**Hive Data Modeling**

1. Tables
2. Partitions – tables organized into partitions for grouping same type of data based on partition key (for more efficient querying)
3. Buckets – partitioned partitions

**Hive Data Types**

Primitive Data Types

1. Numeric
2. String
3. Date/Time
4. Miscellaneous (boolean and binary)

Complex Data Types

1. Arrays
2. Maps (one key per mapped value, central to Hadoop)
3. Structs – collection of complex data with comment
   1. Example: struct<col\_name:data\_type[COMMENT col\_comment],…>
4. Units – a collection of heterogenous data types

**Hive Modes**

Local Mode –

* Is used when Hadoop is having one data node and the data is small
* Processing will be very fast on smaller datasets which are present in local machine

MapReduce Mode-

* Is used when Hadoop is having multiple data nodes and the data is spread across various data nodes
* Processing large datasets can be more efficient using this mode (usually 10-100 computers)

**Differences between RDBMS**

* Hive enforces schema on read (RDBMS on write)
* Hive data size is in petabytes (terabytes)
* Hive is based on notion of write once and read many times (read and write many times)
* Is a data warehouse (database)
* Easily scalable at low cost

**Features of Hive**

* Uses HiveQL (easier than long codes)
* Tables are used which are similar to RDMBS and easy to understand
* Multiple users can simultaneously query data
* Supports variety of data formats

hive.apache.org

**Hive Commands**

$ hive -e ‘select a.col from tab1 a’

$ hive -I initialize.sql

show databases;

drop database office cascade;

create database office;

use office;

cat Employee.csv

gedit Employee.csv

create table employee *//no punctuation so command continues to next line*

(Id INT, Name STRING, Dept STRING)

row format delimited fields terminated by ‘,’

tbleproperties (“skip.header.line.count”=”1”);

show tables;

describe employee;

select \* from employee;

beware of spaces, want to process data before it goes into the db