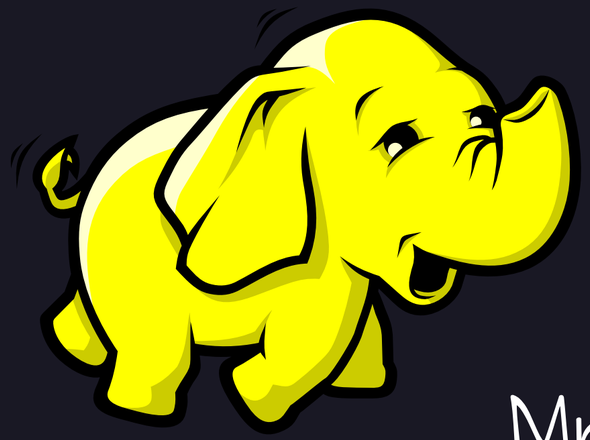
Apache Hive

- History of Hive
- What is Hive
- How & when hive can be used
- When hive cannot be used
- hive architrctutre

Présenté par : Mounir ben romdhane & Chayma hannachi

Saving Life

before

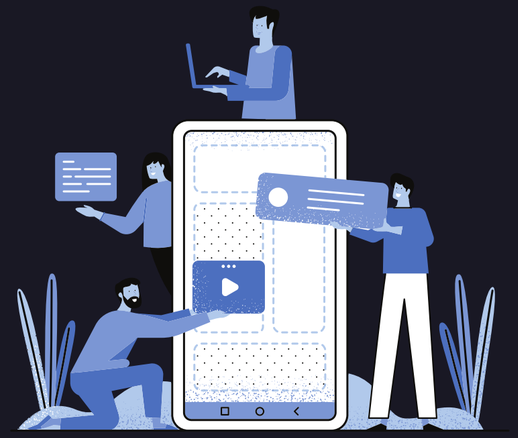


Mr hadoop

after

Ms hive

whose idea was it ??

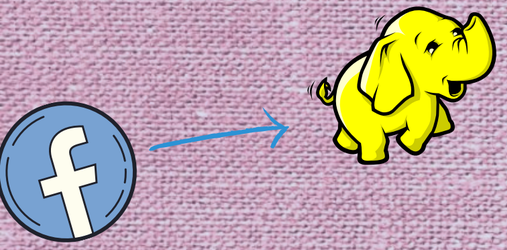


Hive was originally developed by facebook and is now maintained as apache hive by apache software foundation it is used and by biggies such as netflix and amazone as well.

History of Hive

1

FACEBOOK USED HADOOP AS A SOLUTION TO HANDEL THE GROWING BIG DATA



2

AS WE KNOW, HADOOP USES MAPREDUCE FOR PROCESSING DATA.MAPREDUCE REQUIRED USERS TO WRITE LONG CODES

3

NOT ALL USERS WERE WELL VERSED WITH JAVA AND OTHER CODING LANGUAGES.THIS PROVED TO BE DISADVANTAGE FOR THEM

4

USERS WERE COMFORTABLE WITH WRITING QUERIES IN SQL

5

HIVE WAS DEVELOPED WITH A VISION TO INCORPORATETHE CONCEPTS OF TABELS, COLUMNS JUST LIKE SQL

why Hive?

problem

FOR PROCESSING AND ANALYZING DATA USERS FOUND IT DIFFICULT TO CODE AS NOT ALL OF THEM WERE WELL VERSED WITH THE CODING LANGUAGES

solution

USERS REQUIRED A LANGUAGE SIMILARTO SQL WHICHWAS WELL KNOWN TO ALL THE USERS

solution

HIVEQL

what is hive?

- Apache Hive is a datawarehouse for Hadoop. It was created by Facebook to later become an open source Apache project. This is not a relational database or a classic data warehouse.

- If Hive is not a database or a data warehouse, then what is it?

- This is a system that maintains metadata describing data stored in HDFS. It uses a relational database called metastore (Derby by default) to ensure metadata persistence. Thus, a table in Hive is essentially composed:

- From a diagram stored in the metastore,
- Data stored in HDFS

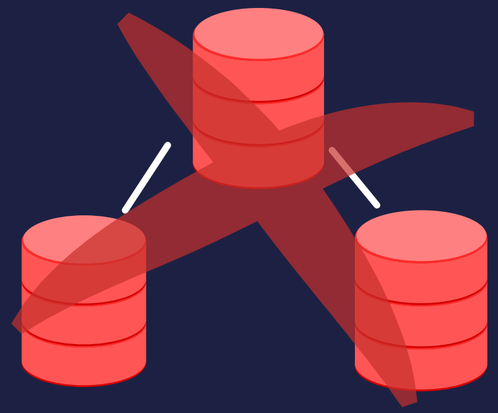
How & when hive can be used ?

