

Paper BB-14

Using PROC SQL and PROC TRANSPOSE to provide Excel's Pivot functions

Mai Nguyen, Shane Trahan, Inga Allred, Nick Kinsey

ABSTRACT

The enormity of data used and collected in all levels of research is overwhelming; to many data analysts this deluge poses not only opportunities but can be a significant hindrance to figuring out "what does my data tell me?" Tools abound but many analysts just need something to get them started and many turn to one simple yet effective tool, Microsoft Excel's PivotTable® utility. The Excel's PivotTable is a versatile function allowing users to view data in a variety of different ways. Large datasets can be easily manipulated by filtering, transforming and aggregating information providing valuable insights including difficult to detect trend identification. Our paper will provide code and illustrate a method of using this data mining technique from Excel and firmly places this unique and simple to understand tool into the hands of SAS® developers.

We will step users through the use of PROC SQL and PROC TRANSPOSE to create a robust pivot table utility easily applied to a variety of SAS based applications. Our goal is to give users a tool that not only can be used across many types of data but also help identify important information to begin analysis.

INTRODUCTION

Managing large data sets can be a challenge and SAS offers many robust analytical tools. However many analysts prefer to see data in simple terms and in a format familiar to others outside of the field of analytics. The knowledge worker of today has grown accustomed to the power and ease of use of software packages such as Microsoft Excel which offers many mid-level analytical tools. With relatively few clicks of a pointing device, Excel users can whip up data summaries with relative ease and results can be copied and pasted into Microsoft Word documents and email messages. One of the most powerful data models available to analyze patterns of data is the Pivot Table. The Pivot Table is a data summarization tool or technique available in spreadsheet programs and in business intelligence software. This paper will introduce how to build PivotTable summaries using PROC SQL and PROC TRANSPOSE to summarize and evaluate data.

PROC SQL AND PROC TRANSPOSE

In our example, two Base SAS® procedures can be used in the creation of a PivotTable. PROC SQL is used to get the data we are summarizing and also includes the functions to aggregate data such as Average and Summations while PROC TRANSPOSE is used to convert data from rows to columns or vice versa. Both of these procedures are feature rich and provide effective analytical capabilities on their own but combining them provides analysts valuable insights into the discovery of information and data patterns. Each one of these procedures are described in further detail below.

PROC SQL

The PROC SQL procedure is SAS software's way of implementing Structured Query Language. The procedure is equipped with a variety of data manipulation tools. PROC SQL can generate reports, generate summary statistics, retrieve data from tables or views, combine data from tables or views, create tables, views, and indexes, update the data values, update and retrieve data from database management system (DBMS) tables, and modify a table by adding, modifying, or dropping columns. A PROC SQL table is the same as a SAS dataset; rows in the table are equivalent to SAS dataset observations and columns are equivalent to SAS dataset variables. To create a PROC SQL data view, the following code can be used.

```
PROC SQL;  
    SELECT Category  
    FROM SampleData;  
RUN;
```

This view will return the variable Category for all observations in the SampleData table. To create a data table, only one extra line of code is needed.

```
PROC SQL;
    CREATE TABLE CategorySales as
    SELECT Category
    FROM SampleData
RUN;
```

This code snippet will create a table called CategorySales containing the variable Category for all observations in the SampleData table. A WHERE condition may also be used to subset the incoming data to only those rows that meet a specified criteria.

There are several useful tools in PROC SQL statement. UPDATE is used to update existing rows of a table or view. INSERT is used to insert new rows into a table or view. DELETE is used to delete rows from a table or view. ALTER TABLE adds, modifies, and deletes columns in existing tables. These are just a few statements that are available for use in the PROC SQL procedure, there are many more.

