Automatización Datawarehouse en GCP - COVAP

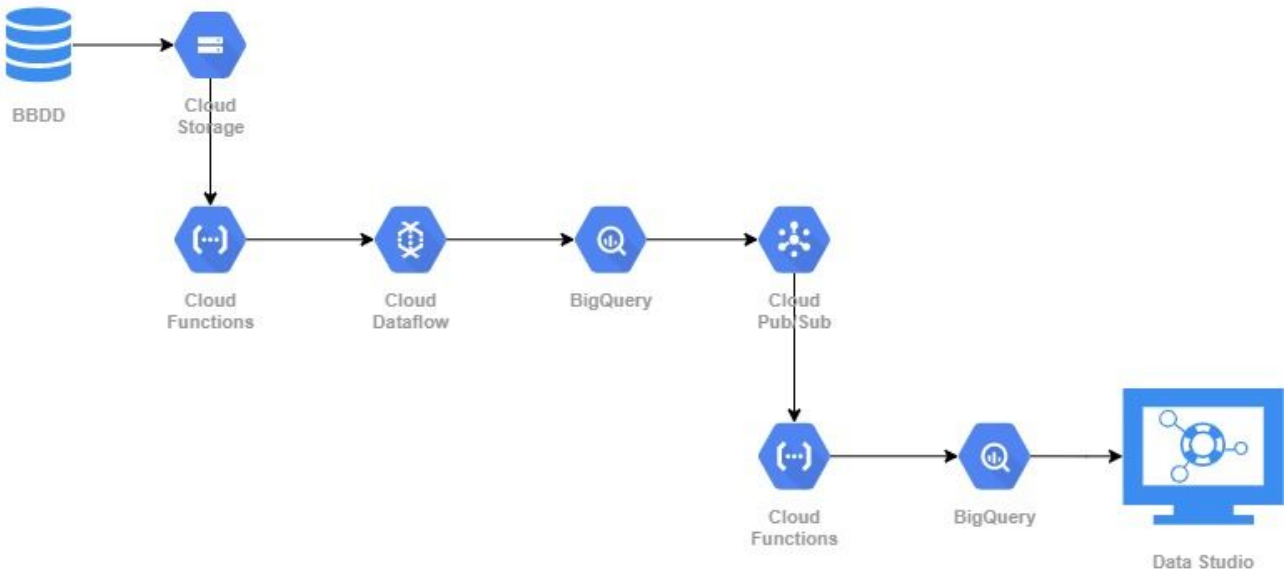
Marcos Remón Salazar

Índice

Estructura del proyecto

Diseño

El diseño de la estructura del proyecto para la automatización de subida de datos desde el servidor de Qlik a BigQuery y su posterior explotación con Data Studio es el siguiente:



