01 Hive Theory - Basics -KirkYagami

- Hive is a data warehousing tool that provides an SQL-like interface to query data stored in Hadoop.
- ETL tool for hadoop ecosystem.
- Is a SQL like querying tool to query the data stored in HDFS and other Filesystems that integrate with Hadoop.
- Developed by Facebook and later on taken by Apache.
- It processes Structured data that can be stored into Tables.
- Efficient for Batch Processing.
- Hive is a lens between MapReduce and HDFS.
- It provides us various storage file formats like Parquet, Sequence file, ORC file, Text file with significant compression.
- Provides Beeline client which is used to connect from Java, Scala, C#, Python, and many more languages.

What hive is not?

- Not a database.
- Not an OLTP tool. Closer to being OLAP tool.
- Does not provide row level insert, update and delete. (- Update is only possible for ORC format.)

WHY?

- To reduce the complexity of MapReduce Programming, Mappers, Reducers and Driver code
- 100s of lines of MR code can be summarized into one line query.
- Analysts can use SQL knowledge, No need of Java.

Hive VS SQL

Hive	SQL
Data warehouse project for data analysis, built on top of Hadoop. Does not store physical data	Query language for RDMS

Hive	SQL
Write once and Read-Many times	Write-Many and Read-Many
OLAP systems	OLTP systems
Highly scalable at lowcost	Not easily scalable

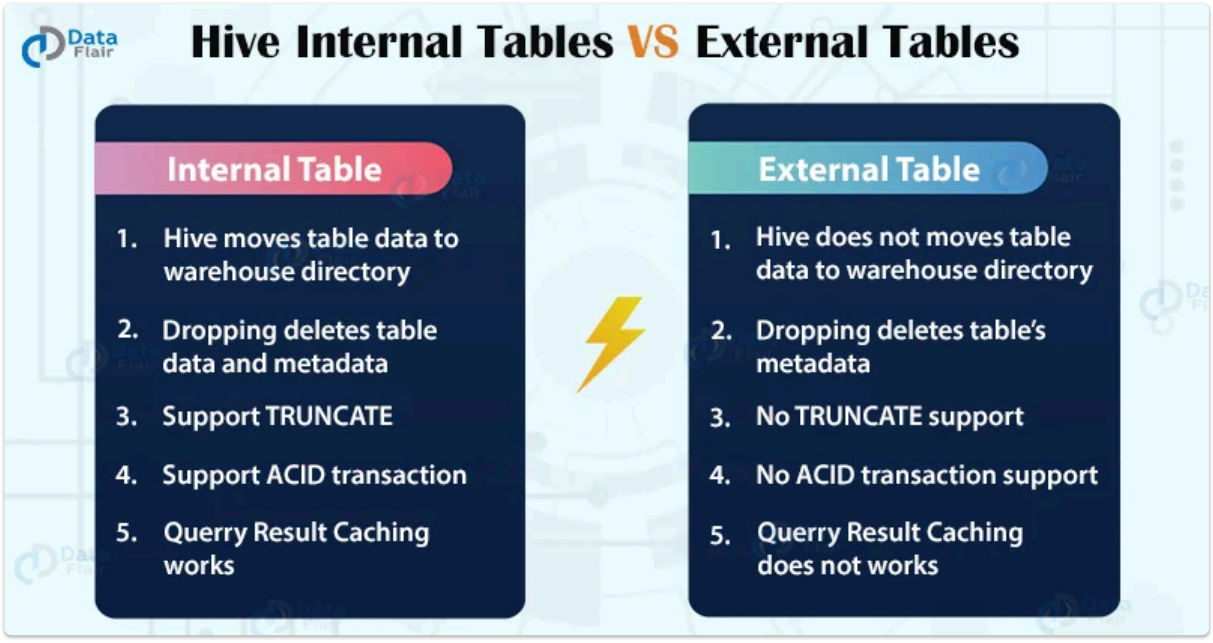
Hive VS RDBMS

Hive	RDBMS
Data warehouse on top of Hadoop. Geared towards Analytical Processing.	RDBMS are for transactional processing
Batch Processing	Transactional Processing
1. Optimal for processing large datasets of data. Gigabytes/Petabytes.	1. OLTP systems are optimal for handling a large number of short online transactions, like insert, update, and delete operations.
2. Data in Hive is meant to be used for Analytical Purposes. For example: To process orders data for trends in the last 3 years.	2. Data in RDBMS is meant to be used for real-time transaction processing.
3. Since Hive uses Hadoop and MapReduce, it leverages the tremendous parallel computing power of these frameworks.	3. RDBMS handles transactions efficiently but is not designed for massive parallel processing.
4. Hive/Hadoop uses a large number of cheap machines in parallel. Performance is linearly increased when more machines are added.	4. RDBMS improves performance by using indexing. Building indexes takes time and a lot of disk space, which can be expensive. Performance does not scale linearly with increasing disk space.
5. Not for transactional processing. Even fetching a single row will launch a MapReduce job that might take a few minutes to run.	5. A well-designed RDBMS can answer transactional queries in milliseconds or microseconds.
6. Hive does not support row-level updates.	6. RDBMS supports row-level updates and deletes.
7. Once the data is written to Hive, its purpose is read-only. Analysts would only read historical data,	7. RDBMS allows data to be frequently updated and

Hive	RDBMS
never update it.	modified.
Schema-on-Read	Schema-on-Write
Schema: Description of a database table: column names, column types, constraints.	Schema: Description of a database table: column names, column types, constraints, enforced at the time of data write.
Data of the table is stored in files in HDFS. Hive is not the owner of files. Data can be modified by other clients. For example: The same files might be shared between Hive, HBase, Cassandra. Because underlying files can be changed at any time, Hive cannot enforce Schema-on-Write. Hive metastore has the instructions for reading and parsing the HDFS files.	Complete control over data, no external program can access the data without going through the database.
During load/insert operations, Hive will just dump the data into a file without checking the schema. Since it is unaware of schema during write operations, you cannot do row-level updates/deletes.	During load/insert operations, RDBMS checks the data against the schema, ensuring data integrity and consistency.

- Hive has extensive support for joins.
- Hive does not support Natural Joins.

```
select a.firstname, b.subordinate
from employees,subordinates;
```



Deleting and updating a single row in a table

> *Hive does not provide a way to perform these two operations!*

- Hive pretends that the underlying data is in the form of a table but it is a batch processing system at heart.
- Hive has massive overheads in job submission and scheduling.
- Hive does not offer real-time queries and row-level updates.