UNIVERSITY everis

Apache Hive

Vinícius M. Bueno

vinicius.manini.bueno@everis.com

UNIVERSITY everis

Agenda

- Introdução
- Definição
- Arquitetura
- Opções Hive client
- Mecanismo de execução
- Modelagem de dados
- Tipos de tabelas
- Tipos de dados
- Formatos de arquivos
- Comandos para praticar

UNIVERSITY everis

Introdução

UNIVERSITY everis

Introdução

O Hive é um framework para soluções de Data Warehousing, que executa no ambiente do Hadoop.

Software construído sobre o Apache Hadoop

Desenvolvido em 2007 pelo Facebook

A partir de 2008

Projeto Open Source

Fornece acesso semelhante ao SQL(HQL)

UNIVERSITY everis

Definição

Definição

O software de data warehouse "**Apache Hive**" facilita a leitura, gravação e gerenciamento de grandes conjuntos de dados residentes no armazenamento distribuído usando a sintaxe SQL (Structured Query Language). A estrutura pode ser projetada nos dados já armazenados. Uma ferramenta de linha de comando e um driver JDBC são fornecidos para conectar os usuários ao Hive.

UNIVERSITY everis

Definição

HQL Hive Query Language

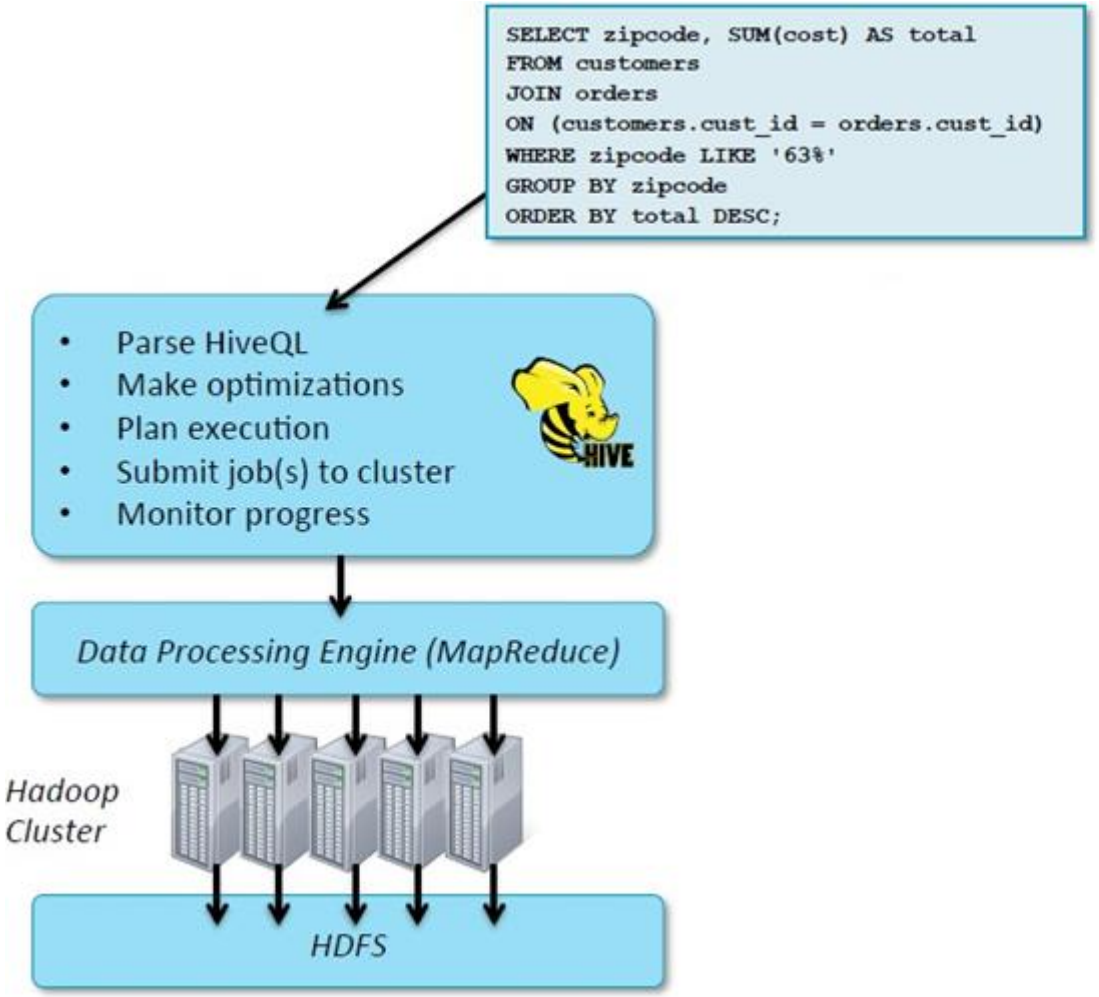
- HQL: Linguagem de consulta do tipo sql;
- Tem semânticas e funções semelhantes às do SQL padrão no banco de dados relacional;
- As consultas HQL são traduzidas implicitamente em um ou mais serviços de MapReduce, protegendo o usuário de programação muito avançada e consumo de tempo
- Gera jobs MapReduce ou Spark no cluster Hadoop;
- A linguagem de consulta do Hive pode ser executada em diferentes mecanismos de computação, como MapReduce, Spark e Tez;

