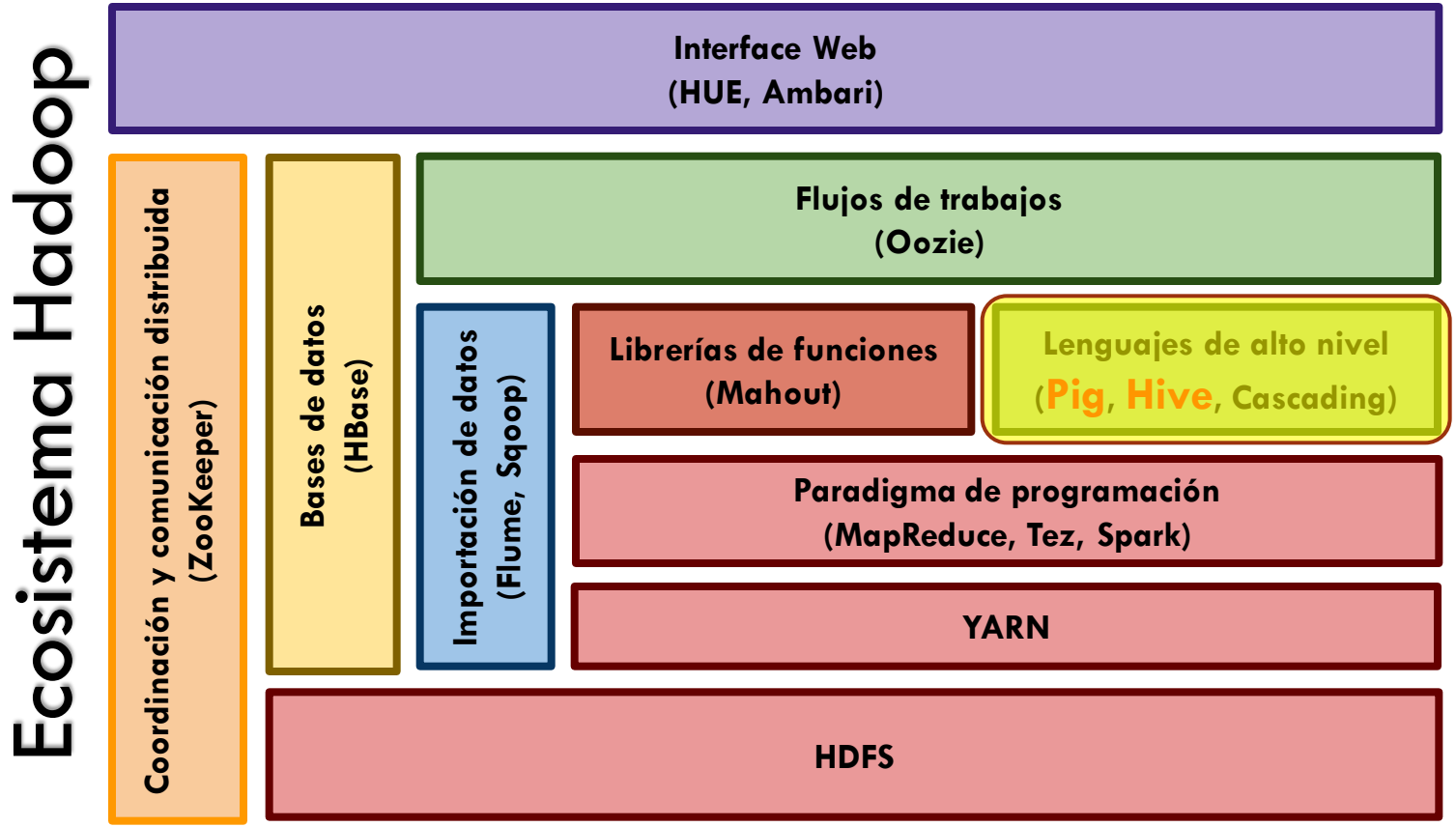
Programación desde lenguajes de alto nivel: Pig y Hive

Interfaces de consulta de alto nivel



Interfaces de consulta de alto nivel

- Introducción (1/2)
  - Hadoop MapReduce ofrece una solución muy potente y versátil.
    - No se necesita implementar detalles como distribución de datos, comunicación o gestión de fallos.
  - Pero todavía es necesario saber programar para poder trabajar sobre un conjunto de datos.
    - También se necesita ser capaz de convertir el algoritmo a implementar al paradigma MapReduce.

Interfaces de consulta de alto nivel

- Introducción (2/2)
  - Algunas tareas de trabajo sobre datos son bastante frecuentes:
    - Especialmente las que trabajan con datos estructurados.
    - Con el paso del tiempo las empresas que más usan Hadoop empiezan a desarrollar herramientas capaces de describir trabajos habituales en MapReduce de forma más ágil.
      - Los primeros en aparecer son **Pig** y **Hive**.
      - A partir de estos proyectos han surgido otros proyectos similares con diferentes objetivos y enfoques.

Interfaces de consulta de alto nivel

- Apache Pig (1/6) https://pig.apache.org/
  - Creado en Yahoo! en 2006 para agilizar la creación de tareas MapReduce.
    - En 2007 pasa a Apache y en 2010 pasa a primer nivel.
    - En 2011, el 40% de los trabajos de Yahoo! que se ejecutan en Hadoop estaban hechos con Pig.
  - Plataforma de análisis de grandes conjuntos de datos.
  - Maneja tuplas (registros), no pares clave-valor.
  - Utiliza el lenguaje PigLatin (parecido a SQL).
    - Se traduce a trabajos MapReduce (*paralelización*)

Interfaces de consulta de alto nivel

- Apache Pig (2/6)

https://pig.apache.org/

  - Cumple las siguientes propiedades (1/4):
    - Facilidad de programación.
      - Es sencillo lograr la ejecución paralela de tareas de análisis de datos simples.
      - Las tareas complejas compuestas de múltiples transformaciones de datos relacionados entre sí están codificados como secuencias de ***flujos de datos***, haciendo que sean fáciles de escribir, entender y mantener.

