

Lesson 5

HBase

HBase

- An Hadoop system database
- Created for storing large structured tables
- HBase is an open-source
- Distributed, versioned
- Non-relational (NoSQL) database
- Written in Java

HDFS Features

1. Uses a partial columnar data schema on top of Hadoop and HDFS.
2. Supports a large table of billions of rows and millions of columns.
3. Provides small amounts of information, called sparse data taken from large data sets which are storing empty or presently not-required data

... HDFS Features

4. Supports data compression algorithms.
5. Provisions in-memory column-based data transactions.
6. Accesses rows serially and does not provision for random accesses and write into the rows.
7. Provides random, real-time read/write access to Big Data.

... HBase Features

8. Fault tolerant storage due to automatic failure support between DataNodes servers
9. Similarity with Google BigTable
10. Provides scalable distributed Big Data Store
11. HBase data store as key-value pairs.

HBase Schema

- Applies a partial columnar scheme on top of the Hadoop and HDFS
- An HBase column represents an attribute of an object

HBase Format

- Row-Key Column-Family: { Column-Qualifier: Version: Value }
- Refer Example 2.3 for
hourly sales of Kit Kat (KKHS),
Milk, Fruit and Nuts (FNHS), Nougat
(NHS) and Oreo (OHS) sold every
hour at an ACVM of ID ACVM_ID

First Row of Hbase Table in Example

- ACVM_id: '2206'
{ 'DT':1600080000024: '121217',
'HR': 1600008007319: '16', 'KKHS':
1600081010821: '28', 'MHS':
1600082010582: '23', 'FNHS':
1600082018001: '38', 'NHS':
1600080158868: '8', 'OHS':
1600038028229: '50' }

Other Rows

- `hbase (main) 001:0> put 'ACVM_id', '2206', 'DT', '121217', 'HR', '16', 'HourlySales: KKHS', '28'` 0 row(s) in 021120 seconds
- `hbase (main) 002:0> put 'ACVM_id', '2206', 'HourlySales: MHS', '23'` 0 row(s) in 001120 seconds

Summary

We learnt :

- HBase Features
- HBase Structured Table
- Example of First Row of HBase Table
- Example of Other Rows

End of Lesson 5 on **HBase**