UNIVERSITY everis

Arquitetura

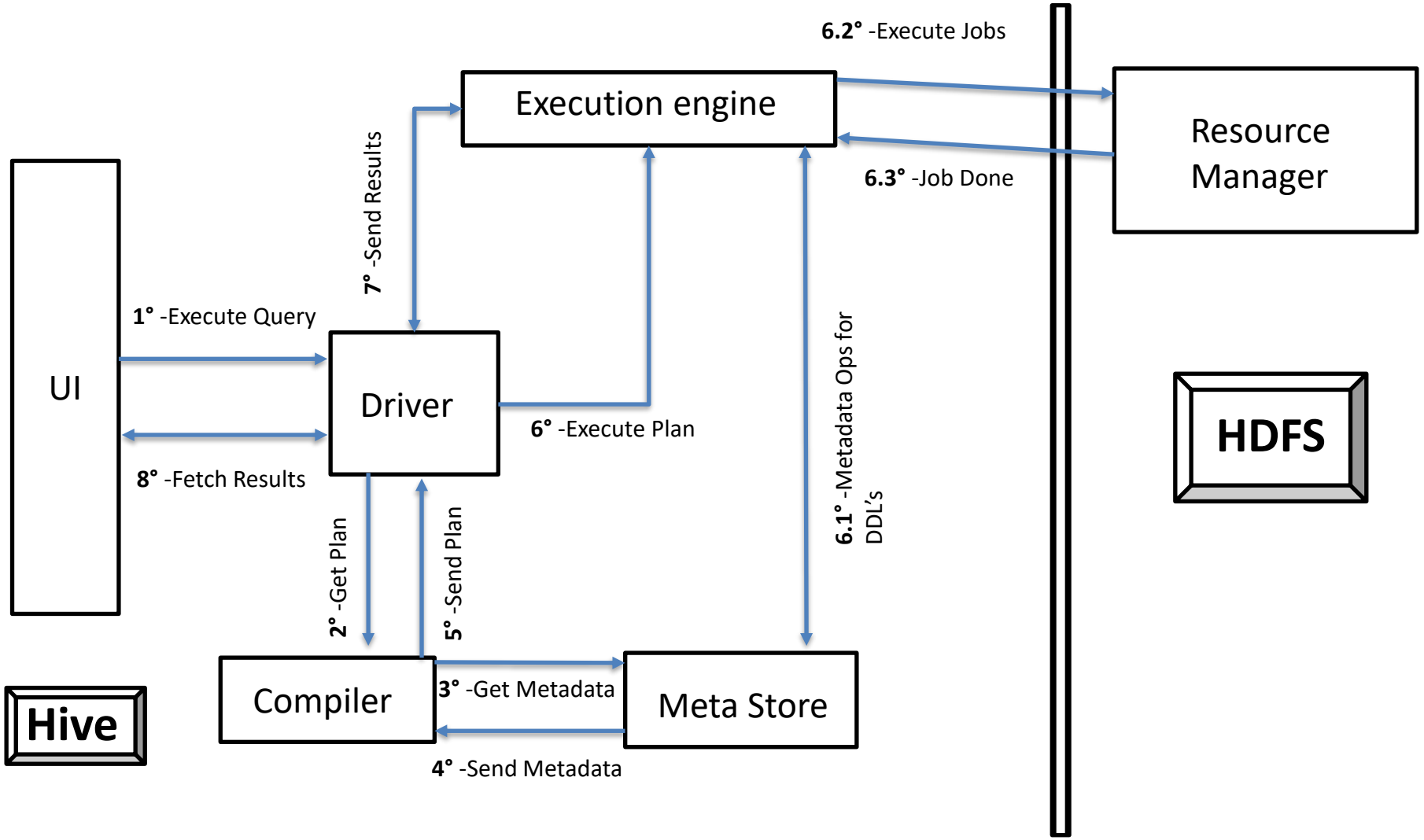
UNIVERSITY everis

Arquitetura



UNIVERSITY everis

Arquitetura



UNIVERSITY everis

Opções Hive client

UNIVERSITY everis

Opções Hive client

Algumas opções estão disponíveis para serem executadas por linha de comando.

