

LƯU TRỮ VÀ TRUY VẤN DỮ LIỆU LỚN VỚI HIVE

Giảng viên: TS. Nguyễn Văn Quyết



- Giới thiệu chung về Hive
- Kiến trúc của Hive
- Quy trình làm việc của Hive
- Các lệnh trong Hive DDL
- Các lệnh trong Hive DML
- Thực hành với Hive
- Hỏi & đáp

What is Hive?

- Apache Hive là một hệ thống kho dữ liệu (data warehouse system) nguồn mở xây dựng trên top của Hadoop hỗ trợ cho việc truy vấn và phân tích dữ liệu lớn lưu trữ trên HDFS.
- Hive hướng đến người dùng quen thuộc với SQL
- Hive sử dụng ngôn ngữ HiveQL, tương tự SQL
- Hive chuyển các câu truy vấn SQL thành chuỗi các MapReduce Jobs chạy trên hệ thống Hadoop cluster.

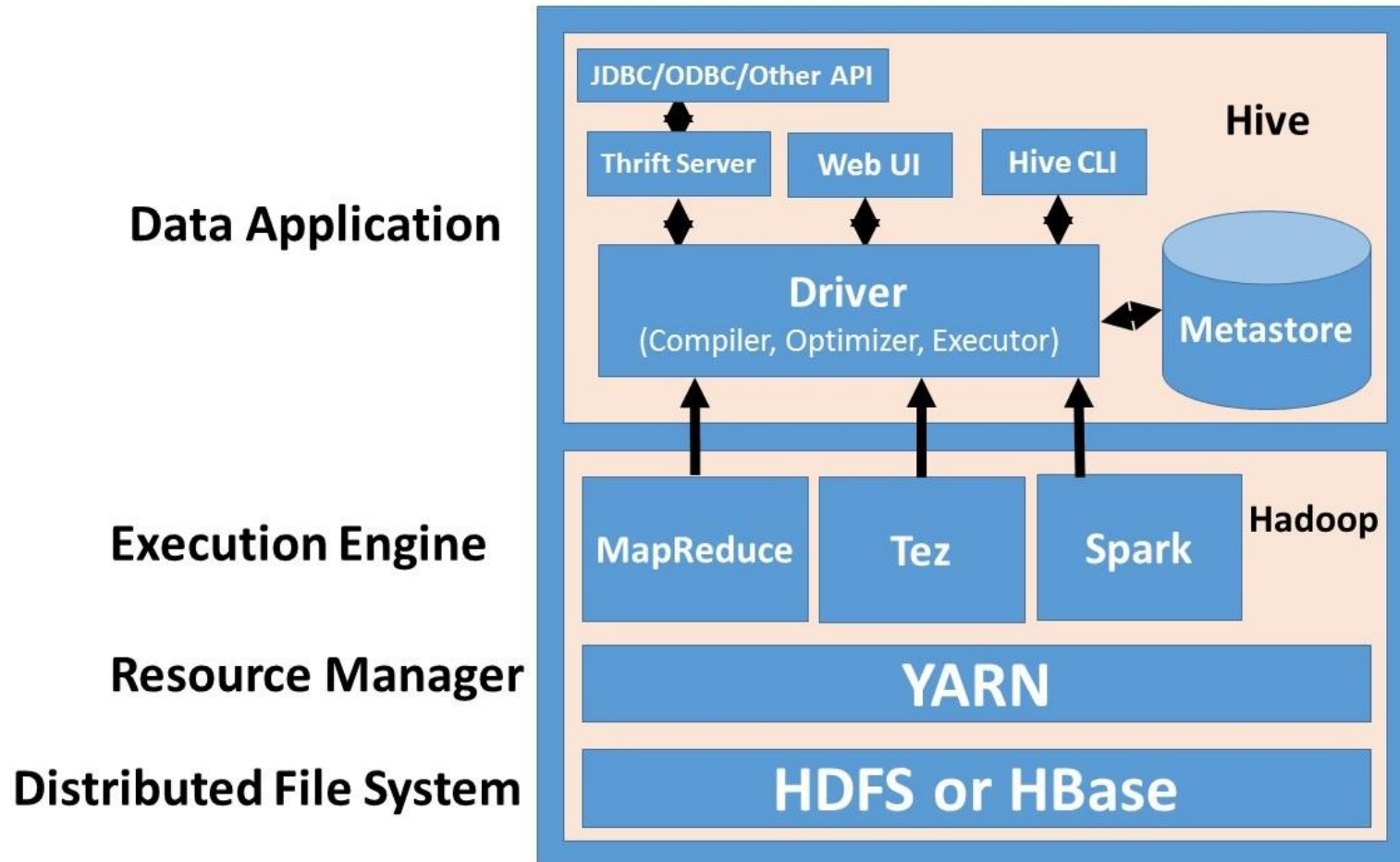
Lịch sử của Hive

- Apache Hive được phát triển bởi Data Infrastructure Team của Facebook năm 2010
- Mục đích ban đầu nhằm giải quyết các bài toán của Facebook
 - Lưu trữ 2PB raw data
 - Xử lý ~ 15TB mỗi ngày
- Hiện được sử dụng nhiều công ty lớn: Amazon, IBM, Yahoo, Netflix...