Interfaces de consulta de alto nivel

- Apache Pig (3/6)

https://pig.apache.org/

  - Cumple las siguientes propiedades (2/4):
    - Código abierto.
      - Los usuarios son libres de descargarlo como fuente o binario, utilizarlo para sí mismos, contribuir a él, y bajo los términos de la licencia de uso de Apache modificarlo si lo consideran conveniente.

Interfaces de consulta de alto nivel

- Apache Pig (4/6)

https://pig.apache.org/

  - Cumple las siguientes propiedades (3/4):
    - MapReduce.
      - PigLatin ofrece operadores para muchas de las operaciones tradicionales de datos: unir, ordenar, agrupar, filtrar, etc.
      - También es posible que los usuarios desarrollen sus propias funciones (UDF: ***U**ser **D**efined **F**unctions*) de lectura, procesamiento y escritura de datos.
      - Las funciones pueden desarrollarse en múltiples lenguajes como JRuby, Python y Java.

Interfaces de consulta de alto nivel

- Apache Pig (5/6)

https://pig.apache.org/

  - Cumple las siguientes propiedades (4/4):
    - Flujo de datos o grafos acíclicos (DAG).
      - PigLatin es un lenguaje de flujo de datos. Esto significa que permite a los usuarios describir cómo los datos de una o más entradas deben ser leídos, procesados y almacenados en una o varias salidas en paralelo.
      - Permite que el sistema pueda optimizar su ejecución de forma automática en los distintos nodos del clúster, ayudando al usuario a centrarse en la semántica en vez de la eficiencia.

Interfaces de consulta de alto nivel

- Apache Pig (6/6) https://pig.apache.org/
  - Modos de ejecución:
    - Modo local interactivo: `pig -x local`
      - Se ejecuta en una máquina virtual simple.
      - Todos los ficheros están en el sistema de ficheros local.
    - Modo MapReduce interactivo: `pig -x` o `pig`
      - Se ejecuta un clúster de Hadoop.
      - Es el modo por defecto.
    - Ejecutar un script Pig: `pig -x local miscript.pig` o `pig miscript.pig`
      - Es un fichero de texto con comandos Pig.
      - Pueden ser ejecutados en modo local o MapReduce.

Interfaces de consulta de alto nivel

- PigLatin (1/18)

https://pig.apache.org/docs/r0.17.0/basic.html

u.item

```
1|Toy Story (1995)|01-Jan-1995||http://us.imdb.com/M/title-exact?Toy%20Story%20(1995)|0|0|0|1|1|1|0|0|0|0|0|0|0|0|0|0|0|0
2|GoldenEye (1995)|01-Jan-1995||http://us.imdb.com/M/title-exact?GoldenEye%20(1995)|0|1|1|0|0|0|0|0|0|0|0|0|0|0|1|0|0
3|Four Rooms (1995)|01-Jan-1995||http://us.imdb.com/M/title-exact?Four%20Rooms%20(1995)|0|0|0|0|0|0|0|0|0|0|0|0|0|0|1|0|0
4|Get Shorty (1995)|01-Jan-1995||http://us.imdb.com/M/title-exact?Get%20Shorty%20(1995)|0|1|0|0|0|1|0|0|1|0|0|0|0|0|0|0|0
5|Copycat (1995)|01-Jan-1995||http://us.imdb.com/M/title-exact?Copycat%20(1995)|0|0|0|0|0|0|1|0|1|0|0|0|0|0|0|1|0|0
6|Shanghai Triad (Yao a yao yao dao waipo qiao) (1995)|01-Jan-1995||http://us.imdb.com/
Title?Yao+a+yao+yao+dao+waipo+qiao+(199
7|Twelve Monkeys (1995)|01-Jan-1995||ht
8|Babe (1995)|01-Jan-1995||http://us.im
9|Dead Man Walking (1995)|01-Jan-1995||
10|Richard III (1995)|22-Jan-1996||http
11|Seven (Se7en) (1995)|01-Jan-1995||ht
```

u.data

1	196	242	3	881250949
2	186	302	3	891717742
3	22	377	1	878887116
4	244	51	2	880606923
5	166	346	1	886397596
6	298	474	4	884182806
7	115	265	2	881171488
8	253	465	5	891628467
9	305	451	3	886324817
10	6	86	3	883603013