Opções disponíveis:

- **`hive -d`** : expressão para substituir uma informação mutável.
- **`hive --database`** : expressão para indicar um banco de dados específico.
- **`hive -e`** : expressão para rodar uma query direto na linha de comando.
- **`hive -f`** : expressão para rodar uma query através de um arquivo.
- **`hive --hiveconf`** :expressão para definir atributos que a sessão terá durante o tempo de execução.
- **`hive --hivevar`** : expressão para substituir uma informação mutável.
- **`hive -i`** : expressão para iniciar uma sessão com atributo que estão definido em um arquivo.
- **`hive -S`** : expressão para rodar em modo silencioso (algumas informações serão ocultadas durante a execução).
- **`hive -v`** : expressão para detalhar cada comando que está sendo executado na tela.

UNIVERSITY everis

Opções Hive client

Para ter acesso a todas as opções execute o comando " **`hive -H`** ".

```
[cloudera@quickstart ~]$ hive -H
usage: hive
 -d,--define <key=value>          Variable subsitution to apply to hive
                                  commands. e.g. -d A=B or --define A=B
    --database <databasename>     Specify the database to use
 -e <quoted-query-string>         SQL from command line
 -f <filename>                    SQL from files
 -H,--help                        Print help information
    --hiveconf <property=value>   Use value for given property
    --hivevar <key=value>         Variable subsitution to apply to hive
                                  commands. e.g. --hivevar A=B
 -i <filename>                    Initialization SQL file
 -S,--silent                      Silent mode in interactive shell
 -v,--verbose                     Verbose mode (echo executed SQL to the
                                  console)

[cloudera@quickstart ~]$
```

UNIVERSITY everis

Mecanismo de execução

Mecanismo de execução

Você tem as seguintes opções de plug ins para o seu mecanismo (Hive Engine):

- ✓ **Map-Reduce:** mecanismo padrão do hive.
- ✓ **Spark:** mecanismo que precisa ser instalado, as suas características principais é a capacidade de utilizar a memória, e a arquitetura DAG (Directed Acyclic Graph).
- ✓ **Tez:** mecanismo que precisa ser instalado, parecido com spark também utiliza memória, e a arquitetura DAG (Directed Acyclic Graph).

Mecanismo de execução

Engine Map-Reduce

Mecanismo padrão do hive que utiliza leitura, e escrita em disco. Indicado para executar queries complexas.

Linha de comando para definir esse mecanismo na sessão:

```
set hive.execution.engine=mr;
```

Um típico map reduce tem as seguintes etapas:

1. Ler dados de arquivos (primeiro acesso aos discos);
2. Rodando os mappers
3. Escrevendo saída map (segundo acesso aos discos);
4. Rodando shuffle e sort (lendo saída map, terceiro acesso aos discos)
5. Escrevendo shuffle e sort --> escrevendo os dados ordenados para os reducers (quarto acesso aos discos);
6. Rodando reducers com leitura ordenada dos dados (quinto acesso aos discos);
7. Escrevendo as saídas dos reducers (sexto acesso aos discos);

Mecanismo de execução

Engine Spark

O Apache Spark é um mecanismo de análise de código aberto e uma estrutura de computação de cluster para processamento de big data.

Linha de comando para definir esse mecanismo na sessão:

```
set hive.execution.engine=spark;
```

Mecanismo de execução

Engine Tez

Como o Spark, o Apache Tez criado pela Hortonworks é uma estrutura de código-fonte aberto para processamento de big data com base na tecnologia MapReduce.

O Spark e o Tez oferecem um mecanismo de execução capaz de usar gráficos acíclicos direcionados (DAGs) para processar quantidades extremamente grandes de dados.