Subida de archivos a BigQuery

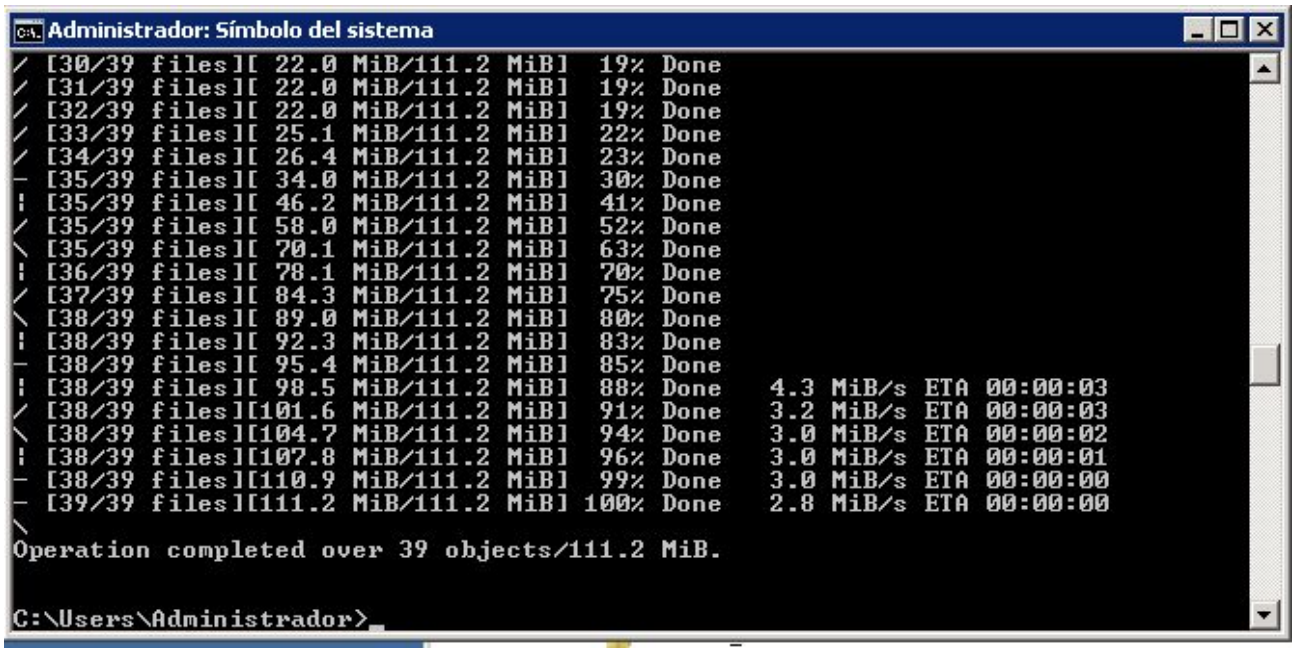
Cloud Storage

En primer lugar, hay que generar un archivo bat que contiene las instrucciones de subida de archivos del servidor On-Premise a Cloud Storage.

gsutil -m cp D:\QlikSense\Desarrollo\QVD\Procesados\Maestros\CSV*.csv gs://covap-gestion-financiera

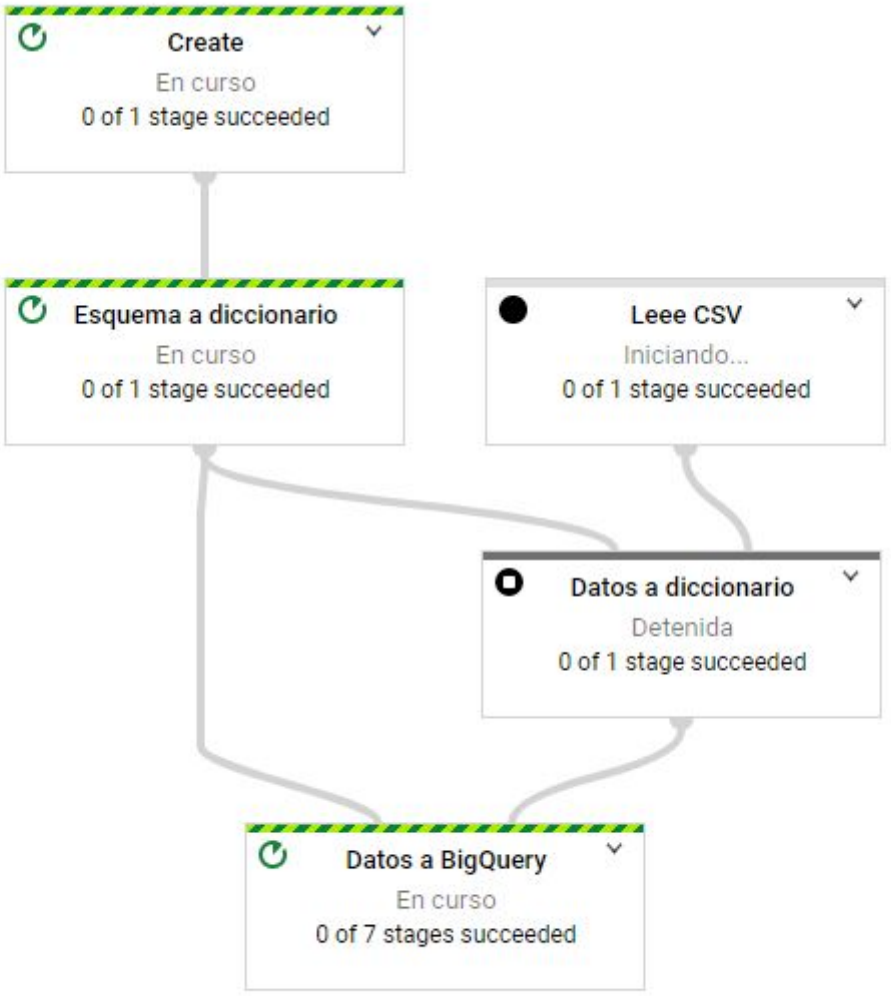
gsutil -m cp D:\QlikSense\Desarrollo\QVD\Procesados\Hechos\CSV *.csv gs://covap-gestion-financiera

Este archivo se encuentra en ***D:\GoogleCloud\subidaCloudStorage.bat*** y se ejecutará al terminar la recarga de la **app procesa_QVD_Ventas_Google** de Qlik Sense.



