# Big Data with Hadoop

Pierre Sauvage
Big Data Consultant
pierre@adaltas.com

# Q/A Assignment 1

# Hive: the Data Warehouse

SQL to MapReduce

# Brief introduction

- Started at Facebook, now an apache Top-Level Project
- MR Generator
- No magic here, still batch-oriented

# Why ?

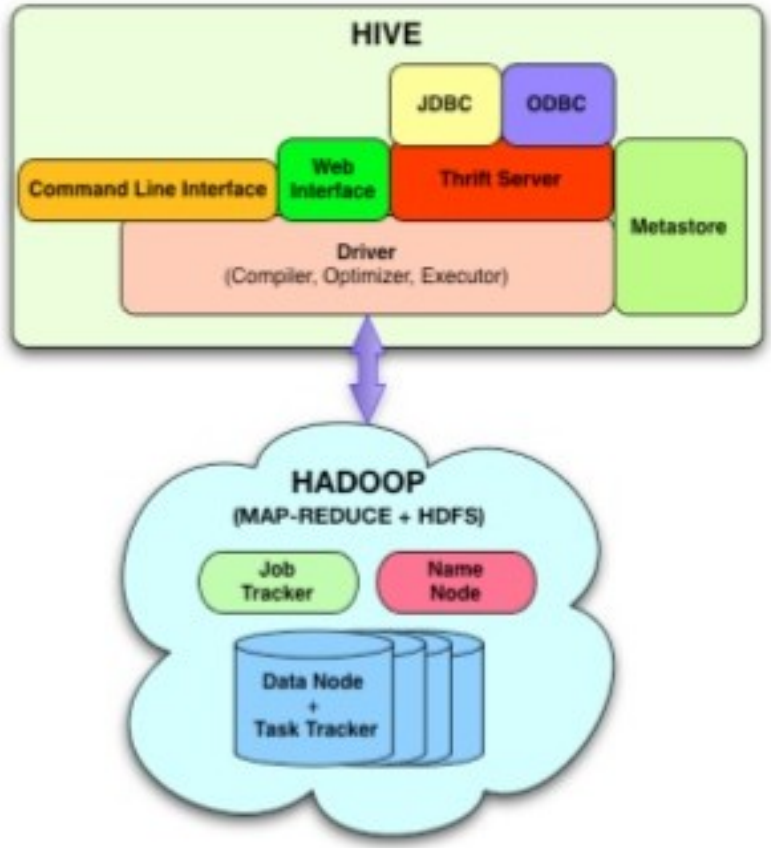
- Data analysts like SQL
- Easier and faster than MR
- Wide acceptance inside the enterprise (enterprise ready)
- REPL client
- JDBC and ODBC

# Hive Applications

- Log processing
- Text mining
- Document indexing
- Business Intelligence
- Etc..

Any Batch-oriented data analysis !

# Hive Architecture



# Data Model

- Tables
  - HDFS directory
- Partitions
  - Don’t abuse (small files problem)
- SQL-like query language called HiveQL
  - Ex: No Update statement

