

5.- Base de datos NoSQL: Column Family

En esta práctica probaremos algunas funcionalidades básicas de una instalación Standalone de sistema de almacenamiento NoSQL de tipo Column Family Apache [HBase](#). En concreto, nos centraremos en el uso de la Shell de HBase y de su API Java para realizar operaciones de lectura y escritura de datos.

5.1 Instalación de la versión standalone de HBase

La versión 2.3.5 de HBase ya está instalada y lista para usarse en modo standalone en la máquina "GreiBD", con lo que no necesitaremos instalar software adicional para realizar esta práctica. Se incluye aquí la secuencia de pasos para realizar esta instalación por si se quieren repetir en otra máquina. La instalación de la versión distribuida no es obligatoria en esta práctica.

En primer lugar descargamos la última versión estable del sistema desde la siguiente URL: <https://www.apache.org/dyn/closer.lua/hbase/>

Elegimos el mirror recomendado y accedemos a la versión "stable" para descargar el archivo "bin". Extraemos el archivo comprimido en la máquina.

Para poder ejecutar HBase necesitamos la versión 8 o 11 del JDK de Java. Las [compatibilidades](#) entre Java y HBase para las distintas versiones pueden verse en la documentación de HBase.

Configuramos ahora la variable de entorno JAVA_HOME. Para esto, primero debemos encontrar la localización de la instalación de Java.

```
update-java-alternatives -l
```

Modificamos el archivo "conf/hbase_env.sh" para que contenga la siguiente línea.

```
export JAVA_HOME=path del jdk
```

Ya podemos iniciar HBase con el siguiente comando

```
bin/start-hbase.sh
```

Podemos comprobar si el proceso está iniciado con "jps".

La interfaz web está disponible en "http://localhost:16010"

La Shell de HBase está disponible en:

```
bin/hbase shell
```

En esta URL tenemos un buen resumen de los comandos de la Shell: <https://learnhbase.wordpress.com/2013/03/02/hbase-shell-commands/>

Para detener el servidor

```
bin/stop-hbase.sh
```

?

5.2 Creación del esquema e inserción de datos de prueba

Empezaremos creando una tabla para almacenar datos de prueba para realizar después consultas con el API de Java. La documentación de la API está disponible en la página de HBase: <https://hbase.apache.org/>

```
create 'películas', 'info', 'reparto', 'personal'
```

La forma habitual de inserción de datos en HBase es a través de un proceso MapReduce. En este caso, al no disponer de una instalación de Hadoop y HDFS realizaremos la inserción a través de un proceso Java, usando la API. Para esto, en la clase "test1.java" del código de ejemplo, nos aseguramos de ejecutar el método "testInsercionHBase()".

Comandos de la shell (En la shell solo vamos a poder ver correctamente columnas en las que hemos introducido texto, no aquellas que son binarias).

```
exit (Salir de la shell)
help (Ayuda - Lista de comandos)
help "create" (ayuda sobre el comando create)
list (lista de tablas)

create 'tabla', 'columnfamily', 'columnfamily', ...
describe 'tabla'

disable 'tabla' (desactiva la tabla. Para realizar ciertas operaciones es necesarioa desactivar)
enable 'tabla'
is_enabled 'tabla'

drop 'tabla' (eliminar la tabla. Primero hay que desactivarla)
alter 'tabla', NAME='columnfamily', versions=>5 (cambia el esquema de la tabla)
alter 'tabla', NAME='columnfamily', METHOD=>'delete' (borra un column family)

count 'tabla' (número de filas)
put 'tabla', 'fila', 'cf:columna', 'valor', instante (instante en milisegundos desde 1970)
get 'tabla', 'fila' (obtiene una fila)
get 'table', 'fila', {COLUMN=>'columnfamily'} (obtiene solo una columnfamily)
delete 'tabla', 'fila', 'columnfamily'
truncate 'tabla' (eliminar y vuelve a crear la tabla vacía)
scan 'tabla' (muestra todas las filas)
scan 'tabla', {COLUMNS =>['columnfamily', 'columnfamily'], LIMIT=>10, VERSIONS=>5}
```

Ejercicios

Programa usando el API de Java de HBase, dos nuevos métodos de las clases DAOPeliculasHBase y DAOPeliculasPgsql para obtener cada uno de los resultados siguientes. Intenta filtrar toda la información que puedas en el servidor proporcionando filtros:

<https://hbase.apache.org/book.html#client.filter>

E1.- Obtener los datos de la película con identificador 1865

E2.- Obtener el reparto de la película "Avatar"

E3.- Obtener los datos básicos (column family "info") de las películas que tengan un presupuesto mayor de 250 millones

E4.- Obtener el título y el protagonista principal (orden = 0, obtener el nombre y el personaje) de las películas de enero del "2015"

E5.- Obtener el listado de los títulos de las películas dirigidas por "Ridley Scott"

Última modificación: xoves, 11 de novembro de 2021, 16:06