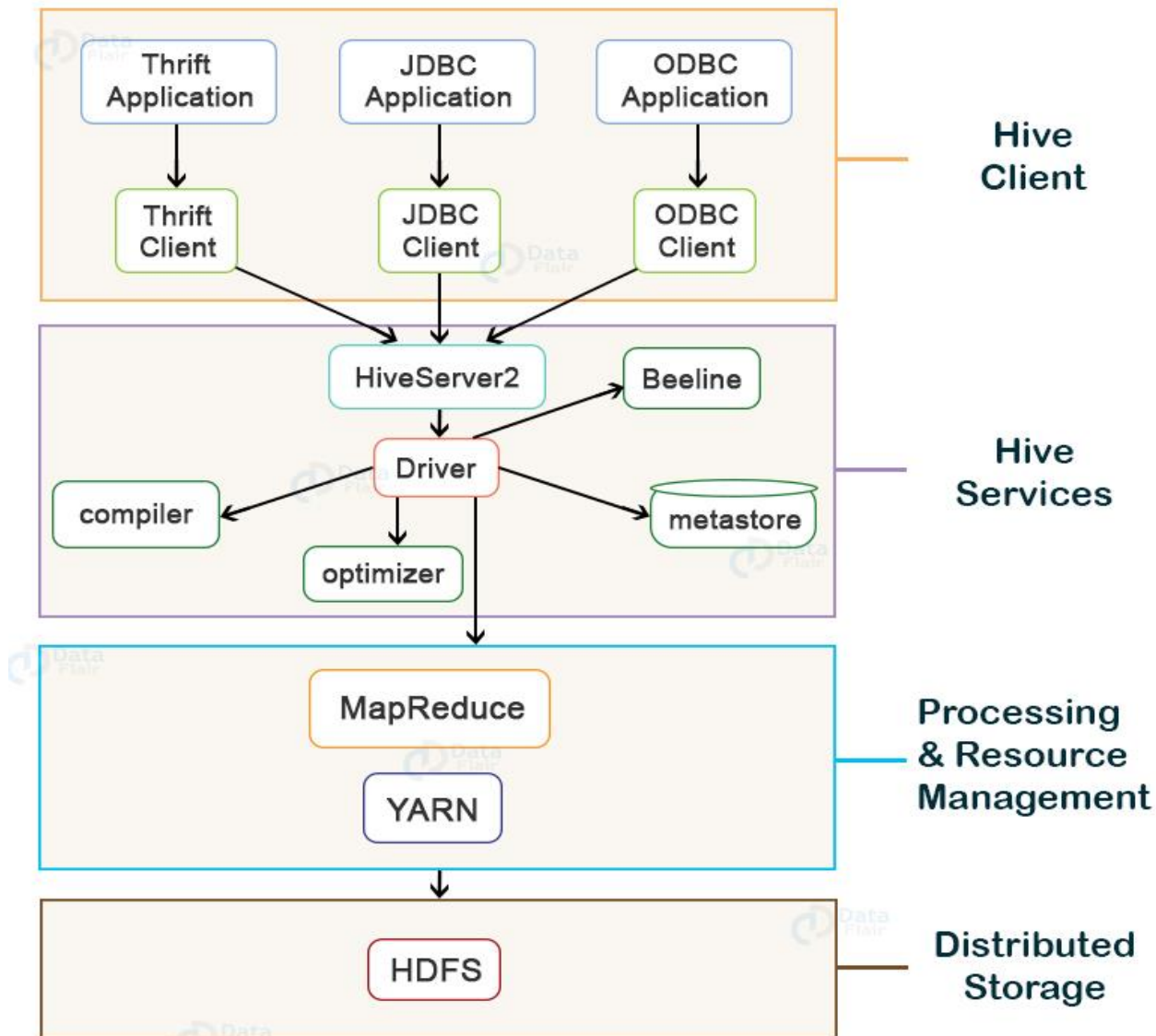
Tại sao nên sử dụng Hive?

- Cơ sở dữ liệu quan hệ không phù hợp cho việc lưu trữ và khai thác dữ liệu lớn (TH của Facebook)
- Sử dụng lập trình MapReduce đôi khi khó và phức tạp
- Hive cho phép:
 - Định cấu trúc dữ liệu linh hoạt, có thể được phân mảnh và lưu trữ trên HDFS
 - Viết lệnh SQL đơn giản
 - Hỗ trợ các trình kết nối truy cập dữ liệu khác như JDBC/ODBC
 - Dễ dàng mở rộng khả năng lưu trữ và tính toán

Hive trên hệ thống Hadoop



Kiến trúc chi tiết của Hive



Kiến trúc chi tiết của Hive

● Hive Client

- Hỗ trợ các ứng dụng viết bằng nhiều ngôn ngữ: Python, Java, C++, Ruby, etc.
- Sử dụng các trình điều khiển:
 - Thrift
 - JDBC
 - ODBC

Kiến trúc chi tiết của Hive

● Hive Service (1/2)