Using the PROC SQL statement, data can be sorted. It can also be grouped together. Column, or variable, attributes such as length, format, informat and label, can also be added and/or modified.

```
PROC SQL;
    CREATE TABLE CategorySales AS
    SELECT Category, Product, Sum(Sales) as Sales
    FROM SampleData
    GROUP BY Category, Product
    ORDER BY Category, Sales Desc;
RUN;

PROC SQL;
    ALTER TABLE CategorySales
        ADD SalesPerson Character(25) label = "Sales Person";

    UPDATE CategorySales
    SET SalesPerson = "JOHN DOE";

    TITLE "Adding SalesPerson Column";
    SELECT Category, Product, Sales, SalesPerson
    FROM CategorySales;
RUN;
```

PROC TRANSPOSE

The PROC TRANSPOSE procedure reads in a dataset and rotates/pivots the specified rows and columns to form a new named dataset. What were once rows or observations in the input dataset are now columns and therefore become the variable names and what were once columns become the rows or observations of the new transposed SAS dataset. This is illustrated in the following code where the input dataset ProductSales is transposed and the output dataset ProductSalesByQuarters is produced.

```
* Pivot dataset by Quarter variable;
PROC TRANSPOSE DATA=ProductSales OUT=ProductSalesByQuarters NAME=Sales;
  BY Category Product;
  VAR Sales;
  ID Quarter;

RUN;
```

Input dataset before Proc Transpose

category	product	sales	quarter
Beverages	Chai	\$705.60	Qtr1
Beverages	Chai	\$878.40	Qtr2
Beverages	Chai	\$1,174.50	Qtr3
Beverages	Chai	\$2,128.50	Qtr4
Beverages	Chang	\$2,720.80	Qtr1
Beverages	Chang	\$228.00	Qtr2
Beverages	Chang	\$2,061.50	Qtr3
Beverages	Chang	\$2,028.25	Qtr4
Beverages	Chartreuse vert	\$590.40	Qtr1
Beverages	Chartreuse vert	\$360.00	Qtr2
Beverages	Chartreuse vert	\$1,100.70	Qtr3
Beverages	Chartreuse vert	\$2,424.60	Qtr4

The procedure knows the number of columns needed based on number of observations in the data, doing this with data steps would be highly inefficient. The BY statement in the code will create an observation for each variable value. The VAR statement tells SAS which variable to transpose and it should be noted that without the VAR statement, numeric variables not specified in the ID or BY statements will also be transposed.

Output Dataset after Proc Transpose

Category	Product	Sales	Qtr_1	Qtr_2	Qtr_3	Qtr_4
Beverages	Chai	Sales	\$706	\$878	\$1,175	\$2,129
Beverages	Chang	Sales	\$2,721	\$228	\$2,062	\$2,028
Beverages	Chartreuse verte	Sales	\$590	\$360	\$1,101	\$2,425

PROC SQL AND PROC TRANSPOSE TOGETHER

Using both of these procedures together we can create a summary table very similar to the results of a PivotTable in Excel. The following four examples walk through a simple PROC SQL statement outputting to Excel in Example 1 and wrapping up with a PROC SQL and PROC TRANSPOSE together to summarize data by particular variables in Example 4.

EXAMPLE 1 - SALES BY CATEGORIES

This example emphasizes the PROC SQL procedure to calculate a SUM of sales grouped by category. The figure below shows the results produced by Excel's PivotTable utility.

Category	Sales
Beverages	\$88,656
Condiments	\$34,006
Dairy Products	\$81,133
Grains/Cereals	\$53,372
Grand Total	\$257,167

Figure 1 Sum of Sales by Category

Similar output can be achieved by using PROC SQL as shown in Exhibit 1:

```

/**
** Excel Pivot Example 1 -- Category Sales
** Total Sales by Categories
**/

* Read sample data into the working dataset SampleData;
%include 'C:\Projects\Papers\SESUG 2011\ReadSampleData.SAS';U

* Generate Total Sales by Categories data using PROC SQL;
PROC SQL;      V
    CREATE TABLE CategorySales AS
    SELECT Category, SUM(Sales) as Sales format=DOLLAR20.0
    FROM SampleData
    GROUP BY Category;
QUIT;

* Output result to Excel;
PROC EXPORT DATA=CategorySales      W
    File='C:\Projects\Papers\SESUG 2011\PivotOutputBySAS.xls'
    dbms=excel
    replace;
RUN;

```

Exhibit 1 SAS Code Example 1

Exhibit 1 includes three distinct steps, which are described below:

1. Read input data into a working SAS dataset. This step is used throughout our examples and is shown in the References section.
2. Using PROC SQL with the GROUP BY clause to generate Sales by Categories data. SQL allows grouping by any number of variables, however variables in the GROUP BY clause must be contained in the SELECT list or an error will occur.
3. Output the results to Excel. Once in Excel, the values can be formatted or summarized as desired.

