Hive

- Hive is a data warehouse system for Hadoop that facilitates easy data summarization, ad-hoc queries, and the analysis of large datasets stored in Hadoop compatible file systems.

- Hive provides a mechanism to project structure onto this data and query the data using a SQL-like language called HiveQL

What is Hive?

- Hive provides relational interface on top of HDFS
- Hive provides access through JDBC driver
- Hive provides limited facilities to define schema in DDL
- No DatabaseMetadata information
- Not JDBC complaint driver
- Provides a way to define a custom map-reduce job

Hive

- Can any data that is already in the HDFS accessed through Hive?

  No, the data must be inserted using the Hive utilities, that data will be saved into to HDFS store. You can then access that data using Hive's HQL interface.

Hive – Creating Tables

- Create Table in Hive

```
CREATE TABLE pokes (
    foo INT,
    bar STRING);
```

-

- CREATE TABLE invites (

```
    foo INT,
    bar STRING)
    PARTITIONED BY (ds STRING)
```

Hive – Loading Data

- Local File

  LOAD DATA LOCAL INPATH './examples/files/kv1.txt'
  [OVERWRITE] INTO TABLE pokes;

- File in HDFS (data partitioned)

  LOAD DATA INPATH './examples/files/kv2.txt' INTO TABLE invites PARTITION (ds='2008-08-15');

- Moves the data into Hive controlled store

Hive – Query Support

- Simple selects
- Aggregate Function support (SUM, AVG, MIN..)
- Group By
- Equi- joins on columns

Hive - EDS

- “hive” translator
- Supports dynamic metadata
- No Designer support, use DDL importer for creating the source models
- No transaction support
- Concern – Hive still is evolving, not much information usage by customers.

GSS API

GSS API enables single sign on (SSO) for Java based applications

- EDS supports Kerberos using GSS API
- EDS makes use of JBoss Negotiation libraries on the JBoss AS as JAAS module.
- Supported with on both JDBC and ODBC
- In JDBC, both Local Connections and Remote connections are supported

GSS API

- If your web app is configured with SSO then you can configure the Teiid local connection as “pass-through” connection to use same security context.
- Redhat, IT uses this feature in their infrastructure.
- Check Admin Guide for configuration.
- You need a KDC store for a DEMO.

Memory Management

- By default uses heap memory and disk.
- On heap Overhead of 100-200 bytes per batch/page
    - If number of storage rows is in the billions then increase batch sizes and also increase “maxStorageObjectSize” from default 8MB, this is per batch/page.
- “memoryBufferSpace” defines the amount of **dedicated** memory used by EDS as a serialization buffer/cache.

memoryBufferSpace

- Can be created ON or OFF heap
- Default size is -1, which means automatically calculates based on “maxReserveKB” and other properties
- Should be at least several times larger than maxStorageObjectSize for better write concurrency.
- Set “memoryBufferOffHeap=true” for direct memory allocated to the process, but off heap

Off heap memoryBufferSpace

- Preferable to on heap for dedicated EDS severs to minimize GC overhead when more than 32 GB of RAM is available for use.
- The Java heap still needs an appropriate amount of memory.
- Consult the VM for additional settings. E.g. with the Oracle VM to allocate 12 GB of direct memory, use the following VM properties

  -XX:MaxDirectMemorySize=12g -XX:+UseLargePages