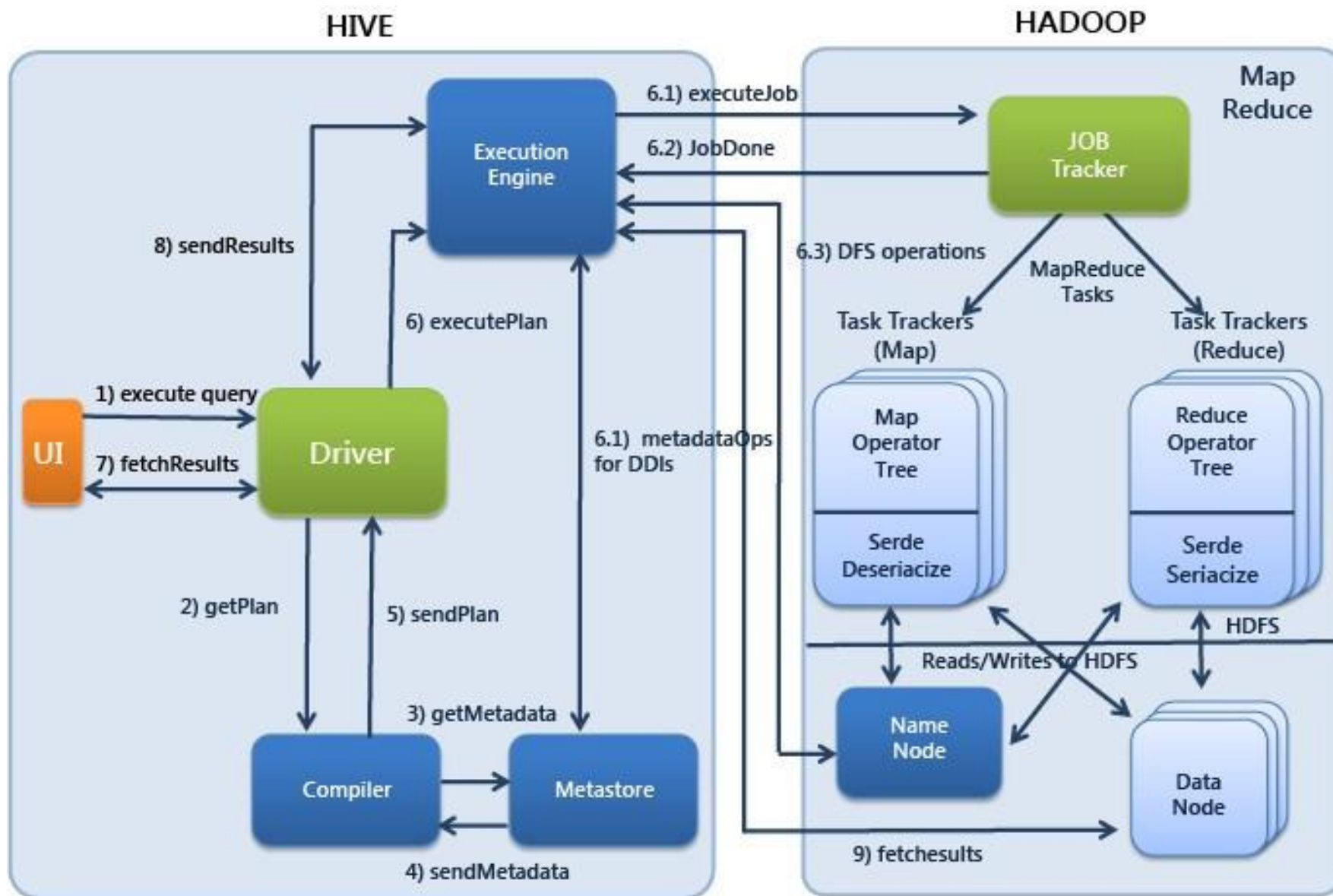
- **Beeline**: command shell hỗ trợ bởi HiveServer2, người dùng có thể submit các câu truy vấn lên hệ thống
- **Hive Server2**: cho phép Clients thực thi các câu lệnh truy vấn trên Hive. Hỗ trợ API cho các trình điều khiển JDBC và ODBC.
- **Hive Driver**: nhận các lệnh HiveQL gửi đến bởi người dùng thông qua Command Shell; tạo các session để xử lý các queries và gửi queries tới Compiler

Kiến trúc chi tiết của Hive

● Hive Service (2/2)

- **Hive Compiler:** Phân tích câu truy vấn để sinh ra các biểu thức truy vấn; tạo ra kế hoạch thực thi dưới dạng DAG (**Directed Acyclic Graph**)
- **Optimizer:** thực hiện việc chuyển đổi các thao tác trên kế hoạch thực thi để chia thành các task sao cho hiệu quả nhất
- **Execution Engine:** Thực thi các tasks theo thứ tự tạo ra bởi Compiler trên Hadoop Cluster
- **Metastore:** lưu trữ các thông tin về cấu trúc bảng và các partitions bao gồm cả thông tin từng columns. Thường lưu trữ trên CSDL quan hệ.

Các bước thực thi của Hive



Các lệnh Hive DDL

- Các lệnh Hive Data Definition Language(DDL) dùng thao tác với CSDL và bảng

Lệnh Hive DDL	Áp dụng cho
CREATE	Database, Table
SHOW	Databases, Tables, Table Properties, Partitions, Functions, Index
DESCRIBE	Database, Table, view
USE	Database
DROP	Database, Table
ALTER	Database, Table
TRUNCATE	Table

Các lệnh Hive DML

- Các lệnh Hive Data Manipulation Language (DML) dùng thao tác với dữ liệu bên trong các bảng

1. LOAD
2. SELECT
3. INSERT
4. DELETE
5. UPDATE
6. EXPORT
7. IMPORT

Chuẩn bị môi trường – Cài đặt MySQL

- **Bước 1:** Cài đặt MySQL trên Ubuntu 20.04:

```
hduser@master:~$ sudo apt-get install mysql-server
```

- **Bước 2:** Cài đặt bảo mật cho MySQL (1/2)

```
hduser@master:~$ sudo mysql_secure_installation
```

Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No: **y [Enter]**

There are three levels of password validation policy:

Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: **1 [Enter]**

Please set the password for root here.

