

Paper 074-2010

Generating Reports with DDE and VBA, the Concept- Alternative to ODS to Produce Complex Reports

Chienchung Chen, HAVI Global Solutions, Downers Grove, IL

ABSTRACT

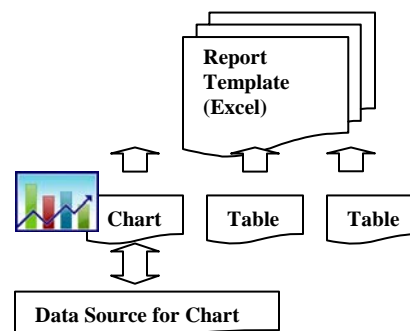
As a SAS programmer, you might have experience of spending hours modifying the thickness/colors of a bar chart, fitting an image into the corner of the page, or putting multiple charts/tables side by side with ODS PDF. Using ODS PDF, you can generate reports directly from the SAS dataset/database and mass-produce reports with the assistance of SAS Macro. Besides ODS LAYOUT, a useful statement, can be used along with SAS/GRAPH and ODS PDF to generate reports with basic number of reporting components, for example, 1-2 table, 1-2 charts, or combination of 1-2 tables with 1-2 graphics effectively. However, it is a time consuming process to modify all the properties (alignments, positions, sizes, colors, etc) on the report while the report is getting complicated with the increasing number of tables/graphics/pictures. Especially, when the report contains varied types of graphics (bar charts/trend line/etc.), tables, and titles/footnotes, it will take the SAS programmer huge amount of time and efforts to develop the codes. The subsequent modifications on the graphics/tables also take time to complete. Further, the changes on previous ODS PDF, if not justified correctly, might cause SAS to drop some of the objects or place the objects in the wrong locations. As a result, the programmer and client usually have no choice but sacrifice the visibility and readability of the report in order to have all the required components on the report. The use of SAS DDE and VBA might be a quick solution to it.

INTRODUCTION

ODS PDF works perfectly when you generate simple reports with limited number of components on each page. With ODS LAYOUT, you can even fit each component into assigned area on the report through varied methods (absolute/gridded). Besides, PROC GREPLAY procedure can help to produce good quality graphics. You can use ODS PDF TEXT to mix text and graphics to edit the report. Nevertheless, in the real world, those procedures are not easy to master when the layout of the report is complex. Besides, people who design the reports might not have knowledge of how SAS is generating reports. As a result, the final template might not be compatible with some of SAS procedures or take long time to develop. Even though some types of the web tools might be capable of resolving those issues, they might be costly to acquire and maintain. For most of the clients I worked with, they only have Microsoft Office and Adobe Acrobat/Reader and often prefer to receive Excel or PDF files. The usage of DDE with VBA certainly serves as a time-saving alternative to ODS. It can also retain the quality of the report especially for complicated reports with varied graphics. Furthermore, you can still take the advantages of SAS Macro to mass-produce the reports and use VBA Macro to customize your reports, for example, to assign logo according to the group or up-arrow when the measure is increasing/down-arrow when the measure is decreasing to improve the visual looks of the report. The VBA can also protect/hide your workbooks or convert the Excel file into PDF file depending on the client's needs. The paper presented provides the SAS users with a basic concept of how to integrate DDE and VBA to generate reports as alternative to other SAS procedures under the PC environment.