Interfaces de consulta de alto nivel

- PigLatin (2/18)

https://pig.apache.org/docs/r0.17.0/basic.html

```
movies = LOAD 'hdfs:///user/vagrant/movielens/ml-100k/u.item'
         USING PigStorage('|')
         AS (movie_id:INT, movie_title:CHARARRAY, release_date:CHARARRAY, video_release_date:CHARARRAY,
             IMDB_URL:CHARARRAY, unknown:INT, Action:INT, Adventure:INT, Animation:INT, Children:INT, Comedy:INT,
             Crime:INT, Documentary:INT, Drama:INT, Fantasy:INT, FilmNoir:INT, Horror:INT, Musical:INT, Mystery:INT,
             Romance:INT, SciFi:INT, Thriller:INT, War:INT, Western:INT);

ratings = LOAD 'hdfs:///user/vagrant/movielens/ml-100k/u.data'
          AS (user_id:INT, movie_id:INT, rating:INT, timestamp:CHARARRAY);

years = FOREACH movies GENERATE movie_id, GetYear(ToDate(release_date,'dd-MMM-yyyy')) AS year:INT;
years_ratings = JOIN ratings BY movie_id, years BY movie_id;
gyear_rating = GROUP years_ratings BY year;
year_rating = FOREACH gyear_rating GENERATE
                          group AS year,
                          AVG(years_ratings.ratings::rating) AS rating,
                          COUNT(years_ratings.ratings::rating);

STORE year_rating INTO 'output/year_rating';
```

Interfaces de consulta de alto nivel

- PigLatin (3/18)

https://pig.apache.org/docs/r0.17.0/basic.html

input.csv

```
1  SFO,2008,1,1,90,100,65
2  LAX,2008,1,2,89,111,67
3  SFO,2008,1,1,90,100,65
4  LAX,2008,1,2,89,111,67
5  DEN,2008,1,3,88,123,67
6  LAX,2009,10,1,12,132,34
7  DEN,2007,12,12,90,111,11
```

```
data_without_schema = LOAD 'input.csv'
          USING PigStorage(',');

DESCRIBE data_without_schema;

DUMP data_without_schema;
```

```
grunt> DESCRIBE data_without_schema;
Schema for data_without_schema unknown.
```

```
grunt> DUMP data_without_schema;
(SFO,2008,1,1,90,100,65)
(LAX,2008,1,2,89,111,67)
(SFO,2008,1,1,90,100,65)
(LAX,2008,1,2,89,111,67)
(DEN,2008,1,3,88,123,67)
(LAX,2009,10,1,12,132,34)
(DEN,2007,12,12,90,111,11)
```

Interfaces de consulta de alto nivel

- PigLatin (4/18) https://pig.apache.org/docs/r0.17.0/basic.html
  - Procedural, no declarativo como SQL.
    - Plan de ejecución explícito, en SQL inferido.
  - Permite producir múltiples resultados.
    - Describe grafos acíclicos dirigidos (DAG, en inglés)
  - Evaluación tardía.
    - Espera a necesitar los datos para ejecutar las sentencias y así optimizar el proceso.
  - Abstrae al usuario del código Java que rige el proceso.

Interfaces de consulta de alto nivel

- PigLatin (5/18) https://pig.apache.org/docs/r0.17.0/basic.html
  - Carga de datos
    - LOAD '[fichero]' USING [formato] AS ([campo:tipo], ...)
      - Devuelve una relación (conjunto de tuplas) con los datos leídos.
      - Utiliza una clase que define el formato del fichero a leer y parámetros:
        - PigStorage (por defecto), BinStorage, JsonLoader, TextLoader, HBaseStorage, AvroStorage, …
        - Es posible implementar una clase para leer otros formatos.
      - Listados de campos opcional. Define el esquema de los datos.

Interfaces de consulta de alto nivel

- PigLatin (6/18) https://pig.apache.org/docs/r0.17.0/basic.html
  - Transformación de los datos (1/4)
    - GROUP/COGROUP [relacion] BY [clave], ...
      - Agrupa una o varias relaciones por los valores de un campo.
    - [relación] UNION [relación]
      - Une las tuplas de varias relaciones.
    - JOIN [relación] BY [clave], [relación] BY [clave], …
      - Une varias relaciones por el valor de un campo.

Interfaces de consulta de alto nivel

- PigLatin (7/18) https://pig.apache.org/docs/r0.17.0/basic.html
  - Transformación de los datos (2/4)
    - JOIN [relación] (LEFT, RIGHT o FULL) BY [clave], [relación] BY [clave]
      - Une dos relaciones incluyendo tuplas que sólo están en una relación.
    - SPLIT [relación] INTO [relación] IF [condición], …
      - Divide una relación en varias a partir de una condición.
    - FILTER [relación] BY [expresión]
      - Filtra las tuplas.

Interfaces de consulta de alto nivel

- PigLatin (8/18) https://pig.apache.org/docs/r0.17.0/basic.html
  - Transformación de los datos (3/4)
    - FOREACH [relación] GENERATE [expresión] AS [nombre], …
      - Crea una proyección de los campos de las tuplas.
      - Permite agregar datos procedentes de una agrupación.
    - ORDER [relación] BY [orden]
      - Ordena las tuplas por el criterio dado.
    - SAMPLE [relación] [tamaño]
      - Obtiene una muestra aleatoria de los datos.

Interfaces de consulta de alto nivel

- PigLatin (9/18) https://pig.apache.org/docs/r0.17.0/basic.html
  - Transformación de los datos (4/4)
    - LIMIT [relación] [tamaño]
      - Limita el tamaño de la relación al tamaño indicado.
    - DISTINCT [relación]
      - Elimina tuplas duplicadas.

Interfaces de consulta de alto nivel

- PigLatin (10/18) https://pig.apache.org/docs/r0.17.0/basic.html
  - Ejecutar código externo (1/2)
    - Funciones definidas por el usuario.
      - Código Java para transformar los datos.
      - Compilar y crear JAR.
    - MAPREDUCE [jar] STORE [relación] INTO [ruta] LOAD [ruta] AS [campos] `[parámetros]`
      - Ejecuta un trabajo MapReduce sobre una relación.

Interfaces de consulta de alto nivel

- PigLatin (11/18) https://pig.apache.org/docs/r0.17.0/basic.html
  - Ejecutar código externo (2/2)
    - STREAM [relación] THROUGH `[script]`
      - Permite hacer pasar los datos de la relación por un script de línea de comandos.
      - Es necesario distribuir el script en todas las máquinas.
        - Se puede hacer a través de HDFS.

Interfaces de consulta de alto nivel

- PigLatin (12/18)

https://pig.apache.org/docs/r0.17.0/basic.html

  - Comandos de depuración
    - DESCRIBE [relación]
      - Muestra la estructura de la relación.
    - DUMP [relación]
      - Muestra los datos de la relación.
    - EXPLAIN [relación]
      - Indica el proceso realizado para obtener la relación.
    - ILLUSTRATE [relación]
      - Muestra tuplas representativas de los distintos pasos que se han dado para obtener la relación.

Interfaces de consulta de alto nivel

- PigLatin (13/18) https://pig.apache.org/docs/r0.17.0/basic.html
  - Escritura de datos
    - STORE [relación] INTO '[fichero]' USING [formato]
      - Almacena la relación dada en un fichero.
      - Soporta diferentes formatos:
        - PigStorage, BinStorage, JsonStorage, PigDumps, HBaseStorage, …
        - Es posible implementar una clase para escribir en otros formatos.

Interfaces de consulta de alto nivel

- PigLatin (14/18) https://pig.apache.org/docs/r0.17.0/basic.html
  - Permite definir dos tipos de datos: simples y complejos.
    - Tipos simples: int, long, float, double, chararray (array de strings codificados en UTF8), bytearray (array de bytes), boolean, datetime y null (los nulos se implementan usando la definición de SQL de null, como desconocido o inexistente).
    - Tipos complejos: tuple, bag y map.

Interfaces de consulta de alto nivel

- PigLatin (15/18) https://pig.apache.org/docs/r0.17.0/basic.html
  - Tipos de datos complejos (1/3)
    - TUPLE
      - Es un conjunto ordenado de campos (pueden ser de cualquier tipo de datos, incluso TUPLE o BAG).
      - Similar a las filas en una tabla.
      - Sintaxis: (field [, field …])
      - Ejemplo de TUPLE con dos campos: (‘Juan’, 32)

Interfaces de consulta de alto nivel

- PigLatin (16/18) https://pig.apache.org/docs/r0.17.0/basic.html
  - Tipos de datos complejos (2/3)
    - BAG
      - Es una colección de tuplas (TUPLE) ordenadas.
      - Sintaxis: { tuple [, tuple …] }
      - Similar a las tablas de un SGBDR (*RDBMS*)
      - Ejemplo de BAG con tres tuplas, cada una con dos campos:

        { (‘jj’, 32), (‘alvaro’, 27), (‘cris’, 31) }

Interfaces de consulta de alto nivel

- PigLatin (17/18) https://pig.apache.org/docs/r0.17.0/basic.html
  - Tipos de datos complejos (3/3)
    - MAP
      - Es un conjunto de pares clave-valor.
      - Sintaxis: [clave#valor <,clave#valor …>]
      - Ejemplo de MAP: [1#rojo, 2#azul, 3#amarillo]

Interfaces de consulta de alto nivel

- PigLatin (18/18)

https://pig.apache.org/docs/r0.17.0/basic.html

  - Otros conceptos
    - Salir de PigLatin:
      - quit
    - Ocultar logs de ejecución de los trabajos:
      - pig -4 nologconf

```
bigdata@bigdata:~$ pig -4 nolog.conf
2020-01-17 19:00:59,271 INFO pig.ExecTypeProvider: Trying ExecType : LOCAL
2020-01-17 19:00:59,273 INFO pig.ExecTypeProvider: Trying ExecType : MAPREDUCE
2020-01-17 19:00:59,273 INFO pig.ExecTypeProvider: Picked MAPREDUCE as the ExecType
2020-01-17 19:00:59,453 INFO pig.Main: Loaded log4j properties from file: nolog.conf
grunt> data_without_schema = LOAD 'input.csv'
>>          USING PigStorage(',');
grunt> dump data_without_schema
(SFO,2008,1,1,90,100,65)
(LAX,2008,1,2,89,111,67)
(SFO,2008,1,1,90,100,65)
(LAX,2008,1,2,89,111,67)
(DEN,2008,1,3,88,123,67)
(LAX,2009,10,1,12,132,34)
(DEN,2007,12,12,90,111,11)
```

nologconf

```
log4j.rootLogger=fatal
```

Demostración

Ejemplo de Pig

Interfaces de consulta de alto nivel

- Apache Hive (1/8) https://hive.apache.org
  - Facebook detectó que obtenía mejores resultados con algoritmos sencillos sobre muchos datos que algoritmos complejos sobre pocos datos.
    - Empezaron a utilizar Hadoop, pero les costaba mucho encontrar programadores que pudieran aprovecharlo.
    - Decidieron implementar un sistema sobre Hadoop que permitiera hacer consultas basado en SQL.

Interfaces de consulta de alto nivel

- Apache Hive (2/8) https://hive.apache.org
  - Creado en 2007 por Facebook, en 2008 pasa a Apache y en 2010 pasa a ser de primer nivel en Apache.
  - Utiliza el lenguaje HiveQL, similar a SQL pero no cumple el estándar SQL-92.
    - Muchas sentencias son exactamente iguales a SQL.
    - Permite aprovechar conocimientos y código desarrollados sobre bases de datos relacionales para acceder a datos HDFS.
    - Convierte sentencias SQL a MapReduce.

Interfaces de consulta de alto nivel

- Apache Hive (3/8) https://hive.apache.org
  - Define el esquema de los datos en la lectura.
    - Almacena metadatos en una base de datos embebida.
    - Los datos se almacenan en ficheros de texto o binarios.
  - Permite crear índices sobre los datos para mejorar el rendimiento de las consultas.
  - Permite consultar y manejar datos estructurados y semiestructurados.

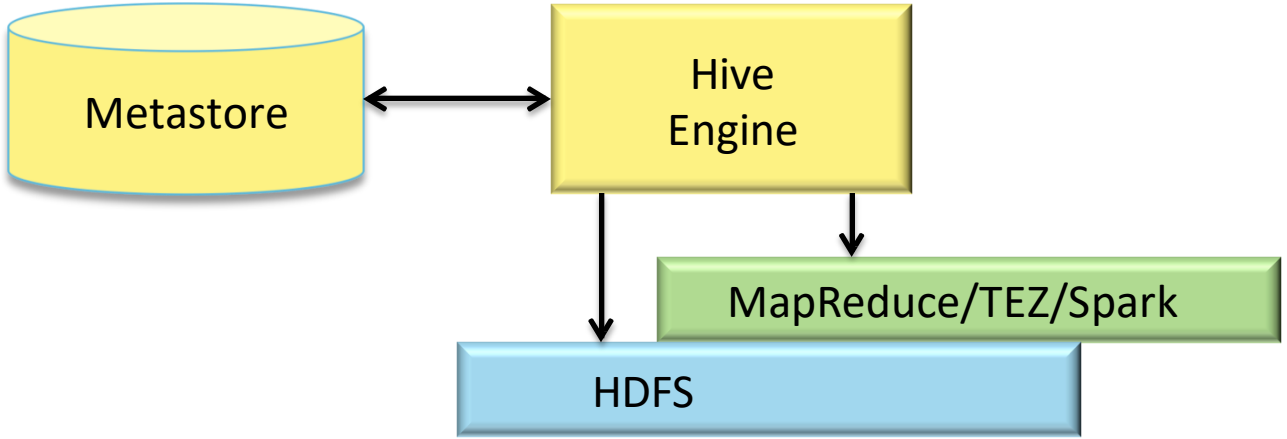
Interfaces de consulta de alto nivel

- Apache Hive (4/8) https://hive.apache.org
  - Usa alguno de los motores de ejecución sobre YARN.
  - Usa HDFS (o HBase) como almacenamiento.
  - Consiste en:
    - Metastore
      - Almacenar información de metadatos.
      - Provee la información de estructura de tabla a los datos almacenados.
    - Hive Engine: procesamiento, compilación, optimización y ejecución de consultas.

Interfaces de consulta de alto nivel

- Apache Hive (5/8)

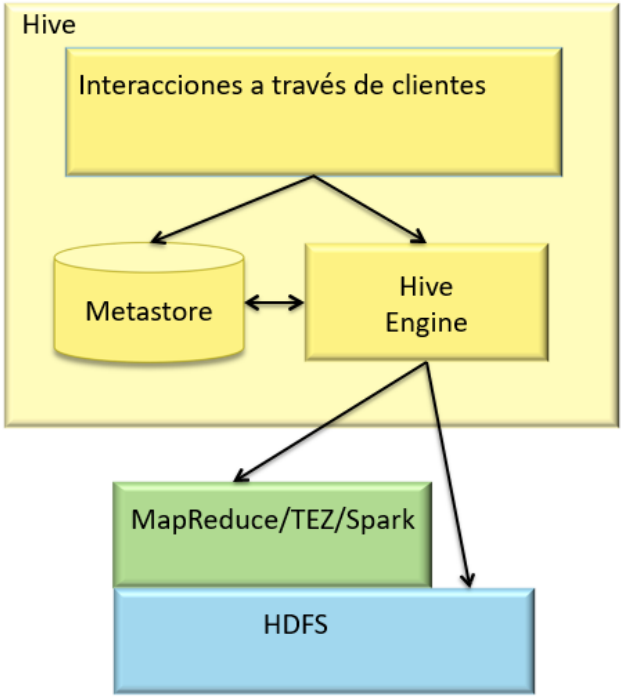
https://hive.apache.org



Interfaces de consulta de alto nivel

- Apache Hive (6/8)
  - Arquitectura

https://hive.apache.org



Interfaces de consulta de alto nivel

- Apache Hive (7/8)

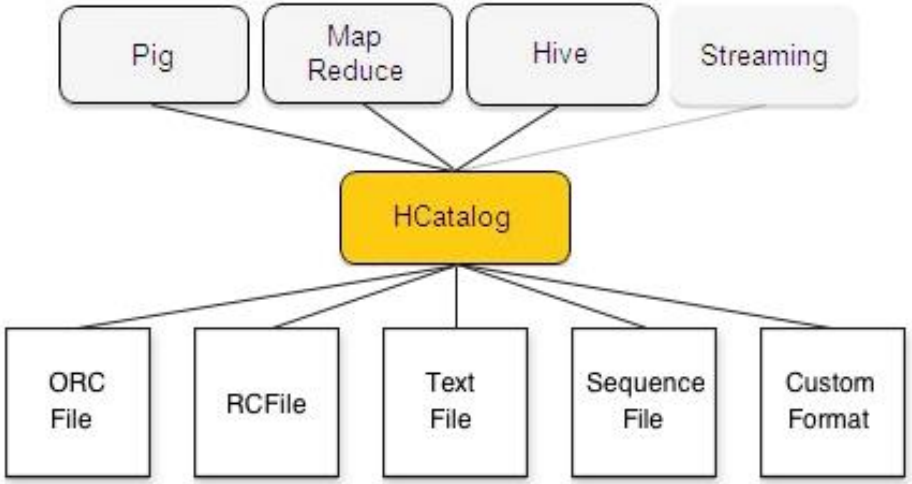
https://hive.apache.org

  - HCatalog (1/2)
    - Servicio que permite guardar metadatos para que distintas aplicaciones de Hadoop puedan usarla.
    - Proyecto que surgió de forma independiente y posteriormente se unió a Hive.
    - En Hive se utiliza el término Metastore.

