**TRASPASO MDM Y DMATRIZCAMPANA**

MDM se está trabajando actualmente con Redshift, S3, Nifi, Scala, EMR y MongoDB

Actualmente se encuentra tanto en QAS como en PRD.

EL proceso comienza cuando MDM deja 3 archivos en un bucket de S3 (belc-bigdata-landing-dlk-prd)

Luego Nifi se ejecuta liberando un Disparador que habilita un .py (python) que administra la ejecución de un “.jar” (el cual está codificado en Scala). Este .jar lee los archivos que se encuentran en S3 y los procesa, leyendo uno por uno y aplicandole reglas que se han solicitado por parte de MDM.

Lo generado se almacena y muestra en 2 interfaces, las cuales son Redshift y MongoDB.

En redshift se dejan en 3 tablas:

* mdm\_producto\_comunicaciones
* mdm\_producto\_sap
* mdm\_producto\_webredes

Ahí se almacena la información para luego consultarla con Redshift. Por otro lado, es cargado en MDM en un schema "info\_product" en el collection "fichaproducto"

La información se almacena en ese Collection y se disponibiliza para que el API pueda consumir la información.

**ACLARACIONES:**

## QAS:

Existe un .jar MDMProduct-QAS-assembly-0.1.jar, el cual está dirigido para el Redshift de QAS y el Mongo de Qas.

Redshift: "10.12.0.94"

Las tablas se encuentran en "fnc\_mdm" las tablas inician con "mdm\_producto\_"

Y apuntan al MONGO QAS: "bigdatainfoservice-0hd3l.mongodb.net"

La collection se encuentra en el database "infoProduct" collection: "fichaproducto"

## PRD

Existe un .jar MDMDatalake-assembly-0.1.jar, el cual está dirigido para el Redshift de PRD y el Mongo de PRD.

Redshift: "10.12.2.26"

Las tablas se encuentran en "fnc\_analitico" las tablas inician con "mdm\_producto\_"

Y apuntan al MONGO PRD: "server: bigdatainfoservice-8x89f.mongodb.net"

La collection se encuentra en el database "infoProduct" collection: "fichaproducto"

Los pasos siguientes son para guardar los cambios en los jar y subirlos a S3:

La carpeta del código se encuentra en el bucket de S3: bel-bigdata-landing-dlk-qas/forecast-data/EMR/MDM.zip

También se encuentra cargado en el EMR de QAS. 10.12.6.150 en la dirección /home/hadoop/jtaco/mdminformation

**MODIFICACIONES AL CODIGO**

### Luego que se hagan las modificaciones dentro de la carpeta mdminformation

Se debe ensamblar la carpeta para generar el nuevo jar dentro de mdminformation:

1. $ cd mdminformation
2. $ sbt assembly
3. Se generará un jar con el nombre de MDMDatalake-assembly-0.1.jar en la siguiente dirección:
   1. /home/hadoop/jtaco/mdminformation/target/scala-2.11/MDMDatalake-assembly-0.1.jar (el nombre del archivo jar dependerá de cual sea la modificación que han hecho, si es para QAS o PRD)

Luego se debe cargar al S3: para el PRD (s3://belc-bigdata-functional-dlk-prd/analitico/jars/) se debe ejecutar el siguiente comando (la ruta dependera si es para un pase a QAS o PRD)

1. $ aws s3 cp /home/hadoop/jtaco/mdminformation/target/scala-2.11/MDMDatalake-assembly-0.1.jar s3://belc-bigdata-functional-dlk-qas/analitico/jars/

Una vez que se encuentre en el bucket de S3, depende si está en el De QAS correrá para ejecutar el Mongo de QAS y si esta en el bucket de S3 de PRD, correrá para el Mongo de PRD.

**ARCHIVOS INPUT**

Todos los días llegan 3 archivos al Bucket de S3: belc-bigdata-landing-dlk-prd/Datalake/input/mdm/

Estos llegan a 3 carpetas: sap , comunicaciones y webredes. Las 3 carpetas tienen una carpeta dentro llamada BackUp donde se almacena el archivo del día luego de haber sido consumido por el proceso diario

**CONFIGURACION NIFI**

Las configuracfiones para la ejecución del archivo Jar se hacen en Nifi (10.12.6.86), ya que en este se ejecutan 2 Livys, 1 para EMR QAS y otro para EMR PRD.

