**ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL**

**SISTEMA DE BASE DE DATOS AVANZADOS**

**II Término, 2021**

**INTEGRANTES**

* Juan José Loor
* Andrea Soriano

**PROYECTO DEL SEGUNDO PARCIAL**

Manual de Usuario

**Objetivos del proyecto**

Crear un cubo de información con el dataset de AdventureWorks

**Descripción del dataset**

Adventure Works es una empresa que vende bicicletas y partes de bicicletas a clientes y distribuidores. Debido a su gran crecimiento y demanda, la compañía está buscando la implementación de una solución que les permita a los gerentes tomar decisiones para su expansión global, este dataset cuenta con la siguiente estructura: tabla de hechos factinternetsales y las dimensiones: dimcustomer , dimcurrency, dimdate, dimsalesterritory, dimproduct ,dimgeography, dimproductcategory, dimproductsubcategory.

**Instalación del apache kylin**

Para instalar apache Kylin nosotros hicimos uso de la guía “Run kylin with Docker” la cual básicamente nos permitía correr kylin a través de una imagen en un contender de Docker la cual tenía implementado y desplegado las siguientes herramientas:

* JDK 1.8
* Hadoop 2.7.0
* Hive 1.2.1
* Hbase 1.1.2 (with Zookeeper)
* Spark 2.3.1
* Kafka 1.1.1
* MySQL 5.1.73

1. Traer la imagen del docker hub la cual contiene todas las herramientas antes descritas con el comando :

docker pull apachekylin/apache-kylin-standalone:3.1.0

1. Una vez traída la imagen, correr el siguiente comando para iniciar el contenedor:

docker run -d \

-m 8G \

-p 7070:7070 \

-p 8088:8088 \

-p 50070:50070 \

-p 8032:8032 \

-p 8042:8042 \

-p 16010:16010 \

apachekylin/apache-kylin-standalone:3.1.0

1. Dado que hemos mapeado el puerto especificado en el contenedor al puerto local, podemos abrir las páginas de cada servicio directamente en el navegador, como:

* Kylin Web UI: http://127.0.0.1:7070/kylin/login
* Hdfs NameNode Web UI: http://127.0.0.1:50070
* Yarn ResourceManager Web UI: http://127.0.0.1:8088
* HBase Web UI: <http://127.0.0.1:16010>

**Importación de datos**

1. Clonar el repositorio que contiene los datos a usar, para esto correr en la consola en la dirección que queramos traer los archivos el siguiente comando:

git clone <https://github.com/vanechev/ejemplo-dw-kylin.git>

1. Instalar la herramienta gzip2 la cual nos ayudara a descomprimir los archivos que necesitemos, para esto correr el comando:

sudo apt install gzip2

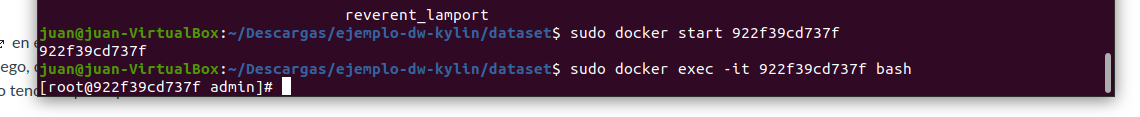
1. Ir a la carpeta que clonamos específicamente a la carpeta dataset la cual contiene un archivo sql, cuando nos encontremos en este directorio correr el comando:

bzip -d adventureworksdw2008.sql.bz2

1. Correr el comando sudo docker ps -a para ver todos los contenedores de docker que están corriendo, aquí encontraremos el contenedor “apachekylin/apache-kylyn-standalone:4.0.0” así como su id de contenedor el cual debemos anotar porque nos servirá en los siguientes pasos.
2. Para iniciar el contenedor antes mencionado correr el comando:

sudo docker start <id\_de\_contenedor>

1. Para poder entrar en el contenedor como tal usar el comando sudo docker exec -it <id\_de\_contenedor> bash el cual nos devolverá básicamente la consola del contenedor.



1. Abrir una nueva terminal e ir a la dirección en la cual extrajimos los datos necesarios en el paso 3, aquí deberíamos tener dos archivos, el archivo sql que queremos cargar y una imagen de esta base
2. Correr el comando sudo docker cp adventureworksdw2008.sql <id\_de\_contenedor>:/home/admin, aquí lo que estamos haciendo, aquí lo que estamos haciendo es copiar el archivo sql dentro del contenedor de docker kylin que tenemos iniciado en la rutina home/admin.
3. Regresamos a la consola del contenedor y corremos el comando mysql -u root -p 123456 para poder entrar al monitor de mysql.
4. Para cargar la base de datos correr el comando mysql -u root -p123456 mysql < adventureworksdw2008.sql
5. Para asegurarnos que hemos cargado la base correctamente correr el comando show databases, nos debería devolver algo así:

Texto

Descripción generada automáticamente

1. Para realizar la carga de datos desde mysql hacia hadoop utilizaremos la herramienta sqoop, por lo cual necesitamos instalarla, para esto abriremos otra terminal.
2. Descargar el archivo **sqoop-1.4.5.bin\_\_hadoop-2.0.4-alpha.tar.gz**.a través del siguiente link <https://sqoop.apache.org/>
3. Correr el comando sudo docker cp sqoop-1.4.5.bin\_\_hadoop-2.0.4-alpha.tar.gz <id\_de\_contenedor>:/home/admin, aquí lo que estamos haciendo es copiar el contenido de este archivo que descargamos al directorio home/admin del contenedor
4. Dentro de la consola del contenedor mover el archivo que le acabamos de pasar a la dirección /ussr/lib/sqoop.
5. Ahora necesitamos editar el bashrc y bash\_profile y asegurarnos que tenga la siguiente estructura export SQOOP\_HOME=/usr/lib/sqoop export PATH=$PATH:$SQOOP\_HOME/bin
6. Correr el comando source ~/.bashrc y source ~/.bash\_profile para actualizar el enviroment.
7. Ahora para cargar los datos desde mysql hacia hadoop utilizando sqoop necesitaremos descargar el mysql-connecto-java-5.1.30
8. Una vez lo hayamos descargado moverlo a la dirección /usr/lib/sqoop/lib
9. Para verificar que tenemos sqoop instalando correctamente ir aa la dirección $SQOOP\_HOME/bin y correr el comando soop-version, deberíamos obtener una ventana similar a esta:

Texto

Descripción generada automáticamente

1. Ahora para completar la carga de datos hacia hadoop correr el siguiente comando $SQOOP\_HOME/bin/sqoop-import --connect jdbc: mysql://localhost:3306/adventureworksdw --username root --table <todas\_las\_tablas\_del\_modelo> --target-dir base/adventureworks/dw/internetsales -P, a continuación empezaría la carga de datos, esperamos un momento y listo, ya podemos visualizar los datos cargados correcamte.

Interfaz de usuario gráfica, Tabla

Descripción generada automáticamenteResultado:

**Creación del modelo**

1. Abrir el archivo hive-create-table.hql que nos fue proporcionado.
2. Realizar las configuraciones necesarias del archivo, sobre escribir la location en todas partes por la ubicación del directorio donde cargamos los datos anteriormente.
3. Una vez cambiada la location en todas las tablas procedemos a crear la base de datos en hive por lo que corremos el comando hive hive-create-table.hql
4. Una vez realizado los pasos anteriores ir a la interfaz de Kylin seleccionar nuestro proyecto, escoger modelo , luego la ventana data source y escogemos el segundo icono ( load table from tree ) , aquí seleccionamos nuestra base de datos y posteriormente todas las tablas.
5. Luego de esto nos dirigimos a la pestana models en la cual vamos a ir a new model y se nos abrirá una ventana emergente con una serie de pasos a seguir.
6. Primero le damos un nombre a nuestro modelo
7. Luego escogemos un fact table , para esto usaremos el factinternalsales
8. Luego en dimensiones escogemos cualquier columna esto realmente no es tan importante porque luego lo editaremos con el archivo model.json que nos fue proporcionado
9. En measures de igual forma coger cualquier columna
10. En settings dejar todo en default y finalmente darle a guardar
11. Ahora una vez creado nuestro modelo vamos a darle a actions seguido de edit(JSON) y vamos a modificarlo con la estructura del archivo model.json que nos fue proporcionada
12. Una vez modificado y guardado deberíamos obtener el modelo listo.

Resultados

Diagrama

Descripción generada automáticamente

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente con confianza media

**Creación del cubo**

1. Para la creación del cubo en la interfaz de Kylin nos vamos a la ventana models y le damos a new cube.
2. Se nos abrirá una ventana emergente con algunos pasos, en el primero escogeremos el modelo creado anteriormente y el nombre del cubo.
3. Luego en dimensiones vamos a escoger las dimensiones que nosotros consideremos convenientes para nuestros cálculos.
4. Luego en measures agregamos las mediciones que vayamos a emplear
5. Luego en refresh settings dejar todo por default
6. Luego en advanced configuration en el engite type del cubo escoger la opción spark.
7. Dejar configuration overwrites por defecto y finalmente guarda
8. Finalmente, en settings dejamos las configuraciones por defecto y le damos a guardar.
9. Ahora iremos a la ventana de model y podremos ver nuestro cubo en estado disabled por lo cual le daremos en actions y luego build para desplegar el cubo
10. Esperamos que se contruya el cubo y listo.

Interfaz de usuario gráfica, Texto

Descripción generada automáticamenteResultado: