Peter Gustafson
Ahana writing sample
Jun 30, 2021

Presto is a fast distributed SQL query engine designed to query relational/non-relational databases and even data lakes. It was built for fast analytic queries.

The need for speed is a requirement every company is looking to solve. Presto delivers best-in-class speed. Read more about [Presto use cases](https://prestodb.io/docs/current/overview/use-cases.html#what-presto-is).

## Presto Deployment
A Presto deployment includes a Presto coordinator and Presto workers.

Presto coordinator	The Presto coordinator is the brain. It parses statements, plans queries, and manages Presto worker nodes. It even tracks the Presto workers via REST.
Presto workers	Presto workers do all the task work and the processing data. Presto workers are nodes sharing data from the coordinator via REST.

```

\-presto coordinator-/
          |  +--fetches data from presto worker nodes
          |      +--then returns final data to the client
          |
\-presto worker nodes-/
          |  +--fetch data from connectors
          |
\-connectors-/
```

## Presto Components

When it comes to managing the data, Presto uses several important components outlined below.

### Presto Catalogs

The Presto catalogs contain data schemas and data source IDs. For example, when you run your SQL statement in Presto, it runs against other catalogs. The catalogs are defined in properties files in the Presto configuration directory.

### Presto Workers
Presto workers manage tasks and processing data. Presto workers are nodes sharing data from the Coordinator via the REST API.

### Tables & Schemas

A table is a set of unordered rows of data that can be organized into named columns/types like below.

CO_ID	AlphaZ	VarI
3535	262646477548	bz7371
5361	262645325615	bz1681
9267	262669591624	bz6261
1724	262652368131	bz4183

- Your schema organizes your tables.
- Catalogs and schemas define your query.

### Connectors

Connectors integrate Presto with external data sources like object stores, relational databases, or Hive.

- You integrate connectors using APIs.
- Presto has over 20 built-in connectors for various data sources.
- Every Presto catalog is associated with a specific connector.
- Multiple catalogs use one connector to access instances & clusters.

### Statements & Queries

Presto executes ANSI-compatible SQL statements. A statement passes the instructions while the query is executing.

When a Presto statement is executed...

```
\-presto parses the statement-/
         |  +--creates a query
         |      +--creates a distributed query plan for presto workers
         |      +--creates queries for presto workers using the query plan
         |
\-presto workers-/
         |  +--fetch data from connectors
         |
\-connectors-/
```

### Stage

Presto executes in stages. Depending on the size of the data, there are several stages that implement different sections of the query.

Stages of a query occur in order like below...

```
\-every query has roots-/
          |  +--which aggregates data from other stages
          |      +--the stages themselves don’t run on Presto workers
          |      +--stages run on the database called push-down
          |
\-database-/
```

### Tasks
Presto tasks are distributed in stages over a network of Presto workers. Tasks have inputs and outputs and are executed in parallel with a series of drivers.

### Splits
 Splits are sections of larger data sets and define how tasks operate. When Presto schedules a query, the coordinator tracks which machines are running tasks and what splits are being processed by tasks.

### Drivers and Operators
Tasks contain one or more parallel drivers and they are operators in memory. An operator consumes, transforms and produces data.

### Exchanges
Exchanges transfer data between Presto nodes for different stages of a query. Tasks produce data into an output buffer and consume data from other tasks using an exchange client.