard with a new name.
    - **Export:** Download the dashboard as a PDF file or an image (JPEG).
    - **Share:** Generate a link to the dashboard for sharing with others.
* **Working with Datasets (Advanced):**
  + Navigate to the "Datasets" tab and select "Edit" for a specific dataset.
  + **Caution:** Changes here affect all charts built on that dataset.
  + **Metrics:** Add or modify metrics (e.g., AVG(cost)). You can define the name and SQL expression for each metric.
  + **Columns:** View and modify column properties. You can change column names, add descriptions, and define data types (e.g., set a column as a temporal dimension for time-series charts).
  + **Calculated Columns:** Create new columns based on SQL expressions applied to existing columns.
* **Using SQL Lab:**
  + **Access SQL Lab:** Click the "+" sign and select "SQL query," or navigate to the "SQL Lab" tab.
  + **Select Data Source:** On the left, choose your "Database," "Schema" (database), and "Table."
  + **Write and Execute Queries:** Write your SQL query in the editor (e.g., SELECT order\_status, COUNT(\*) FROM orders GROUP BY order\_status).
  + **View Results:** Execute the query to see the results dynamically.
  + **Save Query Results as Dataset:** You can "Save" the query result as a new dataset, which can then be used as the input for charts. This is recommended for building charts based on custom SQL.
  + **Create Chart Directly from Query (Less Recommended):** While you can directly create a chart from the SQL Lab, it's generally better practice to save the query as a dataset first.
* **Exporting Components:**
  + From the "Dashboards" or "Charts" list, you'll find an "Export" button (often a download icon).
  + Exporting a dashboard or chart will generate a .zip file with a timestamp. This .zip file can be directly imported into another Superset instance for backup or sharing.
  + It's recommended to export all components (datasets, saved queries, charts, and dashboards) for comprehensive backups.