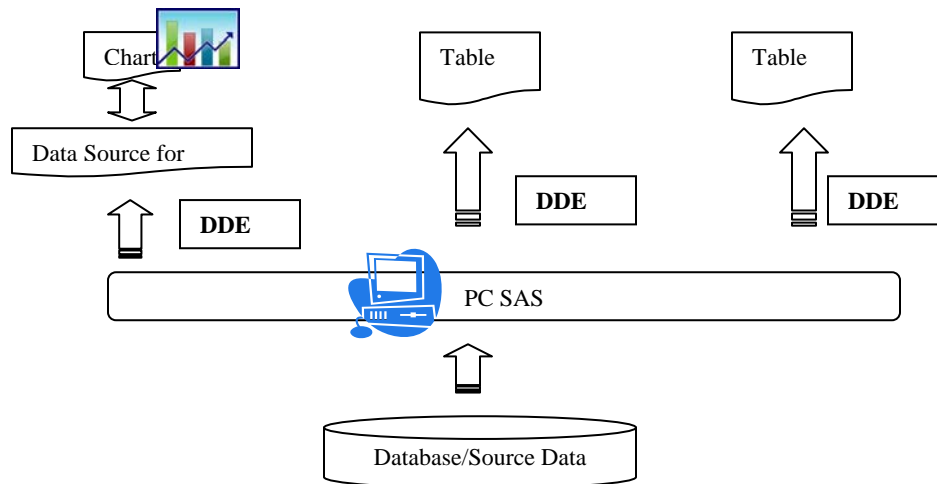
DESIGNING THE REPORT

In our example, the template will be designed on Excel and will serve as foundation to generate reports. During the process of generating reports, the template will remain unchanged unless there are needs to modify the reports. Within the template, each of the charts/graphics should be treated as individual element and should have its own data source on the corresponding worksheets. For tables and other content on the report template,



data can be outputted through DDE directly to the report template or to the linked worksheets. By the end of the process, unwanted worksheets can be hidden or deleted before the reports are generated or converted to other formats like PDF through later steps.

EXPORTING DATA INTO EXCEL TEMPLATE THROUGH DDE



DDE (Dynamic Data Exchange) is a method for exchanging information between Windows applications and PC SAS. During the time of exchange, both applications need to be simultaneously running in Windows. In our sample, SAS data can be exported to template for tables and data source for charts. Below is the basic step by step introduction of how DDE is functioning.

1. Opening application and build the bridge between Windows application and SAS through DDE

SAS code is submitted to call the system to open Windows application, for example, EXCEL in our sample.

```

OPTIONS NOXWAIT NOXSYNC;
x 'cd C:\Program Files\Microsoft Office\Office10\';
x 'excel.exe';

DATA _null_;
rc=SLEEP(1);
RUN;

FILENAME excelclds DDE 'excel|system';
DATA _null_;
FILE excelclds;
PUT '[app.minimize()]';
RUN;

```

2. Opening the Excel template

After the application is initiated, SAS will ask the system to open the EXCEL template where DDE will export data into.

```
FILENAME excelclds DDE 'excel|system';

DATA _null_;
FILE excelclds;
PUT %unquote(%str(%'[open("C:\template.xls")]%'));
RUN;
```

3. Exporting data to desired area on the template (worksheet1), for example ROW #1 – ROW #10 and Column #2 – Column #6 on ‘summary’ workbook

Below DDE procedure will export SAS data to assigned area on the template. Each workbook should be filled with required data to update the tables and charts. You can also submit VBA macro through DDE to edit the report or convert it into different format, like PDF. Those procedures will be introduced next.

```
FILENAME dumpit1 DDE "excel|worksheet1!r1c2:r10c6";
DATA _null_;
DATA Data_To_Export;
FILE dumpit1;
PUT var1- var5;
RUN;
```

CALLING THE VBA MACROS THROUGH DDE TO MODIFY THE REPORTS

In order to manipulate VBA codes to customize the reports after data have be exported to EXCEL workbooks, you can use below statements to call VBA macros through DDE. Those VBA macros can be pre-recorded and used to edit the report as you desire, for example, to insert/rotate pictures, to do basic calculations, etc. Complicated customized functions could also be performed by VBA, for example, color-coding, generating graphics/pivot-table, summarize data, etc.

```
DATA _null_;
FILE excelclds;
PUT '[RUN("MACRO1")]';
PUT '[RUN ("MACRO2")]';
-----More Lines -----
RUN;
```

CONVERTING REPORT TO PDF FORMAT

After the report has been generated and edited to the layout you desire, you might still have to convert it into the format which the clients prefer. In our example, we will use PDF to introduce the process.