New password: **admin@123 [Enter]**

Re-enter new password: **admin@123 [Enter]**

Do you wish to continue with the password provided?(Press y|Y for Yes, any other key for No): **y [Enter]**

Chuẩn bị môi trường – phần mềm

● **Bước 2:** Cài đặt bảo mật cho MySQL (2/2)

Remove anonymous users? (Press y|Y for Yes, any other key for No) : **y [Enter]**

Disallow root login remotely? (Press y|Y for Yes, any other key for No) : **n [Enter]**

Remove test database and access to it? (Press y|Y for Yes, any other key for No) : **n [Enter]**

Reload privilege tables now? (Press y|Y for Yes, any other key for No) : **y [Enter]**

Chuẩn bị môi trường – phần mềm

● **Bước 3:** Kiểm tra cài đặt MySQL

```
hduser@master:~$ sudo mysql
```

```
.....
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
```

```
mysql>
```


Chuẩn bị môi trường – Cài đặt MySQL

● **Bước 4:** Cấu hình MySQL cho phép truy cập từ xa

```
$ sudo gedit /etc/mysql/mysql.conf.d/mysqld.cnf
```

```
# Thêm/sửa lệnh sau về địa chỉ IP của máy cài đặt MySQL:
```

```
bind-address      = 10.0.2.195
```

```
# Khởi động lại mysql
```

```
$ sudo service mysql restart
```

Chuẩn bị môi trường – Cài đặt Hive

● **Bước 1:** Download và cài đặt Hive (273MB)

```
$ cd /usr/local
```

```
$ sudo wget http://apache.mirror.cdnetworks.com/hive/stable-2/apache-hive-2.3.8-bin.tar.gz
```

```
$ sudo tar zxvf apache-hive-2.3.8-bin.tar.gz
```

```
$ sudo mv apache-hive-2.3.8-bin hive
```

```
$ sudo chown -R hduser:hadoop /usr/local/hive
```

Chuẩn bị môi trường – Cài đặt Hive

● **Bước 2:** Thiết lập môi trường cho Hive

\$ sudo gedit ~/.bashrc

Thêm các lệnh sau vào tệp

```
export HIVE_HOME="/usr/local/hive"  
export HCAT_HOME=$HIVE_HOME/hcatalog  
export PATH=$PATH:$HIVE_HOME/bin
```

Thực hiện lệnh sau để cập nhật

\$ source ~/.bashrc

Chuẩn bị môi trường – Cài đặt Hive

● Bước 3: Cấu hình file **hive-env.sh**

```
$ cd /usr/local/hive
```

```
$ cp conf/hive-env.sh.template conf/hive-env.sh
```

```
$ sudo gedit conf/hive-env.sh
```

Thêm cấu hình đường dẫn tới Hadoop vào tệp

```
export HADOOP_HOME="/usr/local/hadoop"
```

Chuẩn bị môi trường – Cài đặt Hive

● Bước 4: Cấu hình file **hive-site.xml** (1/3)

hduser@master:/usr/local/hive\$ sudo gedit conf/hive-site.xml

Thêm cấu hình sau vào tệp

```
<configuration>
  <property>
    <name>javax.jdo.option.ConnectionURL</name>
    <value>jdbc:mysql://10.0.2.195/hivedb</value>
  </property>
  <property>
    <name>javax.jdo.option.ConnectionDriverName</name>
    <value>com.mysql.jdbc.Driver</value>
  </property>
```

Chuẩn bị môi trường – Cài đặt Hive

● **Bước 4:** Cấu hình file **hive-site.xml** (2/3)

Thêm cấu hình sau vào tệp (tiếp)

```
<property>
  <name>javax.jdo.option.ConnectionUserName</name>
  <value>hivedb_user</value>
</property>
<property>
  <name>javax.jdo.option.ConnectionPassword</name>
  <value>Hive@123</value>
</property>
<property>
  <name>datanucleus.autoCreateSchema</name>
  <value>false</value>
</property>
```

Chuẩn bị môi trường – Cài đặt Hive

● Bước 4: Cấu hình file **hive-site.xml** (3/3)

Thêm cấu hình sau vào tệp (tiếp)

```
<property>
    <name>datanucleus.fixedDatastore</name>
    <value>true</value>
</property>
<property>
    <name>datanucleus.autoStartMechanism</name>
    <value>SchemaTable</value>
</property>
<property>
    <name>hive.metastore.warehouse.dir</name>
    <value>/hive/warehouse</value>
</property>
</configuration>
```

Chuẩn bị môi trường – Cài đặt Hive

- **Bước 5:** Tạo Database và tài khoản trong MySQL theo thông tin cấu hình Bước 4

```
$ sudo mysql
```

```
mysql> CREATE DATABASE hivedb;
```

```
mysql> CREATE USER 'hivedb_user'@'master' IDENTIFIED BY 'Hive@123';
```

```
mysql> GRANT ALL PRIVILEGES ON hivedb.* TO 'hivedb_user'@'master'  
WITH GRANT OPTION;
```

```
mysql> FLUSH PRIVILEGES;
```

```
mysql> use mysql
```

```
mysql> update user set host='%' where user= 'hivedb_user';
```

Xác thực tài khoản:

```
$ mysql -u hivedb_user -p -h master
```

Enter password: **Hive@123**

Chuẩn bị môi trường – Cài đặt Hive

● **Bước 6:** Chuẩn bị thư viện mysql-connector-java

```
$ cd /usr/local/hive
```

```
$ wget https://dev.mysql.com/get/Downloads/Connector-J/mysql-connector-java-8.0.23.tar.gz
```

```
$ tar xvf mysql-connector-java-8.0.23.tar.gz
```

```
$ cd mysql-connector-java-8.0.23
```

```
$ cp mysql-connector-java-8.0.23.jar /usr/local/hive/lib/
```