Cloud Functions + Dataflow

Una vez subidos los archivos a Cloud Storage saltará la Cloud Function ***csv-to-dataflow*** que ejecuta la plantilla de Dataflow para cada archivo, para ello es necesario que en el bucket **gs://covap-gestion-financiera/schemas** esté subido un archivo json con el esquema de la tabla correspondiente.



Una vez se haya finalizado correctamente la tarea en **Dataflow**, ya estarán los archivos csv correctamente subidos a **BigQuery.**

BigQuery

Estructura Datasets

La estructura del proyecto ***gestionfinanciera*** en BigQuery es la siguiente:

- STG_GestionFinanciera: En este Dataset se transfieren los datos de Cloud Storage a BigQuery y sin transformaciones.
- ETL_GestionFinanciera: Están las vistas que contienen la lógica ETL que se aplica a las tablas de STG.
- DWH_GestionFinanciera: Dataset destino para las tablas una vez se han realizado las transformaciones correspondientes.
- HECHOS_GestionFinanciera: Este Dataset contiene el modelo de datos que utilizaremos en Data Studio.

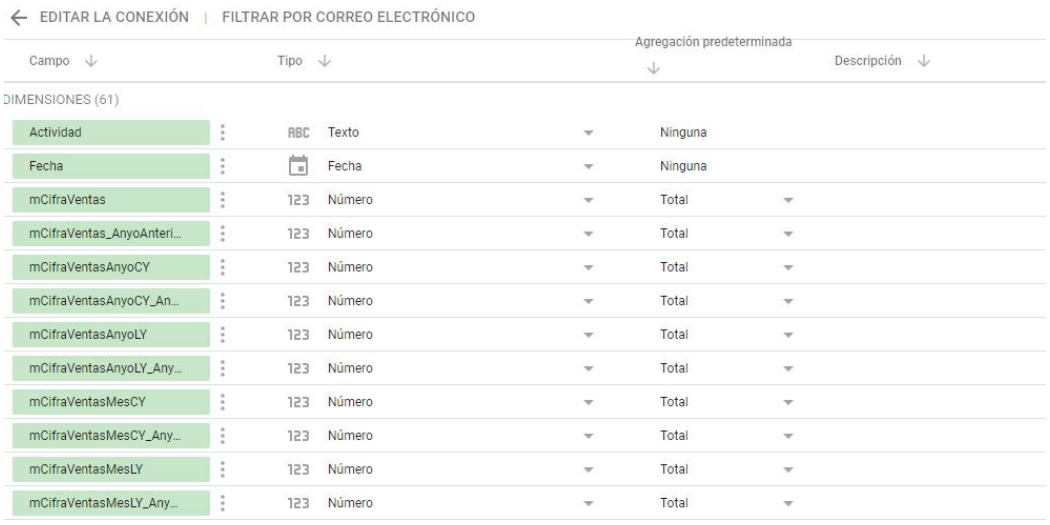
Actualización Datawarehouse

La actualización del Datawarehouse se hace a través de la Cloud Function ***bq-stg-to-dwh*** que ejecuta las vistas ETL y guarda los datos en DWH_GestionFinanciera. También genera el modelo de datos en HECHOS_GestionFinanciera.

Data Studio

Conexión de datos

La aplicación CM Ventas utiliza la tabla ***gestionfinanciera.HECHOS_GestionFinanciera.Hechos*** como modelo de datos.

