

ISIT312 Big Data Management

Hive

Dr Fenghui Ren

School of Computing and Information Technology -
University of Wollongong

Hive

Outline

[Hive ? What is it ?](#)

[Deployment and configuration](#)

[Metastore](#)

[Interfaces](#)

[HQL](#)

[Hive versus relational DBMSs](#)

Hive ? What is it ?

Hive is a software system that provides tabular view of data stored in **HDFS** and **SQL**-like methods for manipulating data in **HDFS**

Apache **Hive** project started at **Facebook** in 2010 to provide a high-level interface to **HDFS**

Contrary to **Pig**, **Hive** provides **SQL**-like abstractions on top of **MapReduce**

A language called **HQL** (Hive Query Language) implements **SQL-92** standard (almost)

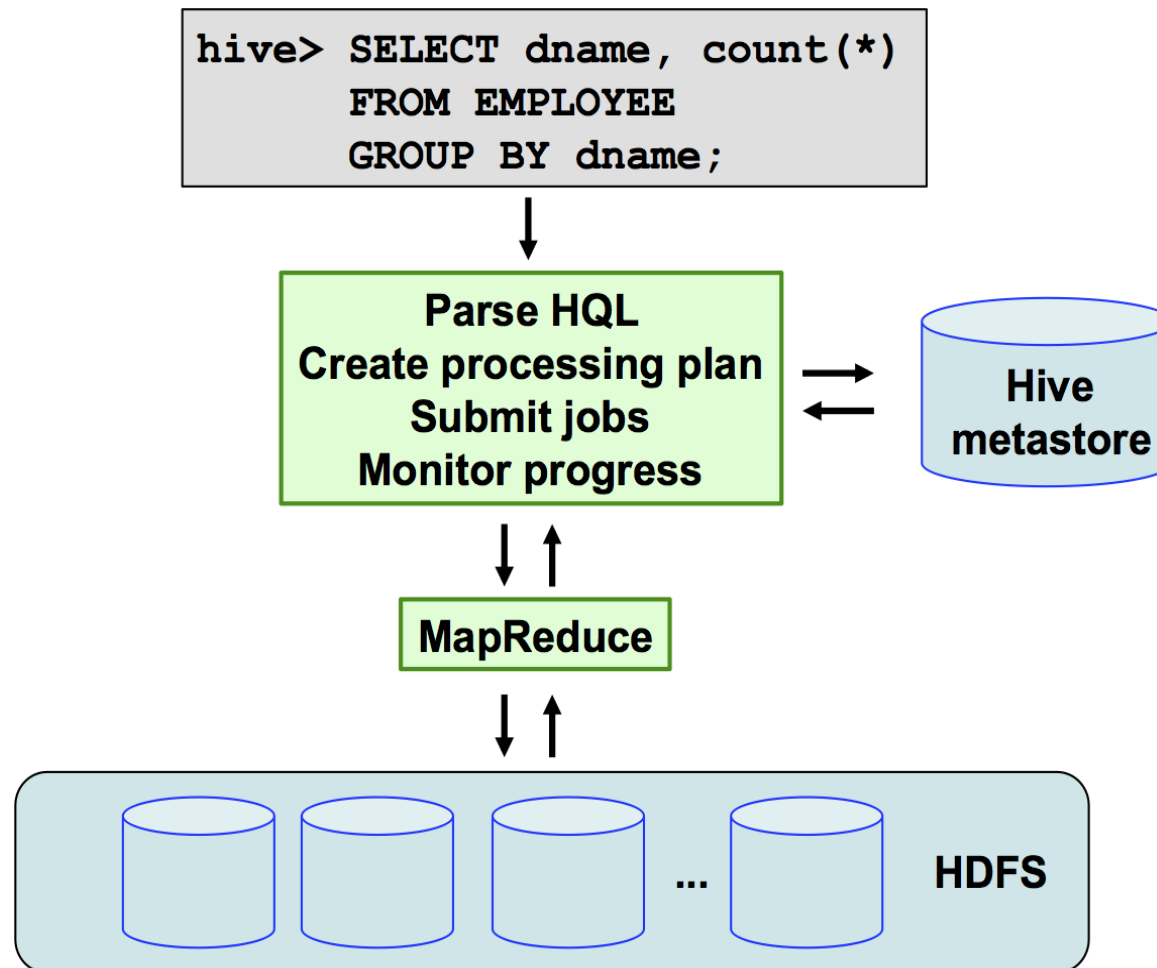
HQL provides a tabular view of data and it can be used to access data located in **HDFS**

Hive frees data analysts from **Java MapReduce** programming skills (not completely)

HQL statements are parsed by the **Hive client** and translated into a sequence of **Java MapReduce** operations, which are later on processed by **Hadoop**

Hive ? What is it ?

