Paper 161-30

Generating Custom Report Tables: Using SAS with DDE and VBA

Ying Feng, Educational Testing Service, Princeton, NJ

ABSTRACT

As a testing company, we are required to present statistical information to our clients in custom tables. SAS® is powerful for delivering statistical information, and Microsoft® Office is excellent for formatting. This paper details how to link the two systems together and how to automate this process using DDE and VBA so that non-programmers can easily generate the custom tables. The process involves the following steps.

First, statistical information is generated in SAS and exported to interim tables using PROC EXPORT. Then, an Excel template is created holding specially formatted tables to meet clients' requirements. Then, DDE capability in SAS is used to open the Excel template and execute VBA code, which is embedded in the template. The VBA code maps information from the interim tables to the template. Thus, professional-looking report tables are generated utilizing the strong data manipulation capability of SAS and the excellent formatting capability of Microsoft Excel.

In the following paragraphs, the concepts of DDE and VBA are introduced first, then the process from template design to data population is described.

INTRODUCTION

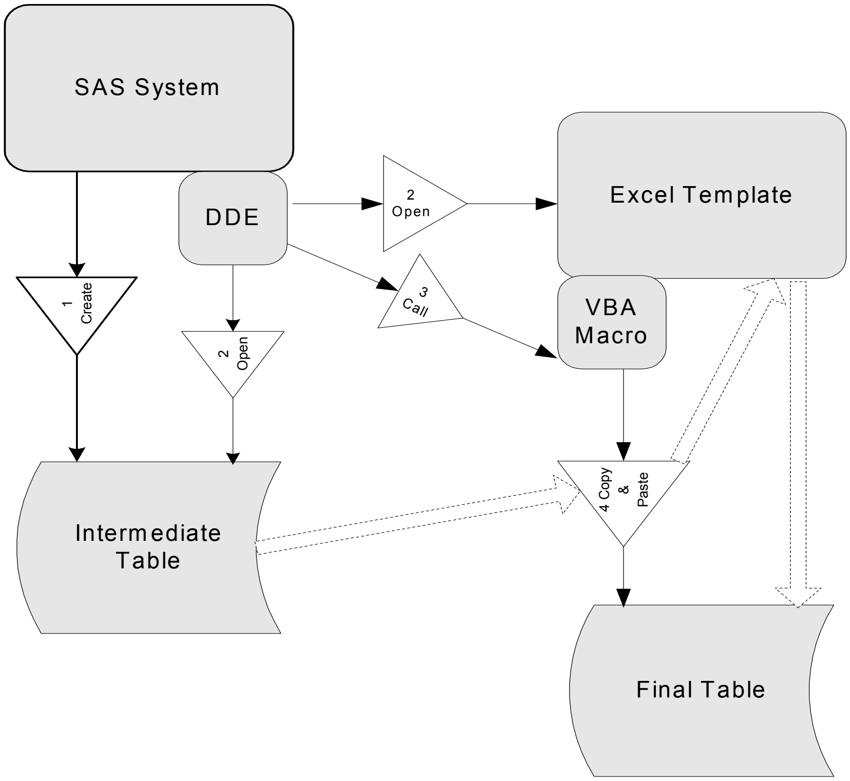
Dynamic Data Exchange (DDE) is a Microsoft Windows 3 hotlink protocol. It is a method of dynamically transferring data between Windows-based applications using a client/server model. In Version 8, the SAS System is always the client. In this role, the SAS System can request data or send data and commands to server applications. To use DDE, both applications must be running and the data that is transferred must be active in both applications.

Visual Basic for Applications (VBA) is Microsoft's common language for manipulating components of its Microsoft Office suite. It is used as a macro language for Microsoft applications to customize and extend their functions. VBA operates on objects representing the application and the entities it manipulates, e.g. a spreadsheet or a range of cells in Microsoft Excel.

Six times per year, we need to send summarized statistical information to our clients. The information is presented in 30 visually pleasing and easy-to-manipulate Excel tables. Making these specially formatted tables in SAS is quite complicated even though SAS is powerful in processing statistical information. It also doesn't make sense to copy and paste SAS output to the tables manually. Since producing these tables is a recurring event and a large amount of information is generated in each event, copy/paste can quickly become a time-consuming and error-prone process. DDE and VBA can come to the rescue in this case. VBA macros can be generated to automate copy/paste process. After SAS outputs the intermediate tables, DDE is utilized to call these VBA macros from

SAS. Thus, the whole process is automated, and producing the tables is a breeze even for non-programmers.

