

# Introducing Apache Impala

Impala provides fast, interactive SQL queries directly on your Apache Hadoop data stored in HDFS, HBase, or the Amazon Simple Storage Service (S3). In addition to using the same unified storage platform, Impala also uses the same metadata, SQL syntax (Hive SQL), ODBC driver, and user interface (Impala query UI in Hue) as Apache Hive. This provides a familiar and unified platform for real-time or batch-oriented queries.

Impala is an addition to tools available for querying big data. Impala does not replace the batch processing frameworks built on MapReduce such as Hive. Hive and other frameworks built on MapReduce are best suited for long running batch jobs, such as those involving batch processing of Extract, Transform, and Load (ETL) type jobs.

**Note:** Impala graduated from the Apache Incubator on November 15, 2017. In places where the documentation formerly referred to "Cloudera Impala", now the official name is "Apache Impala".

## Impala Benefits

Impala provides:

- Familiar SQL interface that data scientists and analysts already know.
- Ability to query high volumes of data ("big data") in Apache Hadoop.
- Distributed queries in a cluster environment, for convenient scaling and to make use of cost-effective commodity hardware.
- Ability to share data files between different components with no copy or export/import step; for example, to write with Pig, transform with Hive and query with Impala. Impala can read from and write to Hive tables, enabling simple data interchange using Impala for analytics on Hive-produced data.
- Single system for big data processing and analytics, so customers can avoid costly modeling and ETL just for analytics.

## How Impala Works with Apache Hadoop

The Impala solution is composed of the following components:

- Clients - Entities including Hue, ODBC clients, JDBC clients, and the Impala Shell can all interact with Impala. These interfaces are typically used to issue queries or complete administrative tasks such as connecting to Impala.
- Hive Metastore - Stores information about the data available to Impala. For example, the metastore lets Impala know what databases are available and what the structure of those databases is. As you create, drop, and alter schema objects, load data into tables, and so on through Impala SQL statements, the relevant metadata changes are automatically broadcast to all Impala nodes by the dedicated catalog service introduced in Impala 1.2.
- Impala - This process, which runs on DataNodes, coordinates and executes queries. Each instance of Impala can receive, plan, and coordinate queries from Impala clients. Queries are distributed among Impala nodes, and these nodes then act as workers, executing parallel query fragments.
- HBase and HDFS - Storage for data to be queried.

Queries executed using Impala are handled as follows:

1. User applications send SQL queries to Impala through ODBC or JDBC, which provide standardized querying interfaces. The user application may connect to any `impalad` in the cluster. This `impalad` becomes the coordinator for the query.
2. Impala parses the query and analyzes it to determine what tasks need to be performed by `impalad` instances across the cluster. Execution is planned for optimal efficiency.
3. Services such as HDFS and HBase are accessed by local `impalad` instances to provide data.
4. Each `impalad` returns data to the coordinating `impalad`, which sends these results to the client.

## Primary Impala Features

Impala provides support for:

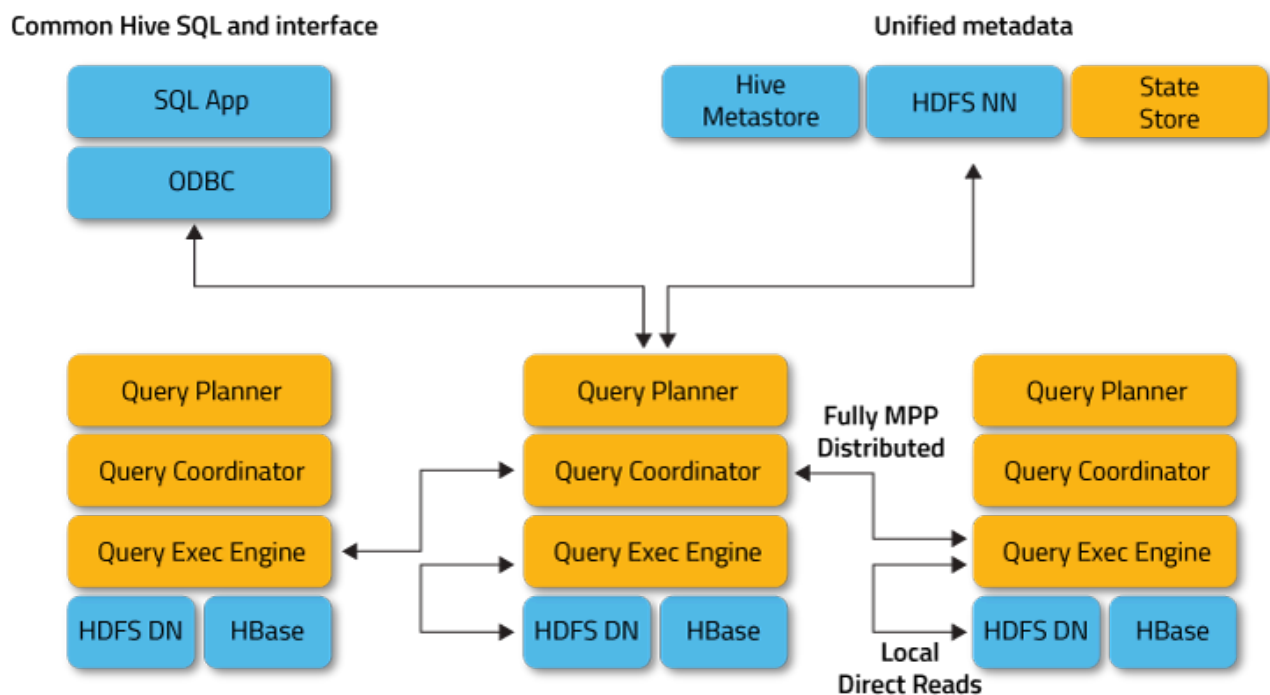
- Most common SQL-92 features of Hive Query Language (HiveQL) including [SELECT](#), [joins](#), and [aggregate functions](#).
- HDFS, HBase, and Amazon Simple Storage System (S3) storage, including:
  - [HDFS file formats](#): delimited text files, Parquet, Avro, SequenceFile, and RCFile.
  - Compression codecs: Snappy, GZIP, Deflate, BZIP.
- Common data access interfaces including:
  - [JDBC driver](#).
  - [ODBC driver](#).
  - Hue Beeswax and the Impala Query UI.
- [impala-shell command-line interface](#).
- [Kerberos authentication](#).

# Overview

Impala raises the bar for SQL query performance on Apache Hadoop while retaining a familiar user experience. With Impala, you can query data, whether stored in HDFS or Apache HBase – including SELECT, JOIN, and aggregate functions – in real time. Furthermore, Impala uses the same metadata, SQL syntax (Hive SQL), ODBC driver, and user interface (Hue Beeswax) as Apache Hive, providing a familiar and unified platform for batch-oriented or real-time queries. (For that reason, Hive users can utilize Impala with little setup overhead.)

## Architecture

To avoid latency, Impala circumvents MapReduce to directly access the data through a specialized distributed query engine that is very similar to those found in commercial parallel RDBMSs. The result is order-of-magnitude faster performance than Hive, depending on the type of query and configuration.



There are many advantages to this approach over alternative approaches for querying Hadoop data, including::

- Thanks to local processing on data nodes, network bottlenecks are avoided.
- A single, open, and unified metadata store can be utilized.
- Costly data format conversion is unnecessary and thus no overhead is incurred.
- All data is immediately query-able, with no delays for ETL.
- All hardware is utilized for Impala queries as well as for MapReduce.
- Only a single machine pool is needed to scale.