EXAMPLE 2 - SALES BY PRODUCTS

Taking the first example a step further, we add in additional GROUP BY variables in our PROC SQL statement. With that, we can break out our categories by product and as can be seen in Figure 2 the sales figures are further detailed. Now we know sales by category and by product.

Category	Product	Sales
Beverages		
	Chai	\$4,887
	Chang	\$7,039
	Chartreuse verte	\$4,476
	Côte de Blaye	\$46,563
	Ipoh Coffee	\$11,070
	Lakkalikööri	\$7,883
	Laughing Lumberjack Lager	\$910
	Outback Lager	\$5,828
Beverages Total		\$88,656
Condiments		
	Aniseed Syrup	\$1,724
	Chef Anton's Cajun Seasoning	\$5,215
	Genen Shouyu	\$1,475
	Grandma's Boysenberry Spread	\$2,500
	Gula Malacca	\$6,543
	Louisiana Fiery Hot Pepper Sauce	\$9,331
	Louisiana Hot Spiced Okra	\$2,958
	Northwoods Cranberry Sauce	\$4,260
Condiments Total		\$34,006
Dairy Products		

	Camembert Pierrot	\$20,652
	Fløtemysost	\$9,643
	Geitost	\$786
	Gorgonzola Telino	\$7,621
	Gudbrandsdalsost	\$14,042
	Mascarpone Fabioli	\$2,669
	Mozzarella di Giovanni	\$11,839
	Queso Cabrales	\$5,547
	Queso Manchego La Pastora	\$8,335
Dairy Products Total		\$81,133
Grains/Cereals		
	Filo Mix	\$2,068
	Gnocchi di nonna Alice	\$32,604
	Gustaf's Knäckebröd	\$4,234
	Ravioli Angelo	\$2,157
	Singaporean Hokkien Fried Mee	\$5,184
	Wimmers gute Semmelknödel	\$7,125
Grains/Cereals Total		\$53,372
Grand Total		\$257,167

Figure 2 Sum of Sales by Category and Product

This output can be achieved using a similar GROUP BY clause as in Example 1, but in this example, the GROUP BY is applied to both Category and Product variables. The SAS code below further illustrates the summarization capabilities of the aggregate functions and GROUP BY clauses in SQL.

```

/**
** Excel Pivot Example 2 -- Product Sales
** Total Sales by Products
**/

* Read sample data into the working dataset SampleData;
%include 'C:\Projects\Papers\SESUG 2011\ReadSampleData.SAS';
* Generate Total Sales by Products data using PROC SQL;
PROC SQL;
    CREATE TABLE ProductSales AS
    SELECT Category, Product, SUM(Sales) as Sales
format=DOLLAR20.0
    FROM SampleData
    GROUP BY Category, Product;
QUIT;

* Output result to Excel;
PROC EXPORT DATA=ProductSales
    File='C:\Projects\Papers\SESUG 2011\PivotOutputBySAS.xls'
    dbms=excel
    replace
    label;
RUN;

```

Exhibit 2 SAS Code Example 2

The SAS output of this program is shown below in Figure 2.1.

Category	Product	Sales
Beverages	Chai	4,887.00
Beverages	Chang	7,038.55
Beverages	Chartreuse verte	4,475.70
Beverages	Côte de Blaye	46,563.08
Beverages	Ipoh Coffee	11,069.90
Beverages	Lakkalikööri	7,883.10
Beverages	Laughing Lumberjack Lager	910.00
Beverages	Outback Lager	5,828.40
Condiments	Aniseed Syrup	1,724.00
Condiments	Chef Anton's Cajun Seasoning	5,214.88
Condiments	Genen Shouyu	1,474.82
Condiments	Grandma's Boysenberry Spread	2,500.00
Condiments	Gula Malacca	6,543.43
Condiments	Louisiana Fiery Hot Pepper Sauce	9,331.08
Condiments	Louisiana Hot Spiced Okra	2,958.00
Condiments	Northwoods Cranberry Sauce	4,260.00
Dairy Products	Camembert Pierrot	20,652.28
Dairy Products	Fløtemysost	9,642.74
Dairy Products	Geitost	786.00
Dairy Products	Gorgonzola Telino	7,620.74
Dairy Products	Gudbrandsdalsost	14,041.80
Dairy Products	Mascarpone Fabioli	2,668.80
Dairy Products	Mozzarella di Giovanni	11,838.60
Dairy Products	Queso Cabrales	5,546.94
Dairy Products	Queso Manchego La Pastora	8,335.30
Grains/Cereals	Filo Mix	2,068.15
Grains/Cereals	Gnocchi di nonna Alice	32,604.00