In order to convert Excel file into PDF file, first you have to find the information about the version of your existing Adobe PDF application. One of the easiest ways to find the version of your PDF is to print your existing Excel report with Adobe PDF while activating the "Record New Macro" function on the computer where you plan to generate the reports. During the process, the Excel report will be converted into Postscript file first and then be converted into PDF file.

```
Dim PSFileName As String
Dim PDFFileName As String
-----
PSFileName = "C:\ filename.ps"
PDFFileName = "C:\ filename.pdf"

----- More Lines -----
Application.ActivePrinter = "Adobe PDF on Ne00:"
ActiveWorkbook.PrintOut , prtfilename:=PSFileName

Dim myPDF As PdfDistiller
Set myPDF = New PdfDistiller
myPDF.FileToPDF PSFileName, PDFFileName, ""
```

UTILIZING SAS MACRO TO MASS PRODUCE REPORTS

You can also incorporate the strength of SAS Macro to mass produce reports. Thus, you don't have to repeat the same process for each of the reports with the same layout.

```
%MACRO REPORT ();

%DO i = 1 %to &Num_Report;

BODY of the SAS codes

%END;

%MEND;
```

CONCLUSION

The technique of using DDE and VBA can serve as an alternative to ODS PDF. However, it is only one of the solutions for SAS programmers who are not familiar with ODS procedures. It can save SAS programmers lots of coding time since it takes time to master on ODS statements and SAS/GRAPH to produce perfect graphics with SAS. Other choices like ODS MARKUP statements (ODS TAGSETS.EXCELXP as one of the most frequently used) can be used for the same tasks. Also, you can consider using PROC REPORT along with ODS RFT/PDF to generate standard reports. However, for those who only have basic knowledge of VBA but not master on ODS and SAS/GRAPH, learning SAS DDE might be one of the quickest ways to generate complex reports with SAS. Besides, it also enables the report designer and SAS programmer to have flexibility to create the report templates and retains the quality and visual looks of the reports, especially for those report templates created with Microsoft Excel.

RECOMMENDED READING

Choon-Chern Lim “*Step-by-Step in Using SAS® DDE to Create an Excel Graph Based on N Observations from a SAS Data Set.*” Available at:

<http://www2.sas.com/proceedings/sugi31/154-31.pdf>

Christopher A. Roper “*Using SAS and DDE to Execute VBA Macros in Microsoft Excel.*” Available at:

<http://www.nesug.org/proceedings/nesug00/cc/cc4016.pdf>

Jennifer S. Harper “*Mission Possible: Putting a Table and Multiple Graphs on a Single-Page PDF with ODS and Basic GOPTIONS.*” Available at: support.sas.com/resources/papers/proceedings09/085-2009.pdf

Mengxi Li, Sandra Archer, Russell Denslow “*Using DDE and SAS/Macro for Automated Excel Report Consolidation and Generation.*” Available at:

<http://analytics.ncsu.edu/sesug/2003/DM08-Li.pdf>

CONTACT INFORMATION

Your comments and questions are values and encouraged. Contact the author at:

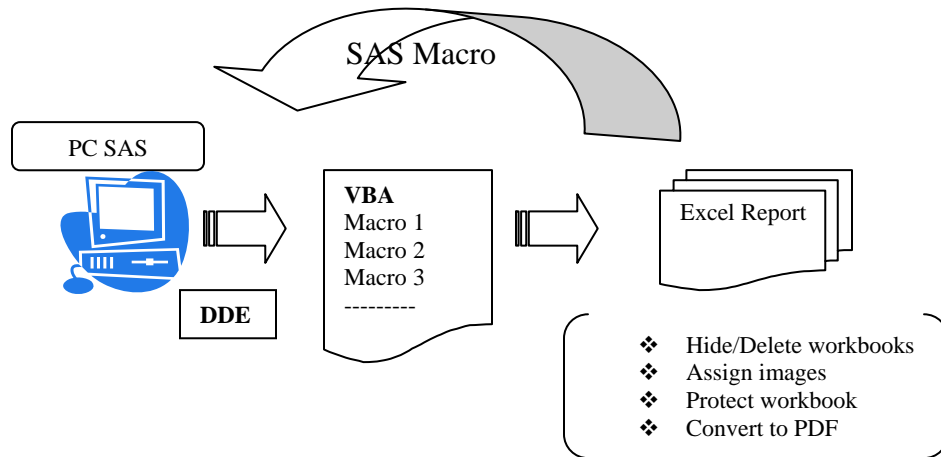
James, Chienchung Chen, Certified Advanced Programmer for SAS®9, MBA
HAVI Global Solutions
3075 Highland Parkway
Suite # 200
Downers Grove, IL 60516
Phone 630-493-6654
Email jchen@havigs.com
Web www.havigs.com



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.

