Centralizing Data Manipulation in SAS and Eliminating Hundreds of Excel Formulas and VBA Programs: a Streamlined Reporting Approach

Amos Shu, Sanofi-Aventis U.S., Bridgewater, NJ
Jim Fang, Sanofi-Aventis U.S., Bridgewater, NJ
Reddy Ayyalapu, Sanofi-Aventis U.S., Bridgewater, NJ
Patrick Dunleavy, Sanofi-Aventis U.S., Bridgewater, NJ

ABSTRACT
Sanofi-aventis currently receives syndicated data from multiple third party vendors. Information Services (IS) integrates data from these vendors with internal data and then generates periodic (weekly and Monthly) reports. Business users (Market Research, Brand Managers, Sales VP’s and Executives) prefer receiving these reports in a customized/pre-defined Excel and PowerPoint format so that they can spend their energy to focus on data interpretation rather than on reports generation. They also prefer receiving data used for generating these reports in Excel.

The team that originally built this application separated the whole reporting process into two parts: data manipulation in SAS and data calculation in Excel worksheets in order to meet the end-user's requirements. Once the data integration is completed in SAS, SAS datasets are transferred into hundreds of Excel worksheets which has hundreds of formulas/functions, defined names, and complex VBA code. This approach has many challenges/issues in data accuracy, reporting performance, flexibility, scalability, maintenance, and managing production runs. It is very difficult to debug and resolve issues.

This paper presents a scalable, flexible, stable, time/space saving, and streamlined approach to automate repetitive reporting activities, simplify process, and significantly improve reporting quality and performance. This reverse engineering process centralizes all data manipulation in SAS by utilizing structured SAS modules and SAS macros while eliminating all Excel functions/formulas and majority of VBA code. It also makes it easy to support this new process. Finally pros and cons are compared between old and new solutions.

INTRODUCTION
It is common to see that lot of large reporting systems at multinational companies usually comprise multiple programming languages such as SAS, SQL, and VBA to integrate multiple different data sources in varying formats for multiple products in different markets simultaneously. This approach has the following challenges.

• It is hard to control the data quality since calculations are performed in SAS and Excel.
• It requires highly skilled personnel who have rich experience in multiple programming languages in order to complete the whole reporting process because final deliverables are highly time sensitive.
• It is very difficult to work with hundreds of worksheets embedded with hundreds of formulas/functions and defined names in a single Excel workbook.
• It lacks flexibility for adding new products, or new markets, or changing market definitions.
• Breaking down a reporting process into multiple parts with multiple languages increases the possibility of failure/errors.

SOLUTION
To overcome all above shortcomings, we researched and presented the following approach using SAS:

Step 1: Use SAS Macros to integrate data from 3rd party vendors with internal data. This simplified the programs and quality control processes.

"select a type of data in a specific market */
%MACRO DataSource (dsource=Data_Source1, type=A);
   PROC SORT data=&dsource.;
      by productid ;
   RUN;
/* get subMarket and product group format */
DATA Market&type. (Where=(Market = "Market1");
   MERGE MarketFormat (in=in1) &dsource. (in=in2);
Step2: Centralize all markets, submarkets, and product categories/groups definition in SAS, which made it flexible and efficient to calculate data in different market levels.

```sas
%MACRO SelectRxType (type1=Rx, type2=NRx, sub= Submarket1, prod= ("Product1"), group= Tot);
  /* get one type data such as NRx or TRx only in DIA market */
  DATA Market&type2. (keep= productid  Market  Submarket1  Submarket2 ProductCategory1  ProductCategory2  TypePill &type2.);
    SET Dia&type1.;
  RUN;
  /* get one type data for just one product or product group */
  PROC SUMMARY data=Market&type2. (where=(&sub. in &prod.)) nway;
    var _numeric_;
    class & sub.;
    output out= Market&type2._&group. sum=;
  RUN;
%Mend SelectRxType;
```
Step3: Centralize all data manipulation and calculations in SAS and eliminate VBA code and functions in Excel, which significantly reduced the size of the final deliverable Excel file (i.e. from 10 MB to 1 MB).

<table>
<thead>
<tr>
<th></th>
<th>Current Report</th>
<th>New Report</th>
</tr>
</thead>
<tbody>
<tr>
<td># of Excel Worksheets</td>
<td>76</td>
<td>10 (PPT need only)</td>
</tr>
<tr>
<td># of Defined Names</td>
<td>544</td>
<td>80 (PPT need only)</td>
</tr>
<tr>
<td>File Size</td>
<td>12.4MB</td>
<td>1MB</td>
</tr>
<tr>
<td>Contents</td>
<td>Raw data and PPT data</td>
<td>PPT data only</td>
</tr>
<tr>
<td>Calculation in Excel</td>
<td>Hundreds of formulas</td>
<td>No</td>
</tr>
<tr>
<td>Maintenance</td>
<td>Time consuming May need hours</td>
<td>Minutes</td>
</tr>
<tr>
<td># of pages of VBA code length</td>
<td>60</td>
<td>10 ( PPT only)</td>
</tr>
<tr>
<td>Trouble Shooting</td>
<td>Time consuming</td>
<td>Quick</td>
</tr>
</tbody>
</table>

Step4: Break down SAS programs and data sets into several groups based on content similarity in the PowerPoint reports. As an example, SAS programs that were used to generate slides for ActBgtTmr were name ActBgtTmr.sas. Any errors on the final deliverables can be easily tracked back to the original data sets easily and fixed quickly. Structured SAS programs were used to replace complex Excel formulas and name ranges.

```
%MACRO SpendPercent (mkt=1, tomt=Market$$, mt= );
   DATA sov_sub_sum&mkt.&mt.2 (drop= _TYPE_ _FREQ_);
      SET sov_sub_sum&mkt.&mt. ;
      if submkt&mkt. = "NA" then delete;
   RUN;
   
   DATA sov_mkt&mkt.&mt. ;
      SET sov_sub_sum&mkt.&mt.2; SubMkt&mkt. = "&totmt" ;
   RUN;
   
   PROC SUMMARY data = sov_mkt&mkt. nway; var _numeric_; class SubMkt&mkt . ; output out=sov_mkt_sum&mkt.&mt. (drop= _TYPE_ _FREQ_ ) sum=; RUN;
   
   DATA sov_percent&mkt.&mt.(keep=submkt&mkt. chansh); SET sov_mkt_sum&mkt.&mt. sov_sub_sum&mkt.&mt.2 ; format chansh01-chansh24 percent7.1 ;
   if submkt&mkt. = "&totmt" then do;
   do i=1 to 24; mkttot(i)=dat(i); end;
   end;
   retain mkttot01-mkttot24;
```
do i=1 to 24;
    if mkttot(i)=0 then chansh(i) = 0;
    else chansh(i) = dat(i)/mkttot(i);
end;
if submkt&mkt."&totmt" then delete;
RUN;
PROC SORT data= sov_percent&mkt.&mt. ;
   by descending SubMkt&mkt . ;
RUN;
%Mend SpendPercent;

%MACRO LaunchLvLnBye (in=DiaNRx_Lev, type=NRx, out=NRxLaunchLvLn);
PROC TRANSPOSE DATA= &in. (rename=(ProductCategory2=name)) OUT=xxx1;
   BY name;
   VAR &type.01-&type.67;
RUN;
DATA xxx2;
   set xxx1;
   where col1 ^= 0;
RUN;
PROC SQL;
   create table xxx as
   select *
   from xxx2;
quit;
%let xcnt = &sqlobs;
PROC TRANSPOSE DATA= xxx2 OUT=xxx3 (keep=name &type.:);
   BY name;
   VAR col1;
RUN;
DATA xxx4 (keep=name Rx:);
   set xxx3;
   array &type.(&xcnt.) &type.01-&type.&xcnt.;
   array Rx(&xcnt.) Rx01-Rx&xcnt.;
   h=0;
   do i=1 to dim(&type.);
      m= dim(&type.) - h;
      h=h +1;
      Rx(m)= &type.(i);
   end;
RUN;
DATA &out. (keep=name &type.:);
   set xxx4;
   array Rx(24) Rx01-Rx24;
   array &type.(24) &type.01-&type.24;
   h=0;
   do i=1 to 24;
      m=24-h;
      h=h +1;
      &type.(m)=Rx(i);
      if i>&xcnt. then &type.(m)=0;
   end;
RUN;
OUTCOME
By applying this approach, the time to complete all SAS processes is less than 2 minutes versus 7 minutes. There are
no calculations in Excel. The file size is reduced from 12MB to 1MB. The number of Excel worksheets in the final
report is cut from 72 to 10. The numbers of defined names are reduced from 544 to 80. The numbers of pages of
VBA code is reduced from 60 to 10. Errors can be tracked back from PowerPoint slides to original data easily thereby
making it easy to resolve the issues within short duration.

ACKNOWLEDGMENTS
The authors would like to thank Sanofi-Aventis IS Reporting and Analytics group for their comments and support.

CONTACT INFORMATION
Your comments and questions are valued and encouraged. Contact the author at:

Name: Amos Shu
Enterprise: sanofi-aventis
Address: 55 Corporate Drive
City, State ZIP: Bridgewater, NJ 08807
Work Phone: (908)981-6088
E-mail: Amos.Shu@sanofi-aventis.com

Name: Jim Fang
Enterprise: sanofi-aventis
Address: 55 Corporate Drive
City, State ZIP: Bridgewater, NJ 08807
Work Phone: (908)981-6080
E-mail: Jim.Fang@sanofi-aventis.com

Name: Reddy Ayyalapu
Enterprise: sanofi-aventis
Address: 55 Corporate Drive
City, State ZIP: Bridgewater, NJ 08807
Work Phone: (908)981-6071
E-mail: Reddy.Ayyalapu@sanofi-aventis.com

Name: Patrick Dunleavy
Enterprise: sanofi-aventis
Address: 55 Corporate Drive
City, State ZIP: Bridgewater, NJ 08807
Work Phone: (908)981-6092
E-mail: Patrick.Dunleavy@sanofi-aventis.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS
Institute Inc. in the USA and other countries. ® indicates USA registration.
Other brand and product names are trademarks of their respective companies.