Figure 2.1 SAS Output Result Example 2

EXAMPLE 3 - TOP THREE PRODUCTS

With very large datasets, we can still be overwhelmed even when using the summarization toolsets that have been previously discussed. In this case, we can add additional criteria to return less data and allow a more focused analysis. This can be done by simply adding a WHERE clause to our SQL statement but what if we want to look at the top three products by sales in each category? Using a macro as shown in this example we can evaluate each category and with the OUTOBS option we can pull out the top three products by sales in each category. Figure 3 below shows the output produced by Excel.

Category	Product	Sales
Beverages		
	Côte de Blaye	\$46,563
	Ipoh Coffee	\$11,070
	Lakkalikööri	\$7,883
Beverages Total		\$65,516
Condiments		
	Chef Anton's Cajun Seasoning	\$5,215
	Gula Malacca	\$6,543
	Louisiana Fiery Hot Pepper Sauce	\$9,331
Condiments Total		\$21,089
Dairy Products		
	Camembert Pierrot	\$20,652
	Gudbrandsdalsost	\$14,042
	Mozzarella di Giovanni	\$11,839
Dairy Products Total		\$46,533
Grains/Cereals		
	Gnocchi di nonna Alice	\$32,604
	Singaporean Hokkien Fried Mee	\$5,184
	Wimmers gute Semmelknödel	\$7,125
Grains/Cereals Total		\$44,914
Grand Total		\$178,052

Figure 3 Top Three Products in Each Category by Sales

The code below in Exhibit 3 steps us through a series of PROC SQL statements and the creation of a Macro that will give our desired criteria of top three products by sales for each category:

```

/**
** Excel Pivot Example 3 -- Top 3 Product Sales by Categories
** Top 3 Product Sales by Categories
**/

%let TopN = 3;

* Read sample data into the working dataset SampleData;
%include 'C:\Projects\Papers\SESUG 2011\ReadSampleData.SAS';

* Sort Total Sales by Products data using PROC SQL;
PROC SQL;    U
    CREATE TABLE ProductSales AS
    SELECT Category, Product, SUM(Sales) as Sales format=DOLLAR20.0
    FROM SampleData
    GROUP BY Category, Product
    ORDER BY Category, Sales Desc;
QUIT;

* select year into macro variables for processing;
PROC SQL NOPRINT;    V
    SELECT COUNT(DISTINCT Category) INTO :n_Categories
    FROM ProductSales;

    SELECT DISTINCT '"' || Category || '"' INTO :Category1 -
:Category%left(&n_Categories)
    FROM ProductSales;
QUIT;

* SELECT TopN ROWS INTO THE RESULT DATASET;
%macro Top_N_By_Group;    W
PROC SQL;
    CREATE TABLE Top3Products LIKE ProductSales;
QUIT;

%do i=1 %to &n_Categories;
    PROC SQL OUTOBS = &TopN;
        INSERT INTO Top3Products
        SELECT * FROM ProductSales
        WHERE Category = &&Category&i;
    QUIT;
%end;
%mend Top_N_By_Group;

%Top_N_By_Group;

* Output result to Excel;
PROC EXPORT DATA=Top3Products •
    File='C:\Projects\Papers\SESUG 2011\PivotOutputBySAS.xls'
    dbms=excel
    replace
    label;
RUN;

```

Exhibit 3 SAS Code Example 3

The Exhibit 3 SAS program includes four distinct steps which are described below:

1. Obtain total sales by products in each category using multiple variables in the GROUP BY statement.
2. Select number of categories and the category names into a macro variables used in the Top_N_By_Group Macro created in step 3.
3. The Top_N_By_Group macro loops through the dataset and select top three products by total sales for each category using the OUTOBS option.
4. Output the results to Excel. Once in Excel, the values can be formatted or summarized.

The SAS output of this program is shown below in Figure 3.1.