The results of processing by [Hadoop](#) are returned to the client or saved in [HDFS](#)



Hive

Outline

[Hive ? What is it ?](#)

[Deployment and configuration](#)

[Metastore](#)

[Interfaces](#)

[HQL](#)

[Hive versus relational DBMSs](#)

Deployment and Configuration

Hive is available on all of commercial distributions of **Hadoop** and on Hadoop installation on our virtual machine

A relational embedded database system **Derby** is used for implementation of **metastore**

It is possible to use other relational database systems for implementation of **metastore** like for example **MySQL**

To use **Hive** **Hadoop** and **HDFS** must be "up and running"

A top level view of data provided by **Hive** consists of **databases** and **tables**

Hive

Outline

[Hive ? What is it ?](#)

[Deployment and configuration](#)

[Metastore](#)

[Interfaces](#)

[HQL](#)

[Hive versus relational DBMSs](#)

Metastore

Metastore contains the mappings of **tables** to the **directory locations** in **HDFS**

Metastore is a relational database read and written by **Hive** client

Metastore also includes the **input and output formats** for the files represented by the table objects, e.g. **CSV InputFormat**, etc, and **SerDes** (Serialization/ Deserialization) **functions**

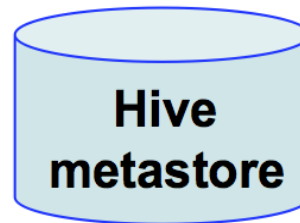
Input and output formats for the files and **functions** are used by **Hive** to extract records and fields from the files

Metastore

```
hive> CREATE TABLE DEPARTMENT  
      ( dname string,  
        budget bigint,  
        cdate date );
```



Saved in



Retrieved from

```
hive> SELECT dname  
      FROM DEPARTMENT  
      WHERE budget > 100000;
```

Hive

Outline

[Hive ? What is it ?](#)

[Deployment and configuration](#)

[Metastore](#)

[Interfaces](#)

[HQL](#)

[Hive versus relational DBMSs](#)

Interfaces

Hive provides **Command Line Interface (CLI)** that accepts and parses **HQL** commands

Hive provides JDBC/ODBC connector (drivers) to work with other tools such as:

- **beeline** (CLI),
- **Oracle SQL Developer** (GUI),
- **Talend Open Studio** (data extraction, transformation, loading, and integration tools),
- **Jasper reports, QlikView** (business intelligence reporting tools),
- **Microsoft Excel 2013** (data analysis tools), and **Tableau** (data visualization tools)

Hive provides a storage handler mechanism to integrate with **HBase**

HUE (Hadoop User Experience) provides a unified web interface to **HDFS** and **Hive** in an interactive environment

HCatalog provides metadata management system for **Hadoop, Pig, Hive,** and **MapReduce**

Hive

Outline

[Hive ? What is it ?](#)

[Deployment and configuration](#)

[Metastore](#)

[Interfaces](#)

[HQL](#)

[Hive versus relational DBMSs](#)

HQL

HQL consists of Data Definition Language, Data Selection and Scope Language, Data Manipulation Language, and Data Aggregation and Sampling Language

Data Definition Language is used for creating, deleting, and altering schema objects like **database tables, views, partitions, and buckets**

Data Selection and Scope Language is used for querying data, linking data, and limiting the data ranges or scopes

Data Manipulation Language is used for exchanging, moving, sorting, and transforming data

Data Aggregation and Sampling Language is used for exchanging, moving, sorting, and transforming data

Hive

Outline

[Hive ? What is it ?](#)

[Deployment and configuration](#)

[Metastore](#)

[Interfaces](#)

[HQL](#)

[Hive versus relational DBMSs](#)

Hive versus relational DBMS

Similarities

- Tabular view of data objects in [HDFS](#)
- Directories and files viewed as tables
- Types of columns in tables
- Access to tables through [HQL](#) very similar to [SQL](#)
- API interface the same as [JDBC](#) programming interface

Differences

- Load and read-only data management system based on implementation of [HDFS](#)
- It is still possible to access data visible in tabular format in Hive directly through [HDFS](#)
- [UPDATE](#) supported as coarse-grained transformation instead of [fine-grained](#) transformation in relational DBMSs
- No transaction processing system
- No verification of consistency constraints, e.g. primary keys, foreign keys, domains constraints, etc

References

Gross C., GuptaA., Shaw S., Vermeulen A. F., Kjerrumgaar D., Practical Hive: A guide to Hadoop's Data Warehouse System, Apress 2016, Chapter 4 (Available through UOW library)

Lee D., Instant Apache Hive essentials how-to: leverage your knowledge of SQL to easily write distributed data processing applications on Hadoop using Apache Hive, Packt Publishing Ltd. 2013 (Available through UOW library)

[Apache Hive](#)