Chuẩn bị môi trường – Cài đặt Hive

● **Bước 7:** Kiểm tra và cập nhật thư viện **guava**

```
$ sudo rm $HIVE_HOME/lib/guava-14.0.1.jar
```

```
$ sudo cp /usr/local/hadoop/share/hadoop/hdfs/lib/guava-27.0-jre.jar  
/usr/local/hive/lib/
```

Chuẩn bị môi trường – Cài đặt Hive

● Bước 8: Khởi chạy Hive

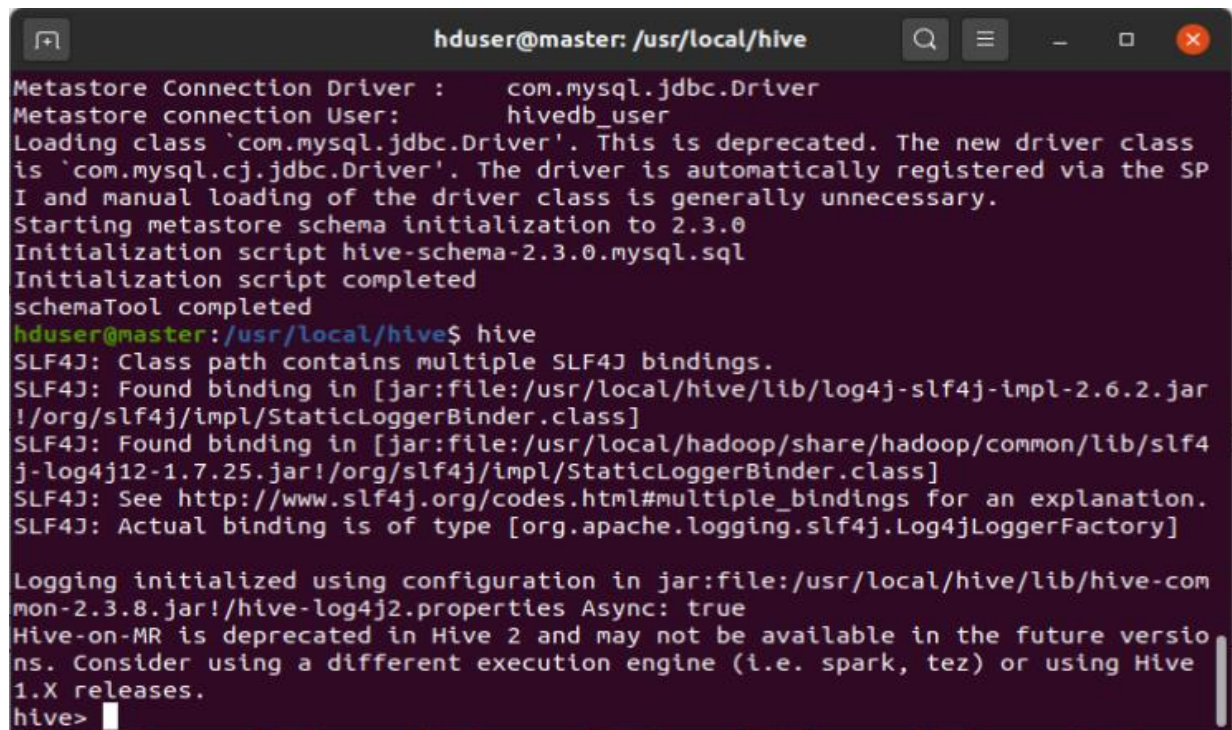
Khởi tạo **hivedb**

\$ cd /usr/local/hive

\$ bin/schematool -initSchema -dbType mysql

Chạy HiveCLI

\$ hive



```
hduser@master: /usr/local/hive
Metastore Connection Driver :    com.mysql.jdbc.Driver
Metastore connection User:      hivedb_user
Loading class 'com.mysql.jdbc.Driver'. This is deprecated. The new driver class
is 'com.mysql.cj.jdbc.Driver'. The driver is automatically registered via the SP
I and manual loading of the driver class is generally unnecessary.
Starting metastore schema initialization to 2.3.0
Initialization script hive-schema-2.3.0.mysql.sql
Initialization script completed
schemaTool completed
hduser@master:/usr/local/hive$ hive
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/local/hive/lib/log4j-slf4j-impl-2.6.2.jar
!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/local/hadoop/share/hadoop/common/lib/slf4
j-log4j12-1.7.25.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]

Logging initialized using configuration in jar:file:/usr/local/hive/lib/hive-com
mon-2.3.8.jar!/hive-log4j2.properties Async: true
Hive-on-MR is deprecated in Hive 2 and may not be available in the future versio
ns. Consider using a different execution engine (i.e. spark, tez) or using Hive
1.X releases.
hive>
```

Thực hành với Hive DDL

● Tạo và sử dụng cơ sở dữ liệu \$ hive

hive > **CREATE DATABASE** eshop;

hive > **CREATE DATABASE** agri;

Browsing HDFS

master:9870/explorer.html#/hive/warehouse

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities

Browse Directory

/hive/warehouse Go!