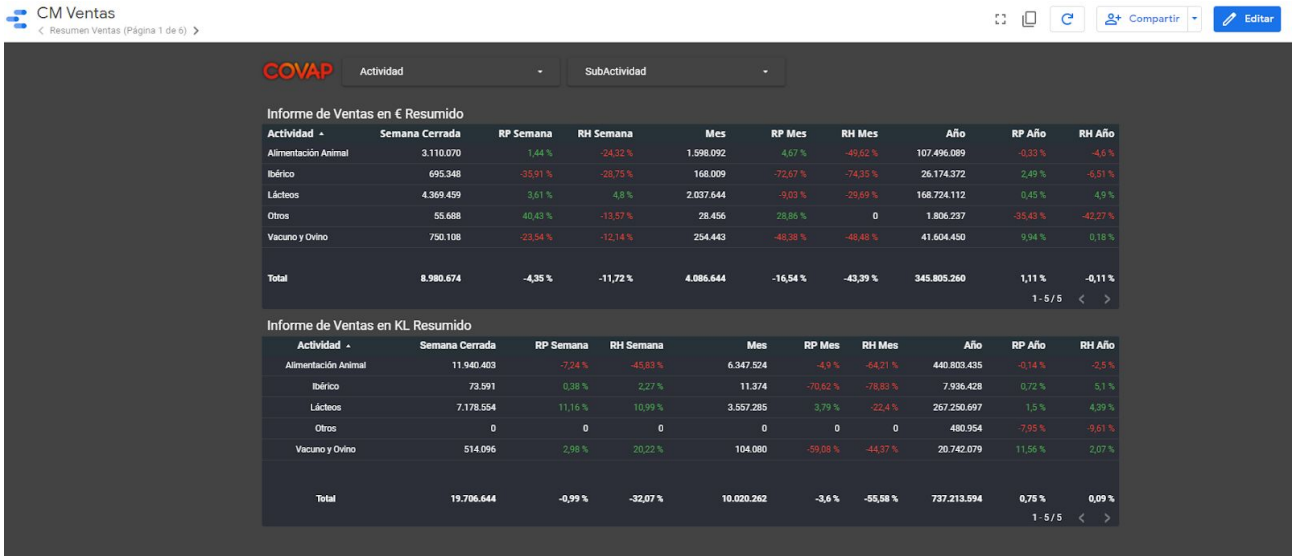


EDITAR LA CONEXIÓN | FILTRAR POR CORREO ELECTRÓNICO

Campo	Tipo	Agregación predeterminada	Descripción
DIMENSIONES (61)			
Actividad	Texto	Ninguna	
Fecha	Fecha	Ninguna	
mCifraVentas	Número	Total	
mCifraVentas_AnyoAnteri...	Número	Total	
mCifraVentasAnyoCY	Número	Total	
mCifraVentasAnyoCY_An...	Número	Total	
mCifraVentasAnyoLY	Número	Total	
mCifraVentasAnyoLY_Any...	Número	Total	
mCifraVentasMesCY	Número	Total	
mCifraVentasMesCY_Any...	Número	Total	
mCifraVentasMesLY	Número	Total	
mCifraVentasMesLY_Any...	Número	Total	

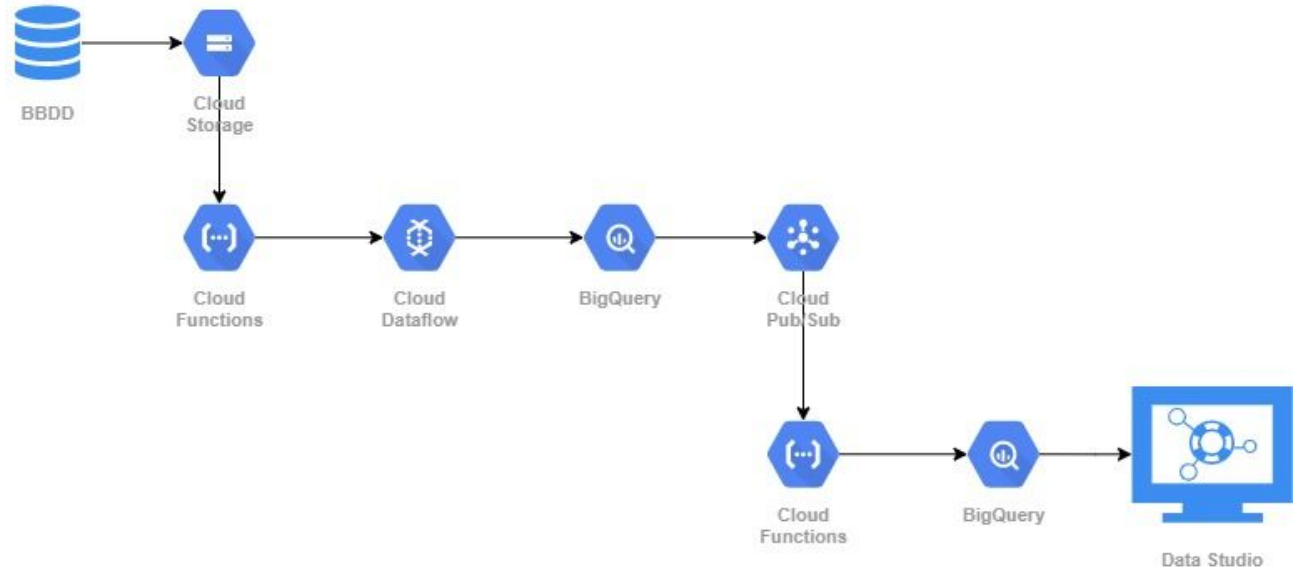
Actualización de los datos

Los datos se actualizan con la tabla ***gestionfinanciera.HECHOS_GestionFinanciera.Hechos*** automáticamente. Desde el cuadro también se pueden refrescar manualmente los datos.