Category	Product	Sales
Beverages	Côte de Blaye	46,563.08
Beverages	Ipoh Coffee	11,069.90
Beverages	Lakkalikööri	7,883.10
Condiments	Louisiana Fiery Hot Pepper Sauce	9,331.08
Condiments	Gula Malacca	6,543.43
Condiments	Chef Anton's Cajun Seasoning	5,214.88
Dairy Products	Camembert Pierrot	20,652.28
Dairy Products	Gudbrandsdalsost	14,041.80
Dairy Products	Mozzarella di Giovanni	11,838.60
Grains/Cereals	Gnocchi di nonna Alice	32,604.00
Grains/Cereals	Wimmers gute Semmelknödel	7,125.47
Grains/Cereals	Singaporean Hokkien Fried Mee	5,184.20

Figure 3.1 SAS Output Result from Top_N_By_Group Macro

EXAMPLE 4 – PRODUCT SALES BY QUARTERS

Using the same dataset we can break this information out by Quarter. Listed below are the sales figures by product and category organized by quarter. Notice that the total sales data has been “morphed” from a single column into four columns, one for each quarter.

Sales						
Category	Product	Quarter				
		Qtr 1	Qtr 2	Qtr 3	Qtr 4	Grand Total
Beverages						
	Chai	\$706	\$878	\$1,175	\$2,129	\$4,887
	Chang	\$2,721	\$228	\$2,062	\$2,028	\$7,039
	Chartreuse verte	\$590	\$360	\$1,101	\$2,425	\$4,476
	Côte de Blaye	\$25,127	\$12,806	\$7,312	\$1,318	\$46,563
	Ipoh Coffee	\$1,398	\$4,497	\$1,196	\$3,979	\$11,070
	Lakkalikööri	\$1,142	\$1,774	\$3,262	\$1,706	\$7,883
	Laughing Lumberjack Lager		\$518	\$350	\$42	\$910
	Outback Lager	\$1,508	\$384	\$1,253	\$2,684	\$5,828
Beverages Total		\$33,193	\$21,445	\$17,709	\$16,309	\$88,656
Condiments						
	Aniseed Syrup	\$544	\$600	\$140	\$440	\$1,724
	Chef Anton's Cajun Seasoning	\$225	\$2,970	\$1,338	\$682	\$5,215
	Genen Shouyu		\$177	\$1,298		\$1,475
	Grandma's Boysenberry Spread			\$1,750	\$750	\$2,500
	Gula Malacca	\$1,995	\$1,754	\$1,093	\$1,702	\$6,543
	Louisiana Fiery Hot Pepper					
	Sauce	\$1,347	\$2,151	\$1,976	\$3,857	\$9,331
	Louisiana Hot Spiced Okra	\$816	\$1,224		\$918	\$2,958
	Northwoods Cranberry Sauce		\$1,300		\$2,960	\$4,260
Condiments Total		\$4,927	\$10,175	\$7,594	\$11,309	\$34,006
Dairy Products						
	Camembert Pierrot	\$3,329	\$3,990	\$10,273	\$3,060	\$20,652
	Fløtemysost	\$4,455	\$174	\$2,541	\$2,473	\$9,643
	Geitost	\$294	\$243	\$100	\$150	\$786
	Gorgonzola Telino	\$487	\$2,993	\$1,459	\$2,682	\$7,621
	Gudbrandsdalsost	\$2,650	\$1,267	\$4,473	\$5,652	\$14,042

	Mascarpone Fabioli		\$2,221		\$448	\$2,669
	Mozzarella di Giovanni	\$1,974	\$4,488	\$3,028	\$2,349	\$11,839
	Queso Cabrales	\$1,357	\$3,029	\$504	\$656	\$5,547
	Queso Manchego La Pastora	\$456	\$1,397	\$1,163	\$5,320	\$8,335
Dairy Products Total		\$15,002	\$19,802	\$23,540	\$22,790	\$81,133
Grains/ Cereals						
	Filo Mix	\$188	\$742	\$227	\$912	\$2,068
	Gnocchi di nonna Alice	\$6,931	\$9,869	\$6,772	\$9,033	\$32,604
	Gustaf's Knäckebröd	\$202	\$504	\$3,318	\$210	\$4,234
	Ravioli Angelo	\$499	\$88	\$585	\$985	\$2,157
	Singaporean Hokkien Fried Mee	\$986	\$913	\$2,307	\$979	\$5,184
	Wimmers gute Semmelknödel	\$2,913	\$1,736	\$1,679	\$798	\$7,125
Grains/Cereals Total		\$11,718	\$13,851	\$14,888	\$12,916	\$53,372
Grand Total		\$64,840	\$65,273	\$63,731	\$63,323	\$257,167