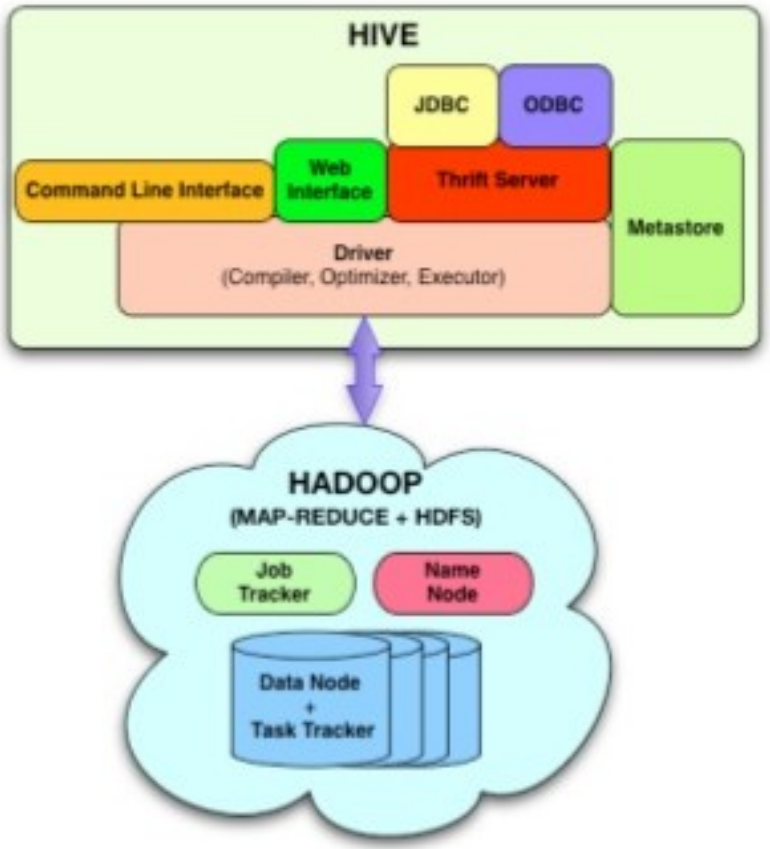
# Data types

- Numeric Types: TINYINT, SMALLINT, INT, BIGINT, FLOAT, DOUBLE, DECIMAL
- Date/Time Types: TIMESTAMP, DATE
- String Types: STRING, VARCHAR
- Complex Types: ARRAY, MAP, STRUCT, UNIONTYPE
- Misc Types: BOOLEAN, BINARY

# Example

- Create table toto(
  s STRING,
  i INT,
  a ARRAY<MAP<String, STRUCT<foo: FLOAT, bar: FLOAT>>>);
- SELECT s, I, a[23]['foobar'].bar FROM toto

# Metastore



# Metastore

- Database: namespace containing a set of tables
- Tables/partitions definitions (column types, HDFS directory of table)
- Stats
- Any RDBMS: derby, mySQL, postgreSQL, etc..

# In HDFS

- Data Warehouse directory
  - Ex: /user/hive/warehouse
- Tables are subdirectory of DWH
- Partitions are subdirectory of DWH
- Data stored in files
  - SequenceFiles, ORC, Parquet, Custom Ser/De

# Basic Commands

- List tables/databases/function
  - hive> show databases;
  - hive> show tables;
- Describe tables/function
  - hive> describe *$tablename*
  - hive> describe extended *$tablename*
- SQL query
  - hive> SELECT * ... LIMIT 10

# Manipulate Tables

- CREATE
- SHOW
- ALTER
- DROP

# Create Tables

- Not so hard
  - CREATE TABLE foo (id INT, msg STRING);
- Default table layout
  - Text files; fields delimited by \001; rows delimited by \n
- Should use more optimised layout

# Partitionning vs Bucketing

- Partitions
  - User Defined → choose well for load-balancing
- Buckets
  - Hash defined
  - Must be properly loaded: set as many reducers as buckets
- Can be combined

# Partitionning vs Bucketing

- CREATE TABLE order (
  username STRING,
  ord STRING,
  amount DOUBLE
  ) PARTITIONED BY (company STRING)
  CLUSTERED BY (username) INTO 25 BUCKETS;

# Partitionning vs Bucketing

- CREATE TABLE order (
  username STRING,
  ord STRING,
  amount DOUBLE
  ) PARTITIONED BY (company STRING)
  CLUSTERED BY (username) INTO 25 BUCKETS;

# Resume

- SQL over Hadoop
- Highly customizable
- MR compiler
- Wait... Did I say Map Reduce ?

# Summary

- SQL over Hadoop
- Highly customizable
- MR compiler
- Wait... Did I say Map Reduce ?