Linha de comando para definir esse mecanismo na sessão:

```
set hive.execution.engine=tez;
```

Um típico job Tez tem as seguintes etapas:

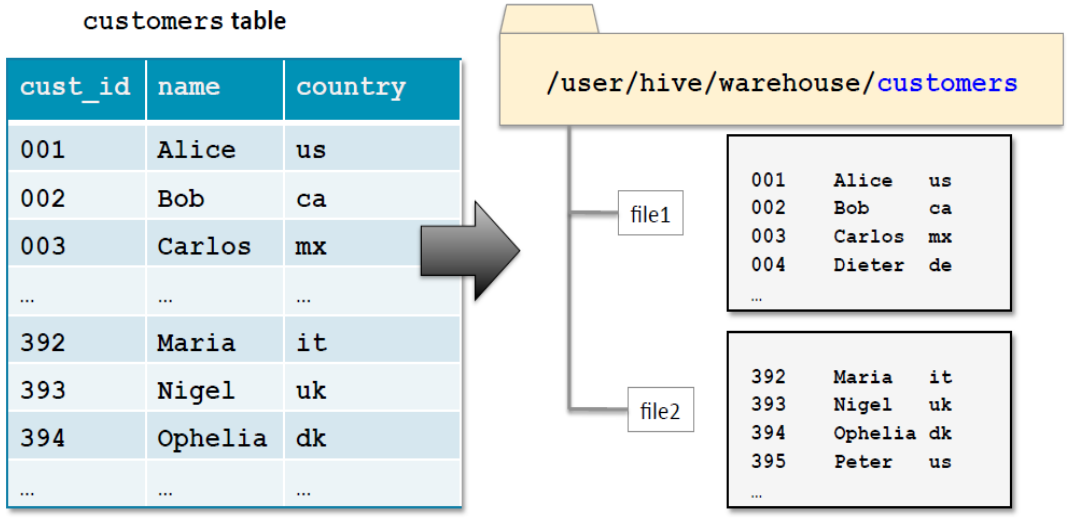
1. Executar o plano (não é necessário ler dados do disco);
2. Quando estiver pronto para fazer alguns cálculos (semelhantes às ações no spark), obtenha os dados do disco e execute todas as etapas e produza a saída (apenas uma leitura e uma gravação).

UNIVERSITY everis

Modelagem de dados

Modelo de dados

- Os dados são organizados em forma de tabelas e partições;
- Uma tabela é simplesmente um diretório no HDFS contendo um ou vários arquivos;



UNIVERSITY everis

Tipos de tabelas

Tipos de tabelas

EXTERNAL TABLE x MANAGED TABLE

EXTERNAL TABLE

- Hive assume que não gerencia os dados;
- DROP apaga somente os metadados ;
- Use tabelas externas quando os arquivos já estiverem presentes ou em locais remotos;
- Use tabelas externas quando os arquivos devem permanecer mesmo se a tabela for descartada;

MANAGED TABLE

- Os dados são armazenados no diretório warehouse do Hive;
- Localizado em /hive/warehouse/
- Sempre que uma tabela for alterada, os dados também serão alterados;
- Use tabelas gerenciadas quando o Hive deve gerenciar o ciclo de vida da tabela ou ao gerar tabelas temporárias.

Tipos de tabelas

External Table

```
CREATE EXTERNAL TABLE if not
exists  table_external(
  id int COMMENT 'codigo de
  identificacao',
  descricao int COMMENT 'nome do
  job',
  salario float COMMENT 'salario
pago pelo  job'
) COMMENT 'tabela de descricao de
jobs'
ROW FORMAT DELIMITED FIELDS
TERMINATED BY '\;'
STORED AS TEXTFILE
LOCATION '/default/job';
```

Managed Table

```
CREATE TABLE if not exists
table_managed(
  id int COMMENT 'codigo de
  identificacao',  descricao int
  COMMENT 'nome do job',  salario
  float COMMENT 'salario pago pelo
job'
) COMMENT 'tabela de descricao de
jobs'  ROW FORMAT DELIMITED FIELDS
TERMINATED BY '\;'
STORED AS TEXTFILE;
```

UNIVERSITY everis

Tipos de dados

UNIVERSITY everis

Tipos de dados

Numeric Types

- `TINYINT` (1-byte signed integer, from `-128` to `127`)
- `SMALLINT` (2-byte signed integer, from `-32,768` to `32,767`)
- `INT/INTEGER` (4-byte signed integer, from `-2,147,483,648` to `2,147,483,647`)
- `BIGINT` (8-byte signed integer, from `-9,223,372,036,854,775,808` to `9,223,372,036,854,775,807`)
- `FLOAT` (4-byte single precision floating point number)
- `DOUBLE` (8-byte double precision floating point number)
- `DOUBLE PRECISION` (alias for DOUBLE, only available starting with Hive 2.2.0)
- `DECIMAL`
  - Introduced in Hive 0.11.0 with a precision of 38 digits
  - Hive 0.13.0 introduced user-definable precision and scale
- `NUMERIC` (same as `DECIMAL`, starting with Hive 3.0.0)