Below is the flow chart for the process.



In the following, an example of producing a correlation table demonstrates this approach step by step.

GENERATE AND OUTPUT INTERMEDIATE TABLES FROM SAS

The first step uses the CORR procedure in SAS to generate the correlation information and the EXPORT procedure to output the results to an intermediate Excel table.

```
proc corr data=ela nosimple noprob noprint outp=tmp1;
      var comp_wela avg_CR_SCR SBGRP_RI_CNT_WA
            SBGRP_RI_CNT_RC SBGRP_RI_CNT_LA SBGRP_RI_CNT_WS
            SBGRP_RI_CNT_WC;
run;

proc append base=out data=tmp1; run;

proc corr data=math nosimple noprob noprint outp=tmp1;
      var SBGRP_RI_CNT_TOT_M SBGRP_RI_CNT_PS SBGRP_RI_CNT_NUM
```

```
                SBGRP_RI_CNT_AF SBGRP_RI_CNT_MG SBGRP_RI_CNT_AG;
run;

proc append base=out data=tmp1; run;

proc export data=out outfile='c:\StatSum\interim\corr.xls'
replace; run;
```

See Appendix A for an example of the intermediate table produced.

The code above is stored in a separate module. To produce all 30 tables, there are several modules, each producing statistics for various tables. Modularization has many benefits, including the creation of reusable and user-friendly code. This paper will only focus on the correlation module. The above PROC CORR code is saved into a SAS program called Statsum_corr.sas, and in a separate program, we can put the following code:

```
filename inn 'c:\StatSum\program';
%include inn(StatSum_corr.sas);
```

Users will only need to submit these two lines to run the PROC CORR program. This program will be referred to as the "Macro Call" program.

GENERATE VBA MACROS IN EXCEL

Next, we generate an Excel template for the correlation information. An example of this final template can be found in appendix B.

Next, a VBA macro is written to automate the process of copying the correlation information from the intermediate table to this template.

For new VBA users, a relatively easy way to create VBA code is by using the Macro Recorder feature provided in Excel. Of course, the Macro Recorder is quite limited and you will need to learn VBA syntax to perform more complicated data manipulation, but it will suffice for our purposes here.

The following steps demonstrate how to create a "**Corr**" VBA macro using the Macro Recorder:

1. Open a blank Excel workbook, create the template as above, and save it as statsum.xlt. (Save as a template, not a normal spreadsheet)
2. Open the intermediate statistics stored in "corr.xls" and the template you just created, "statsum.xlt".
3. Inside statsum.xlt, select **Tools**, **Macro**, **Record New Macro**
4. Type in the **Macro name** as **Corr** and fill in the **Description**, Select **OK**. The Stop Recording toolbar will appear and let you know that Excel is recording your actions. Excel will continue recording them until you click the Stop Recording button on this toolbar.

5. Switch to Corr.xls, select a range and copy-paste it to the appropriate place in template.xlt
6. Repeat step 5 until you finish the copying-pasting all the statistics you need.
7. In Template.xlt, click on cell A1.
8. Now Click the **Stop Recording** button in the Stop Recording toolbar
8. Choose **File - Save**, and save it again as statsum.xlt.
9. Close Excel

Now there is a VBA macro called "**Corr**" saved in the Excel template. Below is a portion of the macro that shows one copy/paste procedure.

```
Windows("corr.xls").Activate
Range("C15:I21").Select
Selection.Copy
Windows("statsum.xlt").Activate
Range("B18").Select
Selection.PasteSpecial Paste:=xlValues, Operation:=xlNone,
SkipBlanks:= False, Transpose:=False
```

USING DDE TO CALL EXCEL VBA MACROS FROM SAS

Now the intermediate statistics table and the template with VBA "Corr" macro have been created. Here comes the fun part – linking SAS and Excel together. We will use DDE in SAS to open the Excel files and call the VBA macro that performs the copy/paste process.

