Apache Hive

Hadoop tool for processing structured data

What is Hive?

- Simply put, Hive is basically a SQL engine thrown on top of Hadoop.

Why was Hive created?

- With Hadoop and HDFS, storing lots and lots of data. We process this with MapReduce.

- **Problem:** MapReduce is programmatic in nature.
  - Hard to understand & lacked expressiveness.

- **Solution:** Hive, compiler for SQL statements to MapReduce jobs.

What does is do for you?

- Tabular view of your raw data. (Relational familiarity).
- Create MapReduce jobs using SQL-like statements
- Scheduled on a cluster.
- Because tabular data, the data can be stored in many different ways, Cassandra or HBase tables.
- Mainstream Hadoop to non-developer community.
- Speed up Development time.

Key Building Principles

- If you are well-versed in SQL, you are well-versed in Hive.

Type System

- Primitive Types
  - Integers. TINYINT, SMALLINT, INT, BIGINT
  - Boolean. BOOLEAN
  - Floating Points. FLOAT, DOUBLE
  - Strings. STRING
- Complex Types
  - Struct: {a: Int, b: String}.
  - Maps: M[‘group’].
  - Arrays: [‘a’, ‘b’, ‘c’], A[1] returns ‘b’.

Data Model - Tables

- **Tables**
  - Similar to Tables in relational DBs.
  - Each table has directory in HDFS
    - Example: user table - ut
    - HDFS directory
      - /wh/ut
- Example:
  - CREATE TABLE user(name string, email string, state: string)

Data Model - Partitions

- Partitions
  - Similar to dense indexes.
  - Allows you to achieve rows quickly.
    - Mapped in file structure
- Example
  - partition columns: name, state
  - HDFS for name: name=Max, state=CO
    - /wh/ut/name=Max/state=CO

Hive Query Language

- Partitioning - how to create a partition.

  CREATE TABLE test_table(a string, b int)
  PARTITIONED BY (a string, b int);

- Map job
    - test_table PARTITION(a='something', b=5)
      SELECT * FROM t

- ALTER TABLE test_table ADD PARTITION (a='something', b=2)

Partitioning Cont'd

- SELECT * from test_table WHERE a='something'
  - Will only scan files within the: /user/hive/warehouse/test_table/a=something directory
- SELECT * from test_table WHERE a='something' AND b=4
  - Will only scan files within the: /user/hive/warehouse/test_table/a=something/b=4 directory

Buckets

- Buckets are modeled using hashes on columns.
- Used to parallelize partitions.
- Example:
    - HDFS file for user hash 0
        - wh/ut/name=Max/state=CO/part=00000
    - HDFS for user hash 20
        - wh/ut/name=Max/state=CO/part=00020

External Tables

- Used when you have other MapReduce jobs running that are not in Hive.

- But you want to point Hive to them.

Serialization

- Let's you create structured data from unstructured data using Hive.

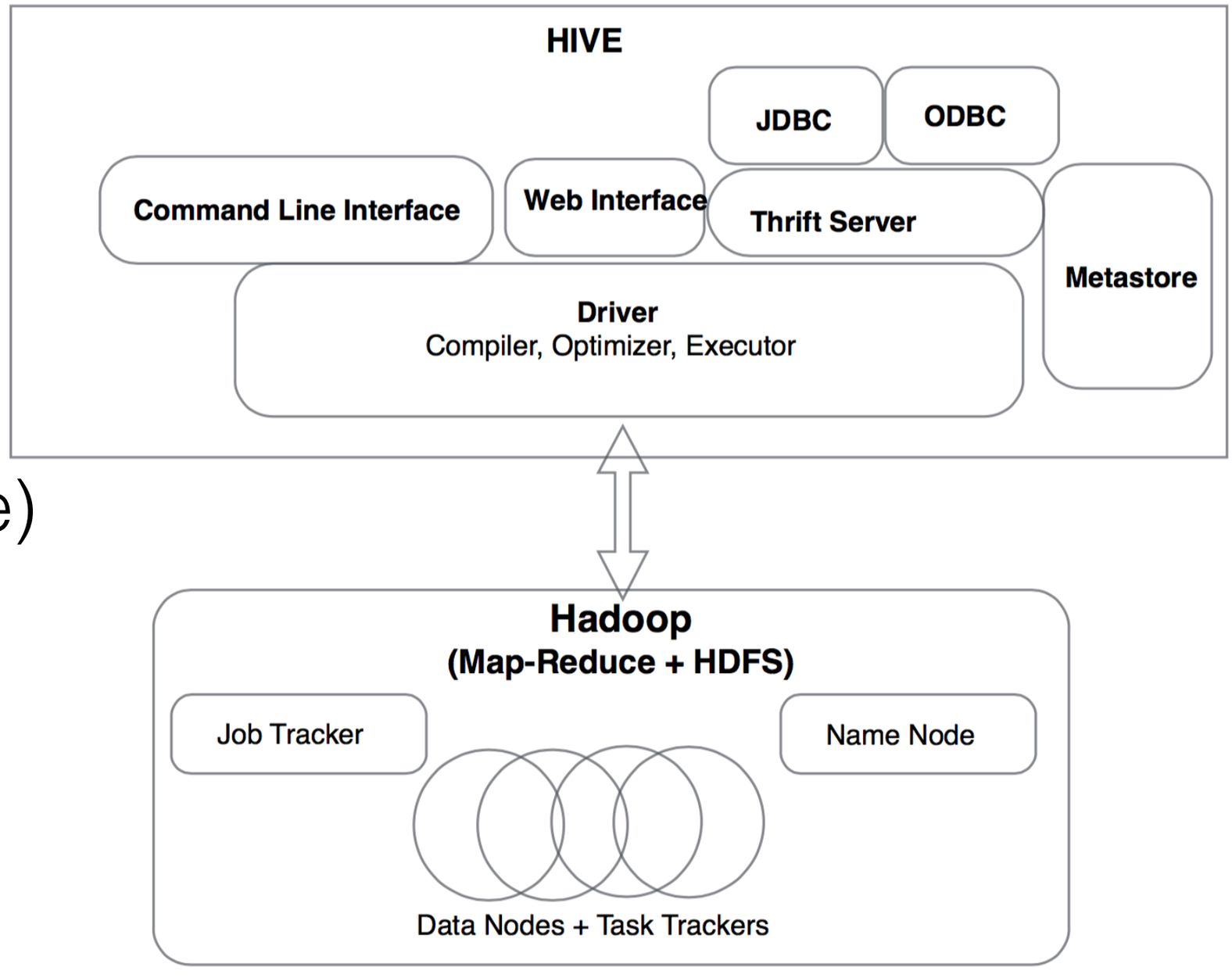
- Designed to read data from various types of delimiter separators. tsv, csv, etc…