Hive can be used for OLAP (online analitique processing)

it is scalable, fast and flexible.

t is a great platform for the SQL users to write SQL like queries to interact with the large datasets that reside on HDFS filesystem

Here is what Hive cannot be used for:



It is not a relational database

It cannot be used for OLTP (online transaction) processing

It cannot be used for real time updates or queries

It cannot be used for scenarios where low latency data retrieval is expected, because there is a latency in converting the HIVE scripts into MAP REDUCE scripts by Hive

Some of the finest features of hive:

1

It supports different file formats like sequence file, text file, avro file format, ORC file, RC file

2

Metadata gets stored in RDBMS like derby database

3

Hive provides lot of compression techniques, queries on the compressed data such as SNAPPY compression, gzip compression

4

Users can write SQL like queries that hive converts into mapreduce or tez or spark jobs to query against hadoop datasets

5

Users can plugin mapreduce scripts into the hive queries using UDF user defined functions

HIVE
Vs
Traditional RDBMS

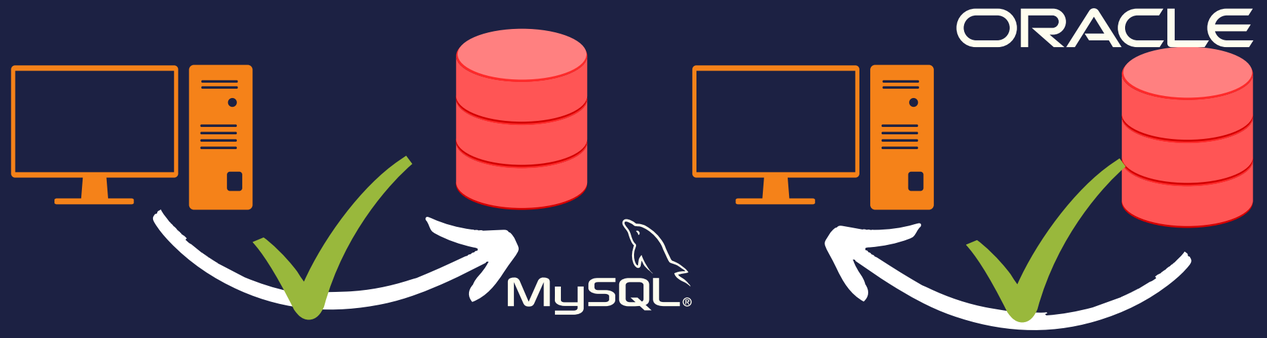
hive

- SCHEMA ON READ

HIVE ENFORCES SCHEMA ON READ SCHEMA ON READ ALLOWS THE COMPONENTS TO INSERT DATA WHITHOUT CHECKING FOR THE TYPE OR SCHEMA DEFENITION OF THE TABLE IT VERIFIES THE DATA ONLY WHEN DATA IS READ FROM THE TABLE

TRADITIONAL RDBMS

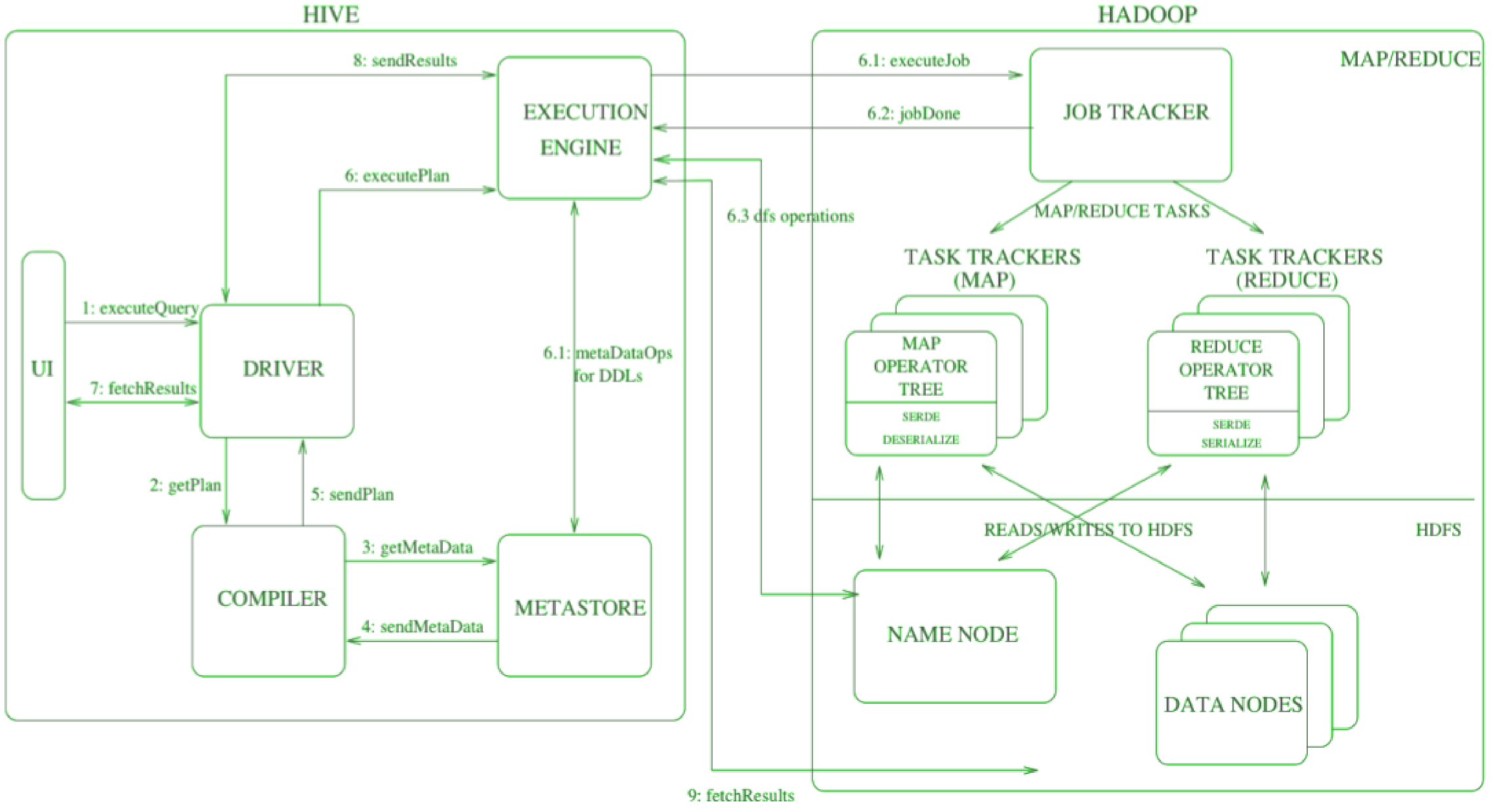
- SCHEMA ON WRITE

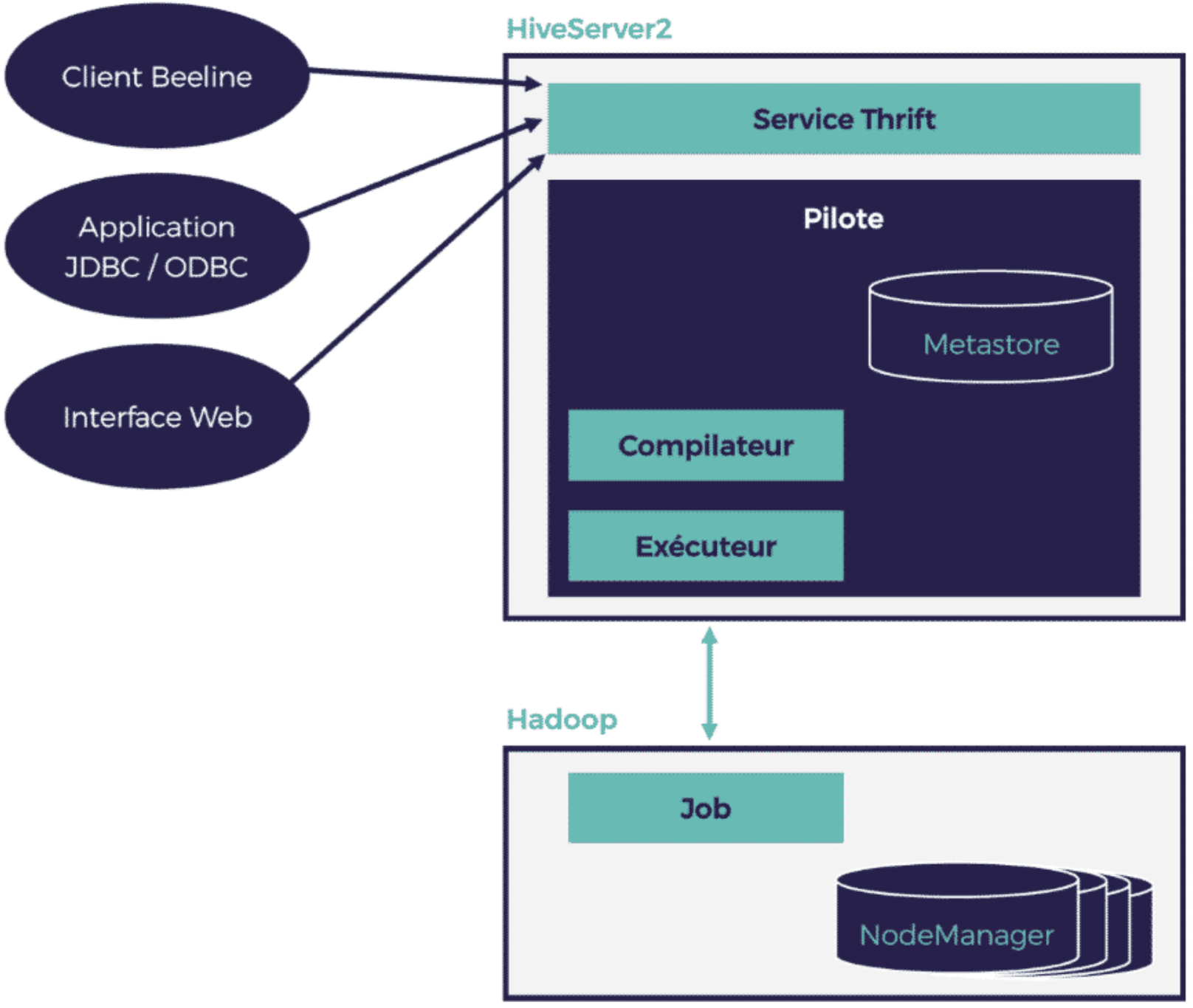


TRADITIONAL RDBMS ENFORCES SCHEMA ON WRITE. SCHEMA ON WRITE INCLUDES VERIFYING IF THE DATA INSERTED AS PER THE TABLE DEFINITION AND SCHEMA DEFINITION DURING THE WRITE PHASE ITSELF THIS IS HOW OUR DBMS DATABASES LIKE MYSQL ORACLE

hive	TRADITIONAL RDBMS
HIVE ALLOWS YOU TO STORE HUNDREDS OF PETABYTES OF DATA BECAUSE HIVE STORES DATA IN HDFS WICH HAS A SCALABLE STORAGE SPACE.	DBMS HAVE A MAC STORAGE CAPACITY AROUND 10 TERABYTES OF DATA AND QUERING SUCH LARGE DATA IS NOT AN EASY TASK.
DOESN'T SUPPORT OLTP	SUPPORTS OLTP

HIVE ARCHITECTURE


HIVE
HADOOP
MAP/REDUCE
8: sendResults
6.1: executeJob
6.2: jobDone
EXECUTION ENGINE
JOB TRACKER
6: executePlan
6.3 dfs operations
MAP/REDUCE TASKS
TASK TRACKERS (MAP)
TASK TRACKERS (REDUCE)
1: executeQuery
UI
DRIVER
7: fetchResults
6.1: metaDataOps for DDLs
MAP OPERATOR TREE
SERDE DESERIALIZE
REDUCE OPERATOR TREE
SERDE SERIALIZE
2: getPlan
5: sendPlan
3: getMetaData
COMPILER
METASTORE
4: sendMetaData
READS/WRITES TO HDFS
HDFS
NAME NODE
DATA NODES
9: fetchResults


Client Beeline
Application
JDBC / ODBC
Interface Web
HiveServer2
Service Thrift
Pilote
Metastore
Compilateur
Exécuteur
Hadoop
Job
NodeManager