Now we will add more code to the macro call program. First, use the X Command to open a DDE Server, in this case, Excel. You need to specify the complete path where excel.exe is stored on your machine since it's stored in different paths on different operating systems. Please note that the XWAIT and XSYNC options should be turned off.

```
options noxwait noxsync;
x '"C:\Program Files\Microsoft Office\Office\excel.exe" ';
```

To allow Excel time to open, we can use the sleep function, which pauses the SAS program the number of seconds specified.

```
data _null_;
   x=sleep(5);
```

To connect the SAS and Excel systems via DDE, we use a filename statement. Turn the XSYNC option on to specify that the operating system commands execute synchronously with your SAS session:

```
filename comma1 dde 'excel|system';
```

```
options xsync;
```

After the Excel system is opened, we send commands through the DDE link to open the intermediate excel table, corr.xls, and the template file, statsum.xlt:

```
data _null_;
  file comma1;
  put '[open("c:\StatSum\interim\corr.xls")]';
  put '[open("c:\StatSum\program\StatSum.xlt")]';
run;
```

Finally, we send a run command to run the Excel VBA macro "**Corr**"; save the formatted file; and quit and close the Excel system.

```
data _null_;
 file comma1 ;
 put '[run("StatSum.xlt!corr")]';
 put '[SAVE.AS("c:\StatSum\output\regular.xls")]';
 put '[quit()]';
run;
```

WHAT'S LEFT TO DO – WRITE MORE USER-FRIENDLY CODE

The whole process has been demonstrated above. For a non-programmer to run this program, they will only need to change the input and output locations when they run the program. It sounds simple? Wait… Remember we have 30 intermediate tables? That means we have more than 30 input and output names to change. To make this process more user-friendly, we can utilize a SAS macro. For example, inside the SAS module StatSum_corr.sas, we can change the code so that the output file name is replaced with a macro variable name.

FROM:

```
proc export data=out outfile='c:\StatSum\interim\corr.xls'
replace; run;
```

TO:

```
proc export data=out outfile="&out."
replace; run;
```

Then in the "Macro Call" program, use a %let statement to set the macro variable.

```
%let out= c:\StatSum\interim\corr.xls;
```

Thus, users don't need to go to each module to change the file names. Everything can be controlled in the macro call program.

CONCLUSION

With the help of DDE and VBA, professional-looking tables can be easily created. The process described in this paper shows how to link SAS and Excel together and how to automate the whole process. Of course, there are a lot of other approaches available to achieve the same goal, such as SAS Output Delivery System (ODS), or a combination of Visual Basic (VB) and SAS. Depending on the clients' preference, the output can also be put into different formats, such as MS Word, PDF, HTML. These approaches have their pros and cons, but they are beyond the scope of this paper and will not be discussed here.

REFERENCES

Roper, C. A. "Intelligently Launching Microsoft Excel from SAS, using SCL functions ported to Base SAS". *Proceedings of the Twenty-Fifth Annual SAS Users Group international Conference*, paper 97, 2000.

SAS Institute, "Technical Support Document #325 – The SAS System and DDE". http://support.sas.com/techsup/technote/ts325.pdf, updated 1999.

Vyverman, K. "Using Dynamic Data Exchange to Export Your SAS Data to MS Excel — Against All ODS, Part I". *Proceedings of the Twenty-Seventh Annual SAS Users Group International Conference*, paper 5, 2002.