Hive File Formats

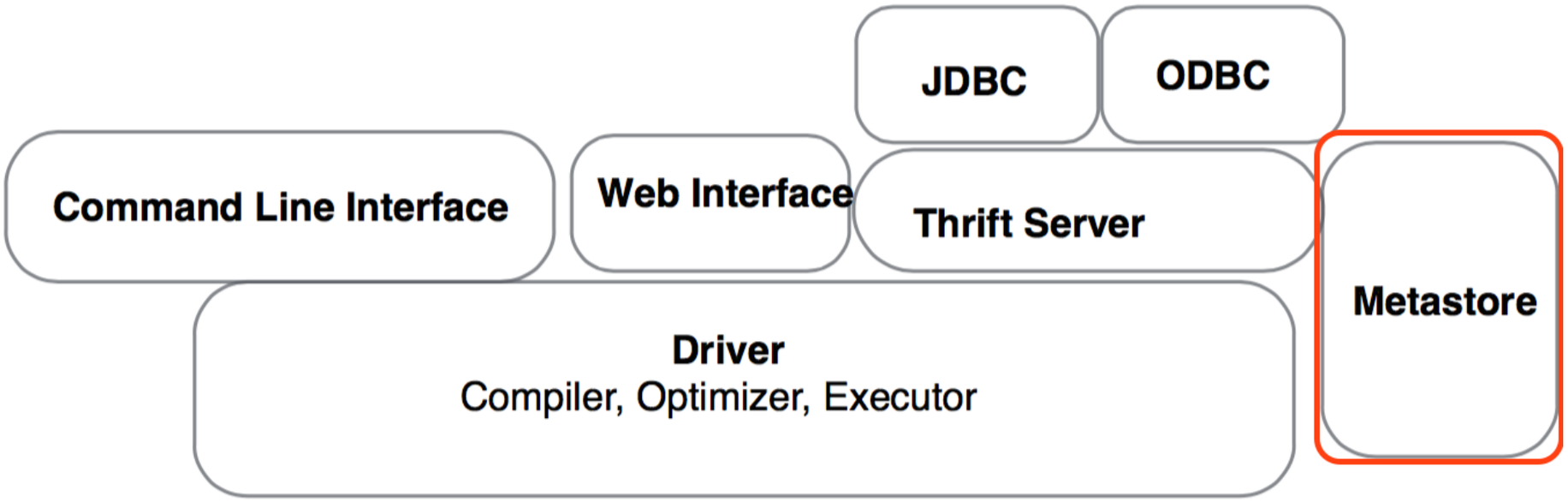
- Hive lets you store various file formats
- Example (SQL):
  - CREATE TABLE dest1(key INT, value STRING)
    STORED AS
    INPUTFORMAT
    ‘org.apache.hadoop.mapred.SequenceFileInputFormat’
    OUTPUTFORMAT
    ‘org.apache.hadoop.mapred.SequenceFileOutputFormat’