Show 25 entries

Search:

Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
drwxr-xr-x	hduser	supergroup	0 B	Apr 10 08:39	0	0 B	agri.db
drwxr-xr-x	hduser	supergroup	0 B	Apr 10 08:37	0	0 B	eshop.db

Showing 1 to 2 of 2 entries

Previous 1 Next

Thực hành với Hive DDL

● Tạo và sử dụng cơ sở dữ liệu

hive > **SHOW DATABASES;**

```
hive> SHOW DATABASES;  
OK  
agri  
default  
eshop  
Time taken: 0.092 seconds, Fetched: 3 row(s)  
hive> 
```

Thực hành với Hive DDL

● Tạo và sử dụng cơ sở dữ liệu

hive > DESCRIBE DATABASE eshop;

```
hive> DESCRIBE DATABASE eshop;  
OK  
eshop          hdfs://master:9000/hive/warehouse/eshop.db      hduser  USER  
Time taken: 0.066 seconds, Fetched: 1 row(s)  
hive>
```

Thực hành với Hive DDL

● Tạo và sử dụng cơ sở dữ liệu

hive > DROP DATABASE agri;

```
hive> DROP DATABASE agri;  
OK  
Time taken: 0.152 seconds  
hive>
```

Thực hành với Hive DDL

● Tạo và sử dụng cơ sở dữ liệu

hive > **USE** eshop;

```
hive> USE eshop;  
OK  
Time taken: 0.059 seconds  
hive> 
```


Thực hành với Hive DDL

● Tạo bảng cơ sở dữ liệu trong Hive

hive > **CREATE TABLE IF NOT EXISTS** Categories(CategoryID integer,
CategoryName string, Description string)

> **COMMENT** 'Loai san pham'

> **ROW FORMAT DELIMITED**

> **FIELDS TERMINATED BY** ','

> **STORED AS TEXTFILE;**

```
hive> CREATE TABLE IF NOT EXISTS Categories(CategoryID integer, CategoryName
string, Description string)
  > COMMENT 'Loai san pham'
  > ROW FORMAT DELIMITED
  > FIELDS TERMINATED BY ','
  > STORED AS TEXTFILE;
OK
Time taken: 0.995 seconds
hive> █
```

Thực hành với Hive DDL

● Tạo bảng cơ sở dữ liệu trong Hive

hive > **CREATE TABLE IF NOT EXISTS** Products(ProductID integer, CategoryID integer, ProductName string, UnitPrice integer, Quantity integer)

> **COMMENT** 'San pham'

> **ROW FORMAT DELIMITED**

> **FIELDS TERMINATED BY** ','

> **STORED AS TEXTFILE;**

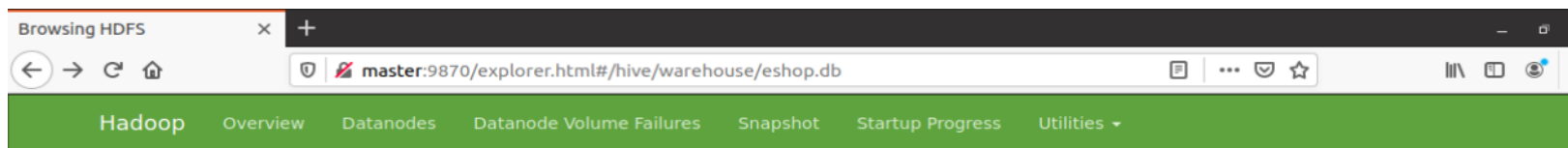
```
hive> CREATE TABLE IF NOT EXISTS Products(ProductID integer, CategoryID integer, ProductName string, UnitPrice integer, Quantity integer)
> COMMENT 'Bang san pham'
> ROW FORMAT DELIMITED
> FIELDS TERMINATED BY ','
> STORED AS TEXTFILE;
OK
Time taken: 0.182 seconds
hive>
```

Thực hành với Hive DDL

● Hive DDL trên bảng

hive > **SHOW TABLES;**

```
hive> SHOW TABLES;
OK
categories
products
Time taken: 0.111 seconds, Fetched: 2 row(s)
hive> 
```



Browse Directory

/hive/warehouse/eshop.db Go!

Show entries Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	<input type="checkbox"/>
<input type="checkbox"/>	drwxr-xr-x	hduser	supergroup	0 B	Apr 10 08:50	0	0 B	categories	<input type="checkbox"/>
<input type="checkbox"/>	drwxr-xr-x	hduser	supergroup	0 B	Apr 10 08:56	0	0 B	products	<input type="checkbox"/>

Showing 1 to 2 of 2 entries Previous **1** Next

Thực hành với Hive DDL

● Hive DDL trên bảng

hive > DESCRIBE Products;

```
hive> DESCRIBE products;  
OK  
productid          int  
categoryid         int  
productname        string  
unitprice          int  
quantity           int  
Time taken: 0.182 seconds, Fetched: 5 row(s)  
hive> █
```

Thực hành với Hive DDL

● ALTER TABLE

hive > **ALTER TABLE** Cities **ADD COLUMNS** (Description string);

```
hive> CREATE TABLE IF NOT EXISTS Cities(CityID int, CityName string)
> COMMENT 'Bang tinh thanh'
> ROW FORMAT DELIMITED
> FIELDS TERMINATED BY ','
> STORED AS TEXTFILE;
OK
Time taken: 0.191 seconds
hive> 
```

```
hive> ALTER TABLE Cities ADD COLUMNS (Description string);
OK
Time taken: 0.214 seconds
hive> DESCRIBE Cities;
OK
cityid                int
cityname              string
description           string
Time taken: 0.085 seconds, Fetched: 3 row(s)
hive> 
```