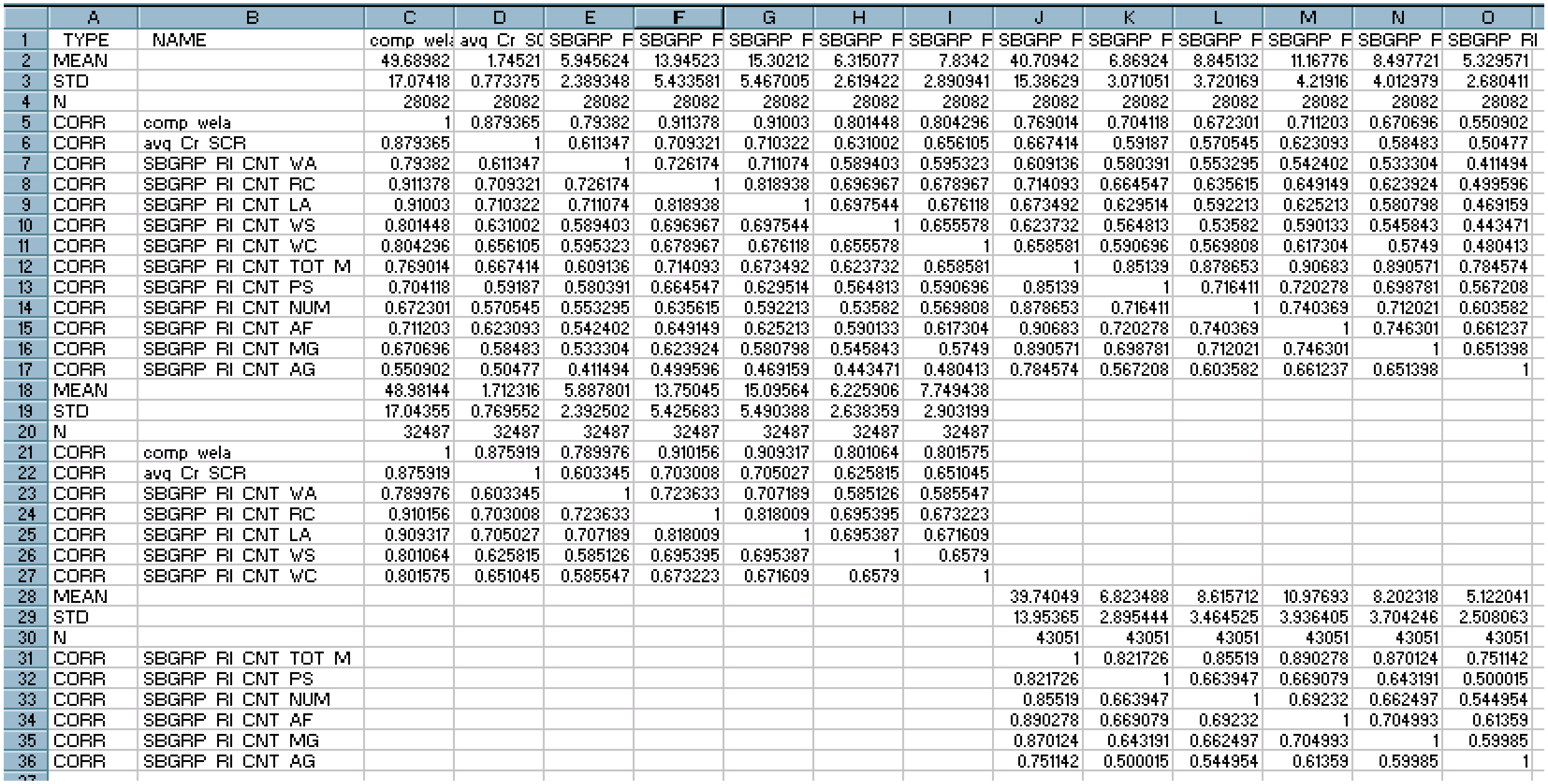
AUTHOR CONTACT INFORMATION

Ying Feng
Senior Statistical Systems Analyst

Educational Testing Service
11-L Rosedale Road
Princeton, NJ 08541

Telephone: 609-683-2360
FAX: 609-683-2400
Email: yfeng@ets.org

Appendix A: The Intermediate Table for the Correlation Information



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	TYPE	NAME	comp wela	avq Cr SC	SBGRP F	SBGRP F	SBGRP F	SBGRP F	SBGRP F	SBGRP F	SBGRP F	SBGRP F	SBGRP F	SBGRP F	SBGRP RI
2	MEAN		49.68982	1.74521	5.945624	13.94523	15.30212	6.315077	7.8342	40.70942	6.86924	8.845132	11.16776	8.497721	5.329571
3	STD		17.07418	0.773375	2.389348	5.433581	5.467005	2.619422	2.890941	15.38629	3.071051	3.720169	4.21916	4.012979	2.680411
4	N		28082	28082	28082	28082	28082	28082	28082	28082	28082	28082	28082	28082	28082
5	CORR	comp wela	1	0.879365	0.79382	0.911378	0.91003	0.801448	0.804296	0.769014	0.704118	0.672301	0.711203	0.670696	0.550902
6	CORR	avq Cr SCR	0.879365	1	0.611347	0.709321	0.710322	0.631002	0.656105	0.667414	0.59187	0.570545	0.623093	0.58483	0.50477
7	CORR	SBGRP RI CNT WA	0.79382	0.611347	1	0.726174	0.711074	0.589403	0.595323	0.609136	0.580391	0.553295	0.542402	0.533304	0.411494
8	CORR	SBGRP RI CNT RC	0.911378	0.709321	0.726174	1	0.818938	0.696967	0.678967	0.714093	0.664547	0.635615	0.649149	0.623924	0.499596
9	CORR	SBGRP RI CNT LA	0.91003	0.710322	0.711074	0.818938	1	0.697544	0.676118	0.673492	0.629514	0.592213	0.625213	0.580798	0.469159
10	CORR	SBGRP RI CNT WS	0.801448	0.631002	0.589403	0.696967	0.697544	1	0.655578	0.623732	0.564813	0.53582	0.590133	0.545843	0.443471
11	CORR	SBGRP RI CNT WC	0.804296	0.656105	0.595323	0.678967	0.676118	0.655578	1	0.658581	0.590696	0.569808	0.617304	0.5749	0.480413
12	CORR	SBGRP RI CNT TOT M	0.769014	0.667414	0.609136	0.714093	0.673492	0.623732	0.658581	1	0.85139	0.878653	0.90683	0.890571	0.784574
13	CORR	SBGRP RI CNT PS	0.704118	0.59187	0.580391	0.664547	0.629514	0.564813	0.590696	0.85139	1	0.716411	0.720278	0.698781	0.567208
14	CORR	SBGRP RI CNT NUM	0.672301	0.570545	0.553295	0.635615	0.592213	0.53582	0.569808	0.878653	0.716411	1	0.740369	0.712021	0.603582
15	CORR	SBGRP RI CNT AF	0.711203	0.623093	0.542402	0.649149	0.625213	0.590133	0.617304	0.90683	0.720278	0.740369	1	0.746301	0.661237
16	CORR	SBGRP RI CNT MG	0.670696	0.58483	0.533304	0.623924	0.580798	0.545843	0.5749	0.890571	0.698781	0.712021	0.746301	1	0.651398
17	CORR	SBGRP RI CNT AG	0.550902	0.50477	0.411494	0.499596	0.469159	0.443471	0.480413	0.784574	0.567208	0.603582	0.661237	0.651398	1