En el caso del Informe Diario por Agentes la información se obtiene desde una consulta personalizada en la que filtramos los datos por Agente y partición de la tabla.

Flujo de datos



Cloud Storage

En primer lugar se ejecutará la tarea **procesa_QVD_Ventas_Google** que lanzará los .bat ***D:\GoogleCloud\subidaCloudStorage.bat*** y ***D:\GoogleCloud\subidaCloudStorageHechos.bat***.

Los csv subidos a **Cloud Storage** en el segmento **gs://covap-gestion-financiera** son los siguientes:

- ActividadCompras.csv

- ActividadVentas.csv

- ActividadVentas_V2.csv

- Agentes.csv

- AgentesCompras1.csv

- AgentesCompras2.csv

- AgentesResp.csv

- Articulos.csv

- ArticulosVentas.csv

- ArticulosVentas_V2.csv

- Cat01.csv

- Cat01_V2.csv

- Cat02.csv

- Cat02_V2.csv

- Cat03.csv

- Cat03_V2.csv

- Cat04.csv

- Cat04_V2.csv

- Cat05.csv

- Cat05_V2.csv

- Cat06.csv

- Cat06_V2.csv

- Cat07.csv

- Cat07_V2.csv

- Cat08.csv

- Cat08_V2.csv

- Clasificacion.csv

- Clasificacion_V2.csv

- Clientes.csv

- ClientesVentas.csv

- ClientesVentas_V2.csv

- Comisionista.csv

- Comisionistas.csv

- Companias.csv

- CompaniasVentas.csv

- CompaniasVentas_V2.csv

- Compañias.csv

- CompañiasVentas.csv

- DireccionClientesPlusVentas.csv

- DireccionClientesPlusVentas_V2.csv

- Direcciones.csv

- FamiliaCompras.csv

- Hechos.csv

- HechosTest.csv

- Hechos_V2.csv

- MAPVENTAS_CP_Coordenadas_V2.csv

- MAPVENTAS_DatosGEO_V2.csv

- MAPVENTAS_HH0_3W0_45GridTable_V2.csv

- MAPVENTAS_HH0_5W0_75GridTable_V2.csv

- MasterCalendar.csv

- MasterCalendar_V2.csv

- Mercados.csv

- MotivoCancelacion.csv

- Provincia.csv

- Rutas.csv

- Secciones.csv

- Sectores.csv

- Sectores2.csv

- SubActividadCompras.csv

- SubActividadVentas.csv

- SubActividadVentas_V2.csv

- SubFamiliaCompras.csv

- SubSecciones.csv

- SubSectores.csv

- UnidadesNegocio.csv

Dataflow

Una vez subidos los archivos csv a **Cloud Storage** se activa la función ***csv-to-dataflow*** que ejecuta la plantilla de Dataflow y convierte los csv en tablas en BigQuery.

Por ejemplo, **ActividadVentas.csv** -> **gestionfinanciera.STG_GestionFinanciera.ActividadVentas**

La función ***csv-to-dataflow*** está configurada para que cada vez que se suba un csv a **Cloud Storage** se active.

BigQuery

En **BigQuery** el proceso de los datos va desde **STG** (datos que provienen de los csv) a **DWH** (datos preparados para explotar desde **BigQuery** o con herramientas como **Data Studio**).

Para pasar los datos de **STG** a **DWH** se generan vistas en **ETL** que contienen la lógica de las transformaciones.

El flujo es el siguiente:

STG_GestionFinanciera.ActividadVentas -> ETL_GestionFinanciera.ActividadVentas -> DWH_GestionFinanciera.ActividadVentas

Para activar automáticamente este proceso se utiliza la función **bq-stg-to-dwh.**

Esta función ejecuta las vistas de **ETL** y guarda los resultados en **DWH**, para que se active la función se ha creado un Tema en **Pub/Sub** con el nombre **bq-Sink**, este tema recoge los logs de las tablas de **BigQuery** (En este caso la última modificación de las tablas de **STG**).

Los Logs se han filtrado desde **Logging** creando un Sumidero (en este caso **bq-jobcompleted-logs**) que recoge esa información.

```
protoPayload.methodName="jobservice.jobcompleted"
protoPayload.serviceData.jobCompletedEvent.job.jobConfigurati
load.destinationTable.datasetId="STG_GestionFinanciera"
```

Una vez los datos están en DWH se activa una última función **hechos_por_agente** que actualiza la tabla de Hechos.

Data Studio

Data Studio se actualiza automáticamente con los datos que hay en la tabla de Hechos en **BigQuery**.