Interfaces de consulta de alto nivel

- Apache Hive (8/8)
  - HCatalog (2/2)

https://hive.apache.org



Interfaces de consulta de alto nivel

- HiveQL (1/9)

https://hive.apache.org

u.item

```
1|Toy Story (1995)|01-Jan-1995||http://us.imdb.com/M/title-exact?Toy%20Story%20(1995)|0|0|0|1|1|1|0|0|0|0|0|0|0|0|0|0|0|0
2|GoldenEye (1995)|01-Jan-1995||http://us.imdb.com/M/title-exact?GoldenEye%20(1995)|0|1|1|0|0|0|0|0|0|0|0|0|0|0|1|0|0
3|Four Rooms (1995)|01-Jan-1995||http://us.imdb.com/M/title-exact?Four%20Rooms%20(1995)|0|0|0|0|0|0|0|0|0|0|0|0|0|0|1|0|0
4|Get Shorty (1995)|01-Jan-1995||http://us.imdb.com/M/title-exact?Get%20Shorty%20(1995)|0|1|0|0|0|1|0|0|1|0|0|0|0|0|0|0|0
5|Copycat (1995)|01-Jan-1995||http://us.imdb.com/M/title-exact?Copycat%20(1995)|0|0|0|0|0|0|1|0|1|0|0|0|0|0|0|1|0|0
6|Shanghai Triad (Yao a yao yao dao waipo qiao) (1995)|01-Jan-1995||http://us.imdb.com/
Title?Yao+a+yao+yao+dao+waipo+qiao+(1995)|0|0|0|0|0|0|0|0|1|0|0|0|0|0|0|0|0|0
7|Twelve Mo
8|Babe (199
9|Dead Man
10|Richard
11|Seven (S
```

```
CREATE TABLE movies (movie_id int, movie_title string, release_date  string, video_release_date  string,
IMDB_URL  string, unknown int, Action int, Adventure int, Animation int, Childrens int, Comedy int,
Crime int, Documentary int, Drama int, Fantasy int, FilmNoir int, Horror int, Musical int, Mystery int,
Romance int, SciFi int, Thriller int, War int, Western int)
  ROW FORMAT DELIMITED
  FIELDS TERMINATED BY '|'
  STORED AS TEXTFILE;

LOAD DATA INPATH 'hdfs:///user/vagrant/movielens/ml-100k/u.item' OVERWRITE INTO TABLE movies;

SELECT * FROM movies WHERE Drama = 1;
```

https://cwiki.apache.org/confluence/display/Hive/LanguageManual

Interfaces de consulta de alto nivel

- HiveQL (2/9) https://hive.apache.org
  - Utiliza los conceptos de BBDD relacionales:
    - Tablas, columnas, vistas, etc.
  - Diseño para manejar datos estructurados.
    - Tiene algunas variaciones respecto al SQL de una BBDD relacional.
  - Traduce las sentencias SQL en programas ejecutables en YARN.

https://cwiki.apache.org/confluence/display/Hive/LanguageManual

Interfaces de consulta de alto nivel

- HiveQL (3/9)

https://hive.apache.org

  - Soporta casos de uso como:
    - Consultas ad-hoc.
    - Agregaciones, sumarizaciones, UDF, etc.
    - Es la herramienta más usada para Data Analysis en Hadoop.

https://cwiki.apache.org/confluence/display/Hive/LanguageManual

Interfaces de consulta de alto nivel

- HiveQL (4/9)

https://hive.apache.org

  - Jerarquía de entidades
    - ***BBDD***: espacio de nombre para evitar conflictos de nombre de tablas y permite también asignar permisos a nivel de usuario.
    - ***Tablas***: unidades de datos homogéneas formadas por filas y columnas con la misma metadata.
    - ***Particiones***: unidad de almacenamiento dentro de una tabla que permite agrupar registros que comparten algún criterio
    - ***Buckets***: agrupación de registros de acuerdo al valor de un campo dentro de una partición.

https://cwiki.apache.org/confluence/display/Hive/LanguageManual

Interfaces de consulta de alto nivel

- HiveQL (5/9)

https://hive.apache.org

  - Tablas (1/4)
    - Consiste en:
      - *Datos*: usualmente uno o más archivos en HDFS.
      - *Schema*: en la forma de metadatos guardados en algún repositorio donde Hive tiene acceso.

https://cwiki.apache.org/confluence/display/Hive/LanguageManual

Interfaces de consulta de alto nivel

- HiveQL (6/9)

https://hive.apache.org

  - Tablas (2/4)
    - Esto implica:
      - Schema y datos están separados.
        - Se puede definir un schema para datos existentes.
        - Los datos pueden agregar o procesarse independientemente.
        - Hive puede "apuntar" a datos que ya existían en cualquier lugar en HDFS (tablas externas).
      - Se debe definir los metadata para acceder a datos que se encuentran en HDFS o para datos que serán insertados usando Hive.