El flujo se encuentra en el Process Group: MDM Producto- Dmatriz

Y en la carpeta /home/bigdatateam/scripts/MDM\_Product

Es ahí donde se encuentran los Livy

1. MDM-livy: dispara el .jar que apunta hacia QAS
2. MDMDatalake-livy ejecuta el .jar para PRD
3. El MDMProductClean.py retira los archivos input del S3 y los pasa a una ruta BackUp dentro de su respectiva carpeta

**CASO: CARGA MANUAL DE ARCHIVO FULL**

En el caso de que se requiera hacer una carga de un archivo full, se debe considerar lo siguiente:

1. Limpiar las tablas en redshit para que estén vacias antes de la carga.
2. Limpiar la collection con un delete o un drop para que se vuelva a crear la collection en MONGO, la creación de esta se hace automaticamente al ejecutar el proceso
3. Cargar los archivos que pasa Gianfranco o Juan Carlos al S3 de PRD y verificar que no hayan archivos con el mismo nombre
4. Ir al Nifi 86
5. Entrar a la carpeta de /home/bigdatateam/scripts/MDM\_Product
6. Ejecutar: python3 "nombre\_de\_archivo\_livy" (si es para QAS: MDM-livy.py, si es para PRD MDMDatalake-livy.py) "QAS o NPRD" "fechadeldía" (ejemplo "20190521")

**MDM Dmatrizcampana**

Este flujo esta automatizado para que se cargue en el Nifi86 y en el Nifi .36

El nifi 86 apunta a redshift QAS y el Nifi .36 Redshift Semantix o Datalake

El proceso consta de una descarga de la Dmatrizcampana de la campana max de la tabla Fstaebecam y descarga desde esa campana hasta el futuro.

Luego esa descarga se inserta en S3, en el bucket de belc-bigdata-landing-dlk-qas o belc-bigdata-landing-dlk-prd en la carpeta Datalake/input/mdm/dmatrizcampana

Luego con el archivo "LoadFileToMongo.py" carga el archivo dmatriz que esta en el S3 al Mongo en el database: "infoProduct" al collection "Dmatrizcampana"

Este proceso se ejecuta cada 3 días a las 3:10 pm

**USUARIOS MONGO QAS Y PRD**

**QAS:**

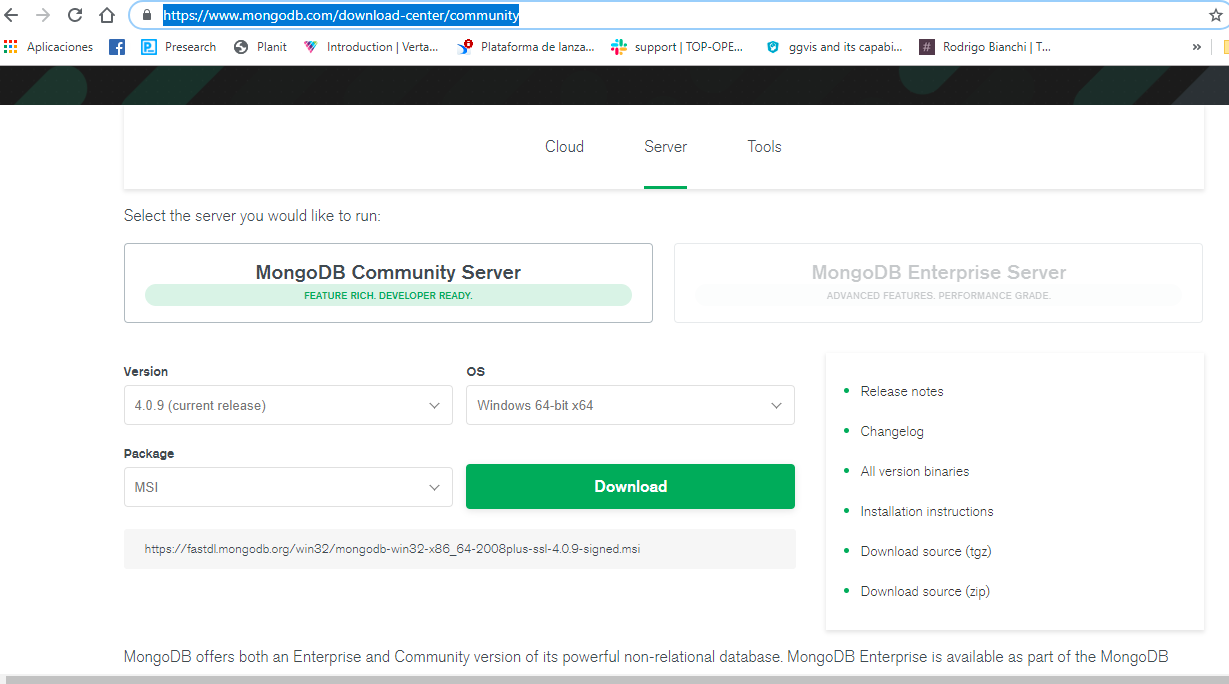
mongodb+srv://userBDInfoServiceQAS:cDi2UqJu6F64FalC@bigdatainfoservice-0hd3l.mongodb.net/admi

**PRD:**

mongodb+srv://userBDInfoServicePRD:7A0BMQ3YHS1gTSg0@bigdatainfoservice-8x89f.mongodb.net/admin

**DESCARGAR COMPASS**

<https://www.mongodb.com/download-center/community>



LUEGO DE HABERLO INSTALADO SOLO DEBEN COPIAR EL USUARIO QAS O PRD QUE ESTÁ EN LA SECCION DE ARRIBA Y AUTOMATICAMENTE EL COMPASS DETECTARÁ Y CONFIGURARÁ LOS PUERTOS Y ACCESOS