APPENDIX

Overview of the Process

Sample of Complex Report Template



SAMPLE



SAMPLE REPORT for SAS Coders' Corner

Data Reported: Jan 2009 - Oct 2009
Data Reported: Jan 2008 - Oct 2008

Manager1: XXXXXXXXXXXX
Manager2: XXXXXXXXXXXX

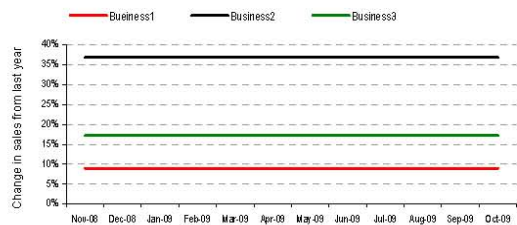
	Measure 1	Measure 2	Measure 3	Measure 4	Measure 5
Group1	10 / +12.1%	2.1 / +121%	44.4% / -2.1%	1 / +66.1%	50% / +1.1%
Total	100 / -2.3% <small>Measure / Δ YTD vs. PY</small>	80.1 / -0.6% <small>Measure / Δ YTD vs. PY</small>	60.0% / +1.5% <small>Measure / Δ YTD vs. PY</small>	30 / +12.1% <small>Measure / Δ YTD vs. PY</small>	50% / +1.1% <small>Measure / Δ from Q4 08</small>

*Note 1:

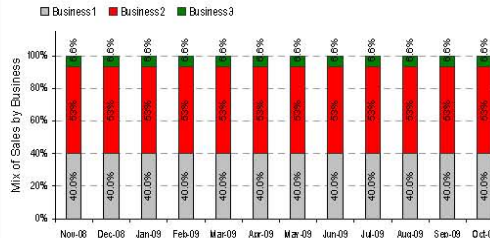
**Note 2:

Key Metrics and Trends

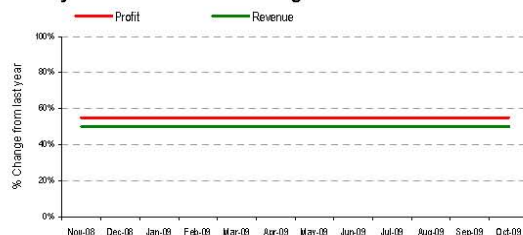
% Revenue Growth by Business Type



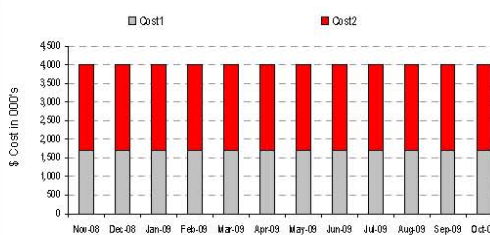
Business Mix



Monthly Profit and Revenue Change



Cost Structure



Business Type Detail

Top 10 Revenue Accounts

Bottom 10 Revenue Accounts

Account	Business			
#1				
#2				
#3				
#4				
#5				
#6				
#7				
#8				
#9				
#10				

Account	Business			
#1				
#2				
#3				
#4				
#5				
#6				
#7				
#8				
#9				
#10				

Top 10 Profitability Accounts by Business Type

Business 1

Business 2

Account	Description			
#1				
#2				
#3				
#4				
#5				
#6				
#7				
#8				
#9				
#10				

Account	Description			
#1				
#2				
#3				
#4				
#5				
#6				
#7				
#8				
#9				
#10				



HIGHLY CONFIDENTIAL