https://cwiki.apache.org/confluence/display/Hive/LanguageManual

Interfaces de consulta de alto nivel

- HiveQL (7/9)

https://hive.apache.org

  - Tablas (3/4)
    - Crear una tabla:

```
CREATE TABLE mitabla (nombre chararray, edad int) ROW
  FORMAT DELIMITED
  FIELDS TERMINATED BY ','
  STORED AS TEXTFILE;
```

      - ROW FORMAT son comandos de Hive que indican que los datos de una tabla están delimitados.
      - Junto a CREATE TABLE se pueden especificar particiones y buckets.

https://cwiki.apache.org/confluence/display/Hive/LanguageManual

Interfaces de consulta de alto nivel

- HiveQL (8/9)

https://hive.apache.org

  - Tablas (4/4)

```
CREATE TABLE mitabla
```

```
PARTITIONED BY (nombre chararray, edad int)
```

```
CLUSTERED BY (nombre) INTO N BUCKETS
```

    - Otras operaciones
      - SHOW TABLE.
      - CREATE/DROP TABLE.
      - ALTER TABLE.

https://cwiki.apache.org/confluence/display/Hive/LanguageManual

Interfaces de consulta de alto nivel

- HiveQL (9/9)

https://hive.apache.org

  - Otros conceptos
    - Salir de HiveQL:
      - exit;
      - quit;
    - Borrar pantalla:
      - !clear;
    - Ocultar logs de ejecución de los trabajos:
      - set hive.server2.logging.operation.level=NONE;

https://cwiki.apache.org/confluence/display/Hive/LanguageManual

Interfaces de consulta de alto nivel

- Particiones (1/2)

https://hive.apache.org

  - Hive hace full scans sobre una tabla cada vez que se ejecuta una consulta sobre ella.
  - Las particiones en Hive permiten que solo se procesen las particiones afectadas en una consulta.
  - La partición es un concepto lógico pero tiene una consecuencia en el almacenamiento físico de los datos.
  - La división se realiza utilizando "*claves de partición*".

Interfaces de consulta de alto nivel

- Particiones (2/2)

https://hive.apache.org

  - Las particiones pueden estar dentro del directorio donde está definida la tabla (internas) o en cualquier ubicación (externas).
  - Las columnas de partición no son parte de los datos almacenados.

Interfaces de consulta de alto nivel

- Buckets (1/2)

https://hive.apache.org

  - Otro método de segregar los datos de una tabla además de particiones.
  - La idea es generar un archivo o directorio que contiene todos los registros que comparten el mismo valor de una columna.

```
CREATE TABLE weblog (userID INT, url STRING, source_ip STRING) PARTITIONED BY (dt STRING) CLUSTERED BY
(userID) INTO 96 BUCKETS
```

Interfaces de consulta de alto nivel

- Buckets (2/2)

https://hive.apache.org

  - Es responsabilidad del programador insertar los datos correctamente (en la partición y la cantidad de buckets definidos).

```
SET hive.enforce.bucketing = true
SET mapreduce.job.reduces = 96
```

Interfaces de consulta de alto nivel

- Ejemplo de uso de particiones (1/3) https://hive.apache.org

```
hive> create table allstates(state string, District string,Enrolments string)
    > row format delimited
    > fields terminated by ',';
OK
```

1

Creation of table "allstates"

```
hive> load data local inpath '/home/hduser/AllStates.csv' into table allstates;
Loading data to table default.allstates
Table default.allstates stats: [numFiles=1, totalSize=36913]
OK
Time taken: 1.459 seconds
hive> create table state_part(District string,Enrolments string) PARTITIONED BY(state string);
OK
Time taken: 0.199 seconds
```

2

Loading Data into "allstates"

3

Creation of partition table "state_part"

Interfaces de consulta de alto nivel

- Ejemplo de uso de particiones (2/3) https://hive.apache.org

Interfaces de consulta de alto nivel

- Ejemplo de uso de particiones (3/3) https://hive.apache.org

```
hduser@datamatics-Ubuntu:/usr/local/hadoop-1.2.1/bin$ ./hadoop dfs -ls /user/hive/warehouse/state_part
Warning: $HADOOP_HOME is deprecated.

Found 38 items
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Andaman_and_Nicobar_Island
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Andhra_Pradesh
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Arunachal_Pradesh
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Assam
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Bihar
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Chandigarh
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Chhattisgarh
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Dadra_and_Nagar_Haveli
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Daman_and_Diu
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Delhi
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Goa
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Gujarat
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Haryana
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Himachal_Pradesh
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Jammu_and_Kashmir
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Jharkhand
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Karnataka
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Kerala
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Lakshadweep
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Madhya_Pradesh
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Maharashtra
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Manipur
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Meghalaya
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Mizoram
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Nagaland
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Odisha
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Others
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Puducherry
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Punjab
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Rajasthan
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Sikkim
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Tamil_Nadu
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Tripura
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Uttar_Pradesh
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=Uttarakhand
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=West_Bengal
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=california
drwxr-xr-x   - hduser supergroup          0 2015-11-04 16:16 /user/hive/warehouse/state_part/state=newyork
```