Figure 4 Quarterly Sales by Product and Category in Excel

We can produce a similar output result with the SAS program below, using a combination of PROC SQL and PROC TRANSPOSE.

```

/**
** Excel Pivot Example 4 -- Product Sales by Quarters
** Product Sales by Quarters
**/

* Read sample data into the working dataset SampleData;
%include 'C:\Projects\Papers\SESUG 2011\ReadSampleData.SAS';

* Generate Total Sales by Products data using PROC SQL;
PROC SQL;
    CREATE TABLE ProductSales AS
    SELECT Category, Product, Quarter, SUM(Sales) as Sales
    format=DOLLAR20.0
    FROM SampleData
    GROUP BY Category, Product, Quarter;
QUIT;

* Pivot dataset by Quarter variable;
PROC TRANSPOSE DATA=ProductSales OUT=ProductSalesByQuarters NAME=Sales; V
    BY Category Product;
    VAR Sales;
    ID Quarter;

RUN;

PROC SQL;
    ALTER TABLE ProductSalesByQuarters DROP Sales;
QUIT;

* Output result to Excel;
PROC EXPORT DATA=ProductSalesByQuarters
    File='C:\Projects\Papers\SESUG 2011\PivotOutputBySAS.xls'
    dbms=excel
    replace
    label;

RUN;

```

Exhibit 4 SAS Code Example 4

The three main steps in this program are:

1. First aggregate total sales by product by quarter.
2. Using PROC TRANSPOSE to pivot data with the Quarter variable.
3. Drop the Sales field from the ProductSalesByQuarters table for a cleaner summary data.

The SAS output of this program is shown below in Figure 4.1. Notice the empty cells for quarters with no sales.

Category	Product	Qtr 1	Qtr 2	Qtr 3	Qtr 4
Beverages	Chai	705.60	878.40	1,174.50	2,128.50
Beverages	Chang	2,720.80	228.00	2,061.50	2,028.25
Beverages	Chartreuse verte	590.40	360.00	1,100.70	2,424.60
Beverages	Côte de Blaye	25,127.36	12,806.10	7,312.12	1,317.50
Beverages	Ipoh Coffee	1,398.40	4,496.50	1,196.00	3,979.00
Beverages	Lakkalikööri	1,141.92	1,774.08	3,261.60	1,705.50
Beverages	Laughing Lumberjack Lager		518.00	350.00	42.00
Beverages	Outback Lager	1,508.40	384.00	1,252.50	2,683.50
Condiments	Aniseed Syrup	544.00	600.00	140.00	440.00
Condiments	Chef Anton's Cajun Seasoning	225.28	2,970.00	1,337.60	682.00
Condiments	Genen Shouyu		176.70	1,298.12	
Condiments	Grandma's Boysenberry Spread			1,750.00	750.00
Condiments	Gula Malacca	1,994.85	1,753.62	1,093.09	1,701.87
Condiments	Louisiana Fiery Hot Pepper				
Condiments	Sauce	1,347.36	2,150.77	1,975.54	3,857.41
Condiments	Louisiana Hot Spiced Okra	816.00	1,224.00		918.00
Condiments	Northwoods Cranberry Sauce		1,300.00		2,960.00
Dairy Products	Camembert Pierrot	3,329.28	3,989.90	10,273.10	3,060.00
Dairy Products	Fløtemysost	4,454.80	174.15	2,541.29	2,472.50
Dairy Products	Geitost	294.00	242.50	99.50	150.00
Dairy Products	Gorgonzola Telino	487.00	2,993.12	1,458.75	2,681.87
Dairy Products	Gudbrandsdalsost	2,649.60	1,267.20	4,473.00	5,652.00
Dairy Products	Mascarpone Fabioli		2,220.80		448.00
Dairy Products	Mozzarella di Giovanni	1,973.80	4,488.20	3,027.60	2,349.00
Dairy Products	Queso Cabrales	1,357.44	3,029.25	504.00	656.25
Dairy Products	Queso Manchego La Pastora	456.00	1,396.50	1,162.80	5,320.00
Grains/Cereals	Filo Mix	187.60	742.00	226.80	911.75
Grains/Cereals	Gnocchi di nonna Alice	6,931.20	9,868.60	6,771.60	9,032.60
Grains/Cereals	Gustaf's Knäckebröd	201.60	504.00	3,318.00	210.00
Grains/Cereals	Ravioli Angelo	499.20	87.75	585.00	984.75
Grains/Cereals	Singaporean Hokkien Fried Mee	985.60	912.80	2,307.20	978.60
Grains/Cereals	Wimmers gute Semmelknödel	2,912.70	1,735.65	1,679.12	798.00

