Qlik Sense: automated QVD creation and re-loads with the use of SQL DB

Author: Yevgen Grynechko

Created: 2-Dec-2019

Index

Issue – managing the DEV and PROD environments

In my organization we are using two environment, DEV and PROD. In dev we go through the design, build and testing phases of the dashboard creation cycle. After getting the app approved by the customer we promote it to PROD. Unfortunately the process became complicated as with the app itself we need to promote or update the apps that create QVDs. I decided to automate the process of the QVD creation so in the future the only thing I need to worry about is to promote one app to prod and manage the QVDs from the SQL table.

SQL Table

Based on my requirement I decided to build the SQL table called 'QlikQVD_A' that looks like this:

TableName	QVDName	LoadStatement	TableFilter	RefreshFrequency	ActiveFl

1. TableName – contains the name of the SQL table I want to load data from, e.g.: dbo.Customer_D
2. QVDName – the name I want to use for the QVD name that I going to store, e.g.: Customer_D.qvd
3. LoadStatement – most of the times I am just going to use '*' to load all of the columns from the table but sometimes there is a need for only certain fields or additional field transformation.
4. TableFilter – usually just an empty field, used to apply filter on the qlik side of things, e.g.: Where ID = '10'
5. RefreshFrequency – I have a need of only 3 refresh frequencies: 15MINUTES, HOURLY and DAILY
6. ActiveFl – set to 'Y' or 'N'

QVD re-load apps

I use 3 apps to generate QVDs, it could be one with additional IF to support all reload frequencies but as the successful re-load of those triggers the re-load of dashboard I decided to go with one for evert frequency.

1. Set to re-load every 15 minutes
2. Set to re-load every hour
3. Set to re-load every day

QVD script

The only difference in QVD is the use of different values of the RefreshFrequency so I am just going to show one.

Configuration section:

```
LET path_QVD_Directory = lib:// Automated/;
LET path_QVD_Extract = $(path_QVD_Directory) \Extract\;


IF v_Environment = DEV THEN
	Let v_SQLEnvironment = Dev.dbo.;
ELSE
	Let v_SQLEnvironment = Prd.dbo.;
END IF;
```

Tables Load section:

```
// 1.Load the list of tables

Table_List:

LOAD

Mid(TableName,5) as Table_Name, //remove dbo. from the name

Left(QVDName, Len(QVDName)-4) as QVDName,//remove .qvd from name

Chr(160) & text(LoadStatement) as Load_Statement,

Chr(160) & TableFilter as Table_Filter; //creation of the Table Filter with Spacing before Where

SELECT

TableName,

TableFilter,

LoadStatement,

QVDName

FROM $(v_SQLEnvironmentEDS)"QlikQVD_A"

WHERE ActiveFl = Y and RefreshFrequency in (HOURLY); //load only the ones with active

flag and hourly re-load

// 2. Loop through Table_List and load each table

FOR i = 0 to (FieldValueCount(QVDName) -1)


   LET v_TableName = Peek(Table_Name, $(i), Table_List);

   LET v_QVDName = Peek(QVDName, $(i), Table_List);

   LET v_TableFilter = Peek(Table_Filter, $(i), Table_List);

   LET v_LoadStatement = Peek(Load_Sta tement, $(i), Table_List);
```

```
Trace SELECT: $(v_LoadStatement);
Trace TableFilter: $(v_TableFilter);
Trace TableName: $(v_TableName);
    [$(v_QVDName)]:
    LOAD $(v_LoadStatement)
    ;
    SELECT *
    FROM $(v_SQLEnvironment)$(v_TableName)$(v_TableFilter);


STORE [$(v_TableName)] INTO [$(path_QVD_Extract)\$(v_TableName).qvd] (qvd);
    SET v_TableName = ;
    SET v_QVDName = ;
    SET v_TableFilter = ;
    SET v_LoadStatement = ;

NEXT i
```

Summary

With the use of the SQL table I am able to automatically manage all of the needed QVDs both in DEV and PROD environments. Of course there is much more you can add to the script like the mechanism to store the meta data which I plan on adding in different document.