UNIVERSITY everis

Tipos de dados

Date/Time Types

- `TIMESTAMP` (Note: Only available starting with Hive 0.8.0)
- `DATE` (Note: Only available starting with Hive 0.12.0)
- `INTERVAL` (Note: Only available starting with Hive 1.2.0)

String Types

- `STRING`
- `VARCHAR` (Note: Only available starting with Hive 0.12.0)
- `CHAR` (Note: Only available starting with Hive 0.13.0)

Misc Types

- `BOOLEAN`
- `BINARY` (Note: Only available starting with Hive 0.8.0)

UNIVERSITY everis

Formatos de arquivos

Formatos de arquivos

Ao se criar uma tabela, é possível que se determine o formato que ela terá

```
CREATE TABLE tablename (colname DATATYPE, ...)
   ROW FORMAT DELIMITED
   FIELDS TERMINATED BY char
   STORED AS format
```

UNIVERSITY everis

Formatos de arquivos

- **TEXTFILE** é o formato mais básico no Hadoop
  - Pode ser delimitado por tab ou vírgula;
  - Boa interoperabilidade, performance ruim.

- **SEQUENCEFILE** grava par de chave-valores em formato de container binário
  - Menos verboso, mais eficiente;
  - Suporte formatos binários (imagens, por exemplo);
  - Boa performance, interoperabilidade ruim.

UNIVERSITY everis

Formatos de arquivos

```
CREATE EXTERNAL TABLE if not exists
TABLE_PARQUET(
  id int COMMENT 'codigo de identificacao',
  descricao int COMMENT 'nome do job',
  salario float COMMENT 'salario pago pelo  job'
) COMMENT 'tabela de descricao de jobs'
STORED AS PARQUET;
```

- É um formato colunar desenvolvido pela Cloudera e Twitter;
- Reduz o espaço de armazenamento;
- Aumenta a performance;
- Mais eficiente ao adicionar muitos registros de uma vez;
- Esquema armazenado no rodapé;
- Formato de armazenamento orientado a colunas;
- Possui compressão e índices integrados;

Formatos de arquivos

```
CREATE EXTERNAL TABLE if not exists
TABLE_AVRO(
  id int COMMENT 'codigo de identificacao',
  descricao int COMMENT 'nome do job',
  salario float COMMENT 'salario pago pelo  job'
) COMMENT 'tabela de descricao de jobs'
STORED AS AVRO;
```

- É o formato principal de linha;
- Seu principal objetivo de projeto era a evolução do esquema;
- No formato avro, armazenamos o esquema separadamente dos dados. Geralmente, o arquivo de esquema avro (.avsc) é mantido;

Formatos de arquivos

```
CREATE EXTERNAL TABLE if not exists
TABLE_ORC(
  id int COMMENT 'codigo de identificacao',
  descricao int COMMENT 'nome do job',
  salario float COMMENT 'salario pago pelo  job'
) COMMENT 'tabela de descricao de jobs'
STORED AS ORC;
```

- Semelhante ao Parquet.
- Formato de armazenamento orientado a colunas;
- Originalmente, é o arquivo Colunar de Linhas da Hive;
- O esquema está com os dados, mas como parte do rodapé;
- Os dados são armazenados como grupos de linhas e faixas;
- Cada faixa mantém índices e estatísticas sobre os dados armazenados;

Formatos de arquivos

Comparação entre os formatos Avro, Parquet e Orc.