Figure 4.1 SAS Output Quarterly Sales by Product and Category

CONCLUSION

Excels PivotTable summary functions are useful for finding pattern matches and assists data analysts when mining data for information. We have shown that the same pivot table functionality commonly utilized by many Excel users can also be done in SAS with very common SAS procedures. The four examples discussed here have shown a very simple pivot table to a more robust and complex summary. Each one of these examples can be easily expanded on and enhanced to use more advanced SQL potentially reaching across multiple data sources including Oracle databases, Microsoft Excel files or even comma delimited text files. Using PROC TRANSPOSE and PROC SQL together will allow users to view data in many different ways and the code presented here demonstrates how easy it is to customize output of data analysis summaries.

CONTACT INFORMATION

We welcome any feedback and comments on the work presented here and look forward to sharing this with those members of the SAS user group community. Questions and comments may be directed to any of the following authors listed below.

Mai Nguyen RTI International 3040 E. Cornwallis Rd RTP, NC 27709 Phone 919-541-8757 Fax: 919-541-6178 mnguyen@rti.org	Shane Trahan RTI International 3040 E. Cornwallis Rd RTP, NC 27709 Phone 919-541-5848 Fax: 919-541-6178 srt@rti.org
Inga Allred RTI International 3040 E. Cornwallis Rd RTP, NC 27709 Phone 919-541-7072 Fax: 919-541-6178 irb@rti.org	Nick Kinsey RTI International 3040 E. Cornwallis Rd RTP, NC 27709 Phone 919-541-6791 Fax: 919-541-6178 nlk@rti.org

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.

REFERENCES

SAMPLE DATA

Sample data titled "Excel 2002 Sample: PivotTable Reports" was used in this paper can be downloaded directly from Microsofts website at <http://www.microsoft.com/download/en/confirmation.aspx?id=14738>. A sample of the used data is included below.

Category	Product	Sales	Quarter
Beverages	Chai	\$705.60	Qtr 1
Beverages	Chai	\$878.40	Qtr 2
Beverages	Chai	\$1,174.50	Qtr 3
Beverages	Chai	\$2,128.50	Qtr 4
Beverages	Chang	\$2,720.80	Qtr 1
Beverages	Chang	\$228.00	Qtr 2
Beverages	Chang	\$2,061.50	Qtr 3
Beverages	Chang	\$2,028.25	Qtr 4
Beverages	Chartreuse verte	\$590.40	Qtr 1
Beverages	Chartreuse verte	\$360.00	Qtr 2
(More data...)			

COMMON CODES

The code below reads the sample data from an Excel spreadsheet into a working dataset called SampleData.

```

/**
** Read data from Excel spreadsheet into a working dataset
**/

PROC IMPORT
    DATAFILE="C:\Projects\Papers\SESUG 2011\SampleData.xls"
    OUT=WORK.SampleData
    REPLACE
    DBMS=EXCEL;
    RANGE="'Source Data$'";
    GETNAMES=YES;
    USEDATE = NO;
    SCANTIME = NO;
RUN;

PROC DATASETS LIBRARY=WORK NOLIST;
    MODIFY SampleData;
        FORMAT
            Category    $CHAR20.
            Product     $CHAR40.
            Sales        DOLLAR20.2
            Quarter      $CHAR5. ;
        INFORMAT
            Category    $CHAR20.
            Product     $CHAR40.
            Sales        DOLLAR20.
            Quarter      $CHAR5. ;
QUIT;

```