18	MEAN		48.98144	1.712316	5.887801	13.75045	15.09564	6.225906	7.749438						
19	STD		17.04355	0.769552	2.392502	5.425683	5.490388	2.638359	2.903199						
20	N		32487	32487	32487	32487	32487	32487	32487						
21	CORR	comp wela	1	0.875919	0.789976	0.910156	0.909317	0.801064	0.801575						
22	CORR	avq Cr SCR	0.875919	1	0.603345	0.703008	0.705027	0.625815	0.651045						
23	CORR	SBGRP RI CNT WA	0.789976	0.603345	1	0.723633	0.707189	0.585126	0.585547						
24	CORR	SBGRP RI CNT RC	0.910156	0.703008	0.723633	1	0.818009	0.695395	0.673223						
25	CORR	SBGRP RI CNT LA	0.909317	0.705027	0.707189	0.818009	1	0.695387	0.671609						
26	CORR	SBGRP RI CNT WS	0.801064	0.625815	0.585126	0.695395	0.695387	1	0.6579						
27	CORR	SBGRP RI CNT WC	0.801575	0.651045	0.585547	0.673223	0.671609	0.6579	1						
28	MEAN									39.74049	6.823488	8.615712	10.97693	8.202318	5.122041
29	STD									13.95365	2.895444	3.464525	3.936405	3.704246	2.508063
30	N									43051	43051	43051	43051	43051	43051
31	CORR	SBGRP RI CNT TOT M								1	0.821726	0.85519	0.890278	0.870124	0.751142
32	CORR	SBGRP RI CNT PS								0.821726	1	0.663947	0.669079	0.643191	0.500015
33	CORR	SBGRP RI CNT NUM								0.85519	0.663947	1	0.69232	0.662497	0.544954
34	CORR	SBGRP RI CNT AF								0.890278	0.669079	0.69232	1	0.704993	0.61359
35	CORR	SBGRP RI CNT MG								0.870124	0.643191	0.662497	0.704993	1	0.59985
36	CORR	SBGRP RI CNT AG								0.751142	0.500015	0.544954	0.61359	0.59985	1

Appendix B. The Excel Template for the Correlation Information

Intercorrelations[1] and Reliability Estimates By Section
**** **** Administration

	ELA[4]	Essay[3]	Word Anal.	Read. Comp.	Lit. Resp. & Anal.	Writing Strat.	Writi Conv.	Math	Prob. & Stat.	Number Sense	Alg. & Func.	Meas. & Geom.	Alg1
ELA[2]													
Essay													
Word Analysis													
Reading Comprehension													
Literary Responses & Analysis													
Writing Strategies													
Writing Conventions													
Math[3]													
Probability and Statistics													
Number Sense													
Algebra & Functions													
Measurement & Geometry													
Algebra 1													
Number of Items													
Mean													
SD													
Reliability													

[1] All correlations are for raw scores.

[2] Correlations for ELA section are reported for examinees completing ELA section. Number of examinees:

[3] Correlations for Math section are reported for examinees completing Math section. Number of examinees:

[4] Correlations between ELA and Math are reported for examinees taking both sections. Number of examinees: