



European
Commission

Hive SQL over Hadoop

THE CONTRACTOR IS ACTING UNDER A FRAMEWORK CONTRACT CONCLUDED WITH THE COMMISSION

Introduction

- **Apache Hive is a high-level abstraction on top of MapReduce**
 - Uses an SQL/like language called HiveQL
 - Generates MapReduce jobs that run on the Hadoop cluster
 - Originally developed by Facebook for data warehousing
 - Now an open/source Apache project

Overview

- HiveQL queries are transparently mapped into MapReduce jobs at runtime by the Hive execution engine
 - **Also makes optimizations**
- Jobs are submitted to the Hadoop cluster

Hive Tables

- Hive works on the abstraction of table, similar to a table in a relational database
- Main difference: the table is simply a directory in HDFS, containing one or more files
 - **By default files are in text format but different formats can be specified**
- The structure and location of the table are stored in an backing SQL database called the metastore
 - **Transparent for the user**
 - **Might be any RDBMS, specified at configuration time**

Hive Tables

- At query time, the metastore is consulted to check if the query is consistent with the tables it invokes
- The query itself operates on the actual data files stored in HDFS

Hive Tables

- By default, tables are stored in a warehouse directory on HDFS
 - **Default location:**
`/user/hive/warehouse/<db>/<table>`
- Each subdirectory of the warehouse directory is considered a database
- Each subdirectory of a database directory is a table
- All files in a table directory are considered part of the table when querying
 - **Must have the same structure**

Hive Tables

- Data files are moved under the warehouse directory when a table is created and/or loaded with new data
- Possible to create external tables if data files must be maintained in original location

Hive Data Format

- Data format by default is plain text files
 - **Columns are delimited by a separator**
- It is possible to import text data in a compressed format, such as gzip
 - **The compression will be detected automatically and the file will be decompressed on-the-fly during query execution**
 - **However, file cannot be split, hence query cannot run in parallel**
- The alternative way to compress data is use the SequenceFile format, that compresses data and can also split files on different nodes

Hive Data Format

- Other non-text data formats can be used
 - **Parquet: compressed, columnar data format that can be used in the whole Hadoop ecosystem. Natively supported in Hive starting from version 0.13**
 - **SerDe: arbitrary binary or text format, specifying a custom Serializer/Deserializer**

Hive Queries

- Querying data is very similar to plain SQL with familiar syntax
- This facilitates especially join operations that are very complex in MapReduce
- Results of the output of a query can be written back to HDFS

Hive Shell

- Hive commands can be executed interactively in the hive shell

```
>hive
```

- Can work better than Hue sometimes...
 - **However, be careful when issuing commands that can return a big output...**
- Queries can be also directly issued from the command line (useful for output redirection)

```
>hive -e 'SELECT * FROM yourtable'
```

Hive Limitations

- Not all “standard” SQL is supported
 - **Subqueries are only supported in the FROM clause**
 - **No correlated subqueries**
- No support for UPDATE or DELETE
- No support for INSERTing single rows