System Architecture and Components

Hive is external to Hadoop (MapReduce) cluster

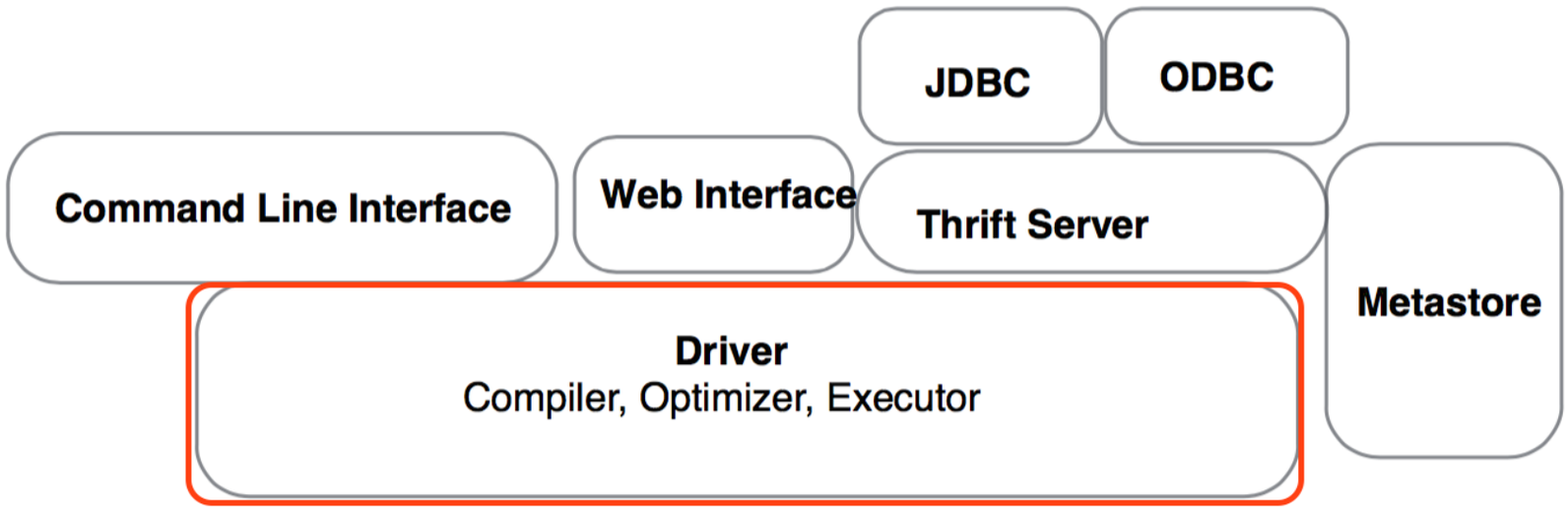


System Architecture and Components



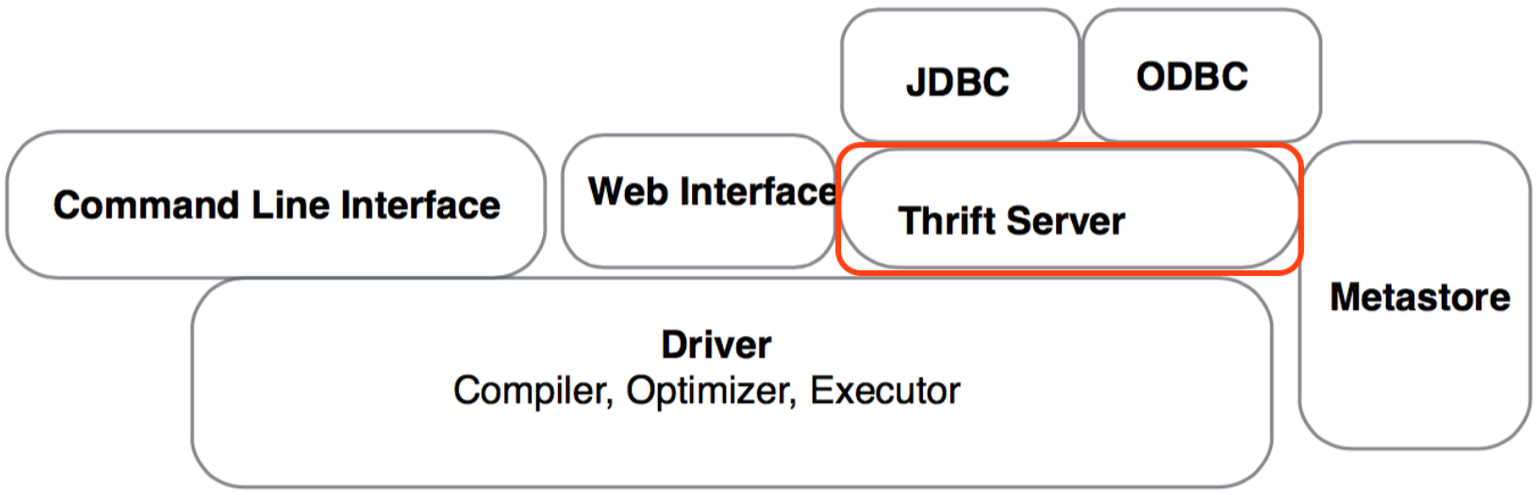
- Stores metadata about tables, columns, and partitions
- Stored on traditional RDBMS

System Architecture and Components



- Driver: Manages Lifecycle of map reduce jobs created with Hive

System Architecture and Components



- Allows you to connect other components and applications to Hive

References

- Binarylore, inc. (2014, Feb 20). *Introduction to Hive.* Retrieved from: https://www.youtube.com/watch?v=gA8_6d5Fs8Q