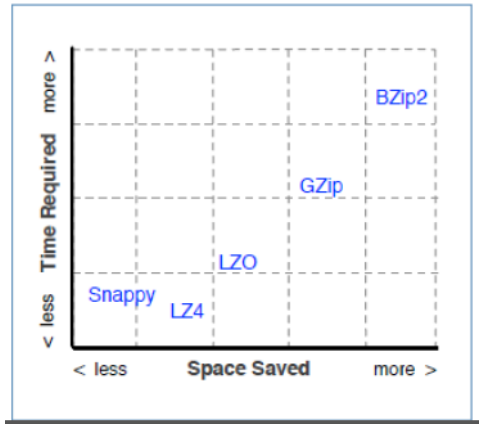
BIG DATA FORMATS COMPARISON

	Avro	Parquet	ORC
Schema Evolution Support			
Compression			
Splitability			
Most Compatible Platforms	Kafka, Druid	Impala, Arrow Drill, Spark	Hive, Presto
Row or Column	Row	Column	Column
Read or Write	Write	Read	Read

Formatos de arquivos

Vantagens

- Pode melhorar a performance significativamente;
- Melhora o trafego de dados na rede;
- Algoritmos mais agressivos ocupam menos espaço, mas pode ser lentos para leitura;
- Algoritmos menos agressivos, ocupam menos espaços, mas geralmente são mais rápidos;
- A implementação do algoritmo de compressão chama-se codec;
- Muitos codecs são utilizados com Hadoop;
- Cada codec tem características de performance diferentes;
- LZ4 e Snappy são os mais rápidos;



UNIVERSITY everis

Comandos para praticar

Comandos para praticar | DataBase

```
CREATE DATABASE loudacre;
```

```
CREATE DATABASE IF NOT EXISTS loudacre;
```

```
/user/hive/warehouse/loudacre.db
```

```
DROP DATABASE loudacre;
```

```
DROP DATABASE IF EXISTS loudacre;
```

Comandos para praticar | Table

```
CREATE TABLE tablename (colname DATATYPE, ...)
  ROW FORMAT DELIMITED
  FIELDS TERMINATED BY char
  STORED AS {TEXTFILE|SEQUENCEFILE|…}
```

- Default database:
  /user/hive/warehouse/*tablename*
- Named database:
  /user/hive/warehouse/*dbname*.db/*tablename*

Comandos para praticar | Table

```
CREATE TABLE jobs (
    id INT,
    title STRING,
    salary INT,
    posted TIMESTAMP
  )
  ROW FORMAT DELIMITED
  FIELDS TERMINATED BY ',';
```

```
1,Data Analyst,100000,2013-06-21 15:52:03
```

Comandos para praticar | Table

```
CREATE TABLE jobs_archived LIKE jobs;
```

```
CREATE TABLE ny_customers AS
  SELECT cust_id, fname, lname
  FROM customers
  WHERE state = 'NY';
```

Comandos para praticar | Table

Criando uma tabela MANAGED

- Basicamente um arquivos criado pelo próprio no HDFS;
- Caso você drope a tabela, todo o dado posteriormente criado será eliminado no processo!

```
CREATE TABLE jobs (
    id INT,
    title STRING,
    salary INT,
    posted TIMESTAMP
  )
  ROW FORMAT DELIMITED
  FIELDS TERMINATED BY ','
  LOCATION '/loudacre/jobs';
```

Comandos para praticar | Table

Criando uma tabela EXTERNAL

- Cria um metadado para acesso ao arquivo no HDFS;
- Nesse caso, se dropar a tabela, o dado permanecerá.

```
CREATE EXTERNAL TABLE adclicks
  ( campaign_id STRING,
    click_time TIMESTAMP,
    keyword STRING,
    site STRING,
    placement STRING,
    was_clicked BOOLEAN,
    cost SMALLINT)
  LOCATION '/loudacre/ad_data';
```

Comandos para praticar

Table

```
SHOW TABLES;
+---------------+
|   tab_name    |
+---------------+
| accounts      |
| employees     |
| job           |
| vendors       |
+---------------+
```

```
DESCRIBE jobs;
+--------+-----------+---------+
| name   | type      | comment |
+--------+-----------+---------+
| id     | int       |         |
| title  | string    |         |
| salary | int       |         |
| posted | timestamp |         |
+--------+-----------+---------+
```

Comandos para praticar | Table

```
SHOW CREATE TABLE jobs;
+----------------------------------------------------+
| CREATE TABLE default.jobs                          |
|   id INT,                                          |
|   title STRING,                                    |
|   salary INT,                                      |
|   posted TIMESTAMP                                 |
| )                                                  |
| ROW FORMAT DELIMITED FIELDS TERMINATED BY ','      |
...
```

Referências

- ✓ http://hive.apache.org/
- ✓ https://cwiki.apache.org/confluence/display/Hive/Home
- ✓ https://cwiki.apache.org/confluence/display/Hive
- ✓ https://cwiki.apache.org/confluence/display/Hive/GettingStarted

UNIVERSITY everis

Consulting, Transformation, Technology and Operations