7

Total 38 states present in table

38 partition tables stored in HDFS system

Interfaces de consulta de alto nivel

- Ejemplo de uso de buckets (1/2)

https://hive.apache.org

```
hive>create table samplebucket {first_name      string,
                                job_id          int,
                                department      string,
                                salary          string,
                                country         string }
                  clustered by (country) into 4 buckets
                  row format delimited
                  fields terminated by ',';
```

column names

Creating 4 buckets

```
from employees
     insert overwrite table samplebucket
     select first_name,job_id, department,salary, country;
```

loading Data

Interfaces de consulta de alto nivel

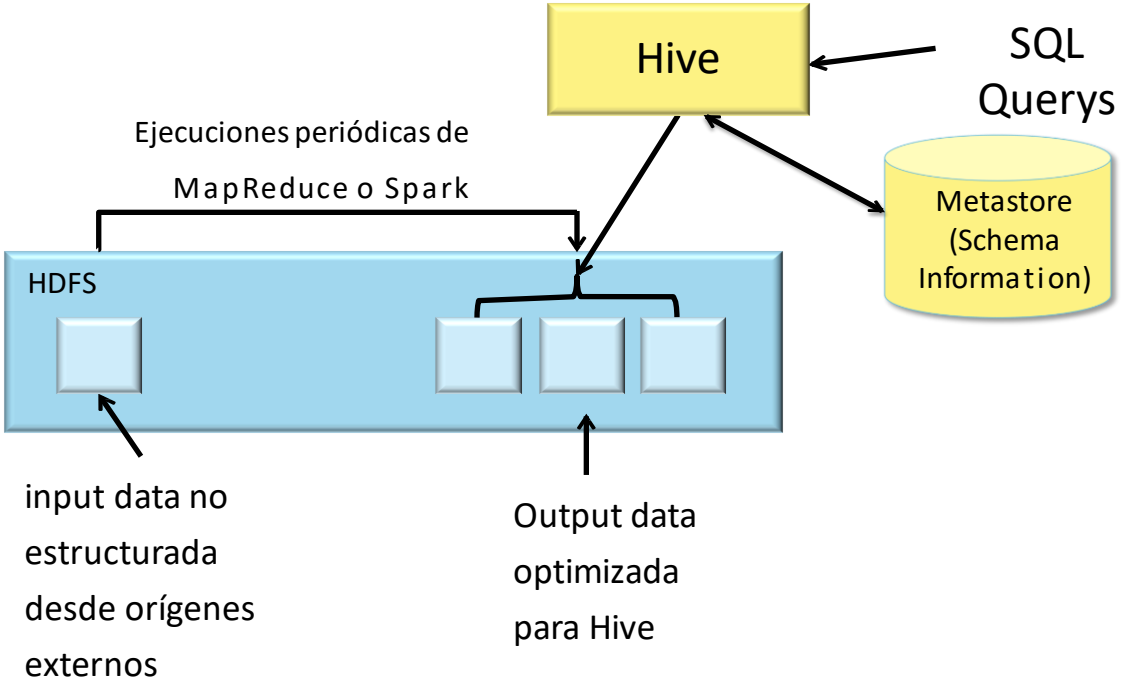
- Ejemplo de uso de buckets (2/2)

https://hive.apache.org

Interfaces de consulta de alto nivel

- Caso de uso clásico de Hive

https://hive.apache.org

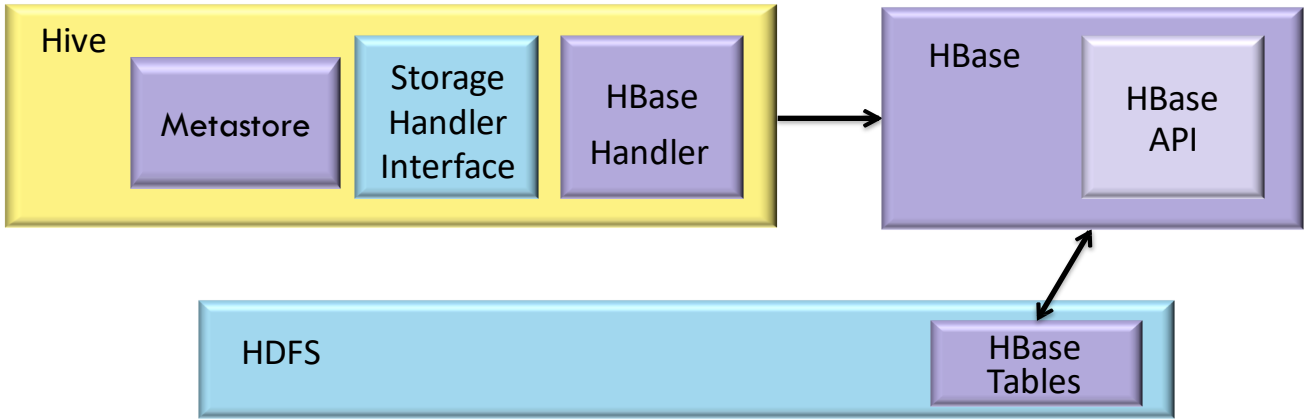


Interfaces de consulta de alto nivel

- Hive con HBase

https://hive.apache.org

  - Hive tiene integrado nativamente HDFS.
  - Hive incluye "storage handlers" para HBase.
  - A través de estos, Hive puede leer y escribir en HBase.



Interfaces de consulta de alto nivel

- Pig vs Hive

	Pig	Hive
Desarrollado por	Yahoo!	Facebook
Lenguaje	Pig Latin	HiveQL
Join/Order/Sort	Sí	Sí
Esquema opcional	Sí	Sí
Turing completo	Sí, cuando se usan UDFs en Java	
Tiempo de desarrollo	Corto/medio	Corto
Rendimento	Medio/Bajo (*)	Bajo (*)

(*) Comparado con MapReduce

Demostración

Ejemplo de Hive