Thực hành với Hive DDL

● ALTER TABLE

hive > ALTER TABLE Cities RENAME TO Locations

```
hive> ALTER TABLE Cities RENAME TO Locations;
OK
Time taken: 0.242 seconds
hive> SHOW TABLES;
OK
categories
locations
products
Time taken: 0.071 seconds, Fetched: 3 row(s)
hive> █
```

Thực hành với Hive DDL

● Hive DDL trên bảng

hive > DROP TABLE Cites;

```
hive> DROP TABLE Cites;  
OK  
Time taken: 0.466 seconds  
hive>
```

Thực hành với Hive DML

● LOAD

hive > **LOAD DATA LOCAL INPATH** '/home/hduser/products.csv'
OVERWRITE INTO TABLE Products;

```
hive> LOAD DATA LOCAL INPATH '/home/hduser/products.csv' OVERWRITE INTO TABLE Products;
Loading data to table eshop.products
OK
Time taken: 2.374 seconds
hive>
```

products.csv

```
1,1,Samsung Galaxy S8,7500000,50
2,1,Samsung Galaxy S9,1050000,75
3,1,iPhone 8,18000000,20
4,1,Xiaomi Redmi 8,4990000,100
5,1,Xiaomi Note 8,5990000,100
6,2,LG SMART TV 4K 43INCH,8490000,15
7,2,Smart TV Samsung 49inch,7780000,20
8,2,Smart TV TCL 55inch,5950000,30
9,3,Acer TravelMate X,25800000,16
10,3,MacBook Air 2020,28000000,12
```


Thực hành với Hive DML

● SELECT

hive > **SELECT** * **FROM** Products;

```
hive> SELECT * FROM Products;
OK
1      1      Samsung Galaxy S8      7500000  50
2      1      Samsung Galaxy S9      1050000  75
3      1      iPhone 8      18000000  20
4      1      Xiaomi Redmi 8  4990000  100
5      1      Xiaomi Note 8  5990000  100
6      2      LG SMART TV 4K 43INCH  8490000  15
7      2      Smart TV Samsung 49inch  7780000  20
8      2      Smart TV TCL 55inch  5950000  30
9      3      Acer TravelMate X      25800000  16
10     3      MacBook Air 2020      28000000  12
Time taken: 0.353 seconds, Fetched: 10 row(s)
hive> 
```

Thực hành với Hive DML

● SELECT

hive > **SELECT COUNT(*) FROM Products;**

```
hive> SELECT COUNT(*) FROM Products;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions.
Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
Query ID = hduser_20210410093044_a7267efe-4bf5-466e-a8da-e8671d2be462
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1618018248314_0001, Tracking URL = http://master:8088/proxy/application_1618018248314_0001/
Kill Command = /usr/local/hadoop/bin/hadoop job -kill job_1618018248314_0001
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2021-04-10 09:31:17,495 Stage-1 map = 0%, reduce = 0%
2021-04-10 09:31:33,524 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 2.96 sec
2021-04-10 09:31:57,771 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 6.95 sec
MapReduce Total cumulative CPU time: 6 seconds 950 msec
Ended Job = job_1618018248314_0001
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 6.95 sec HDFS Read: 8439 HDFS Write: 102
SUCCESS
Total MapReduce CPU Time Spent: 6 seconds 950 msec
OK
10
Time taken: 74.672 seconds, Fetched: 1 row(s)
hive> █
```

Thực hành với Hive DML

● INSERT .. VALUES

hive > **INSERT INTO TABLE** Categories **VALUES** (1,'Mobile','Dien thoai');

```
hive> INSERT INTO TABLE Categories VALUES(1,'Mobile','Dien thoai');
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions.
Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X releases.
Query ID = hduser_20210410093856_70a20f96-5be2-4964-b469-a8367eb276e4
Total jobs = 3
Launching Job 1 out of 3
Number of reduce tasks is set to 0 since there's no reduce operator
Starting Job = job_1618018248314_0002, Tracking URL = http://master:8088/proxy/application_1618018248314_0002/
Kill Command = /usr/local/hadoop/bin/hadoop job -kill job_1618018248314_0002
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 0
2021-04-10 09:39:19,561 Stage-1 map = 0%, reduce = 0%
2021-04-10 09:39:37,601 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 4.62 sec
MapReduce Total cumulative CPU time: 4 seconds 620 msec
Ended Job = job_1618018248314_0002
Stage-4 is selected by condition resolver.
Stage-3 is filtered out by condition resolver.
Stage-5 is filtered out by condition resolver.
Moving data to directory hdfs://master:9000/hive/warehouse/eshop.db/categories/.hive-staging_hive_2021-04-10_09-38-56_761_2417518671912221982-1/-ext-10000
Loading data to table eshop.categories
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Cumulative CPU: 4.62 sec HDFS Read: 4481 HDFS Write: 92 SUCCESS
Total MapReduce CPU Time Spent: 4 seconds 620 msec
OK
Time taken: 43.488 seconds
```

Thực hành với Hive DML

● INSERT .. VALUES

hive > INSERT INTO TABLE Categories VALUES (2,'TV','Ti vi');

hive > INSERT INTO TABLE Categories VALUES (3,'Computer','May tinh');

INSERT INTO .. SELECT .. FROM

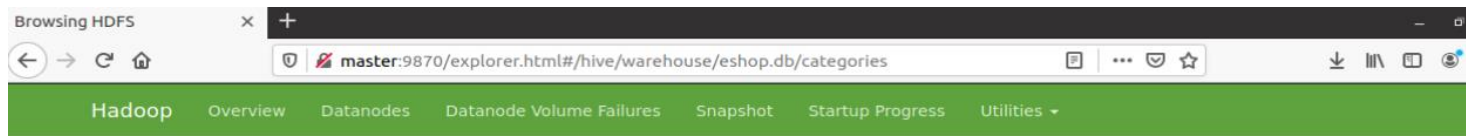
INSERT OVERWRITE .. SELECT .. FROM

Thực hành với Hive DML

● INSERT .. VALUES

hive > SELECT * FROM Categories;

```
hive> SELECT * FROM Categories;
OK
1      Mobile  Điện thoại
2      TV      Tivi
3      Computer      Máy tính
Time taken: 0.199 seconds, Fetched: 3 row(s)
hive>
```



Browse Directory

/hive/warehouse/eshop.db/categories

Show entries

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name	<input type="checkbox"/>
<input type="checkbox"/>	-rwxr-xr-x	hduser	supergroup	20 B	Apr 10 09:39	2	128 MB	000000_0	<input type="checkbox"/>
<input type="checkbox"/>	-rwxr-xr-x	hduser	supergroup	10 B	Apr 10 09:40	2	128 MB	000000_0_copy_1	<input type="checkbox"/>
<input type="checkbox"/>	-rwxr-xr-x	hduser	supergroup	20 B	Apr 10 09:43	2	128 MB	000000_0_copy_2	<input type="checkbox"/>

Showing 1 to 3 of 3 entries

Thực hành với Hive DML

● UPDATE

*Mặc định Hive không enable việc Update hay Delete, cần bổ sung các thuộc tính dưới đây vào **hive-site.xml** hoặc thiết lập bằng **Command-Line***

```
hive> Set hive.support.concurrency = true
hive> Set hive.enforce.bucketing = true
hive> set hive.exec.dynamic.partition.mode = nonstrict
hive> set hive.txn.manager =
org.apache.hadoop.hive.ql.lockmgr.DbTxnManager
hive> set hive.compactor.initiator.on = true
hive> set hive.compactor.worker.threads = 1
```

```
hive > UPDATE Products SET ProductName = 'Iphone X' WHERE ProductID
= '3';
```

Thực hành với Hive DML

● DELETE

```
hive > DELETE FROM Products WHERE ProductID = '3';
```

Thực hành với Hive DML

EXPORT

hive > **EXPORT TABLE** Products **TO** **'/hdfs_export_products'**

Browsing HDFS

master:9870/explorer.html#/hdfs_export_products/data

Hadoop Overview Datanodes Datanode Volume Failures Snapshot Startup Progress Utilities

Browse Directory

/hdfs_export_products/data Go!

Show 25 entries Search:

<input type="checkbox"/>	Permission	Owner	Group	Size	Last Modified	Replication	Block Size	Name
<input type="checkbox"/>	-rwxr-xr-x	hduser	supergroup	344 B	Apr 10 10:21	2	128 MB	products.csv

Showing 1 to 1 of 1 entries

Previous 1 Next

Thực hành với Hive DML

● IMPORT

```
hive> CREATE TABLE IF NOT EXISTS TopProducts(ProductID integer,  
CategoryID integer, ProductName string, UnitPrice integer, Quantity  
integer)
```

```
> COMMENT 'San pham'
```

```
> ROW FORMAT DELIMITED
```

```
> FIELDS TERMINATED BY ','
```

```
> STORED AS TEXTFILE;
```

```
hive > IMPORT TABLE TopProducts FROM '/hdfs_export_products'
```

Thực hành với Hive DML

● IMPORT

```
hive> IMPORT TABLE TopProducts FROM '/hdfs_export_products';
Copying data from hdfs://master:9000/hdfs_export_products/data
Copying file: hdfs://master:9000/hdfs_export_products/data/products.csv
Loading data to table eshop.topproducts
OK
Time taken: 0.415 seconds
hive> SELECT * FROM TopProducts;
OK
1      1      Samsung Galaxy S8      7500000  50
2      1      Samsung Galaxy S9      1050000  75
3      1      iPhone 8      18000000  20
4      1      Xiaomi Redmi 8  4990000  100
5      1      Xiaomi Note 8  5990000  100
6      2      LG SMART TV 4K 43INCH  8490000  15
7      2      Smart TV Samsung 49inch 7780000  20
8      2      Smart TV TCL 55inch  5950000  30
9      3      Acer TravelMate X      25800000  16
10     3      MacBook Air 2020      28000000  12
Time taken: 0.214 seconds, Fetched: 10 row(s)
hive> 
```

Thực hành với Hive DML

● JOIN

hive > **SELECT** p.* **FROM** Products p **INNER JOIN** Categories c **ON**
p.CategoryID = c.CategoryID **WHERE** c.CategoryName = 'TV';

```
hive> SELECT p.* FROM Products p INNER JOIN Categories c ON p.CategoryID = c.CategoryID
WHERE c.CategoryName = 'TV';
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future vers
ions. Consider using a different execution engine (i.e. spark, tez) or using Hive 1.X r
eleases.
Query ID = hduser_20210410100106_e58366b3-3fab-466b-989e-d42164cca005
Total jobs = 1
```

```
Total MapReduce CPU Time Spent: 4 seconds 700 msec
OK
6      2      LG SMART TV 4K 43INCH      8490000 15
7      2      Smart TV Samsung 49inch 7780000 20
8      2      Smart TV TCL 55inch      5950000 30
Time taken: 62.14 seconds, Fetched: 3 row(s)
hive> █
```

Trân trọng cảm ơn!
Q&A