Paper 165-2012

Bringing Optimization to the Business: Interfacing SAS/OR[®] with SAS[®] Stored Processes, Microsoft Excel, SAS[®] Enterprise Guide[®], SAS[®] Forecast Studio, and Microsoft Project

Andrew Pease, SAS Institute Inc., Cary, NC

ABSTRACT

SAS/OR® software provides a powerful array of optimization, simulation, and project scheduling techniques to identify the actions that will produce the best results, while operating within resource limitations and tight restrictions. This paper focuses on ways to get this information to business users in an intuitive and interactive way. Via SAS® Stored Processes, users can change parameters and run different configurations of a base SAS/OR program. These stored processes can be surfaced via the SAS® Add-In for Microsoft Office, SAS® Enterprise Guide®, and SAS® Forecast Studio. Also, via Microsoft Project macros, Microsoft Project users can export parameters to SAS/OR, which produces optimized planning, and then push back to the Microsoft Project user. The technical setup of these various channels is explored in demonstrations.

INTRODUCTION

Optimization has often been described as prescriptive analytics and as the top level on the data and analytic continuum. Despite the perceived and demonstrated value that it can bring, it remains an underused analytic capability in the SAS® analytics arsenal. Part of the problem is because of the perceived and real complexity in operations research problem formulation and execution. However, this complexity can be masked to the business user by presenting optimization algorithms via simplified, ready-for-business interfaces.

This paper presents the necessary steps for developing interfaces to SAS/OR programs for use by business users.

The steps include the following:

- 1. State the business challenge.
- 2. Formulate the mathematical model.
- 3. Program the mathematical model.
- 4. Parameterize the program.
- 5. Create a SAS stored process.
- 6. Use it.

For the purposes of this paper, the descriptions of the first three steps are brief. These steps are generic to any operations research application, and they are better outlined elsewhere. That said, because these steps are important to any successful business utilization, the key points are emphasized.

For an introduction to optimization with SAS software, see the "Optimization with SAS® Software" white paper, available at http://www.sas.com/reg/wp/corp/17649.

For more information about SAS stored processes, see http://support.sas.com/documentation/cdl/en/stpug/62758/PDF/default/stpug.pdf.

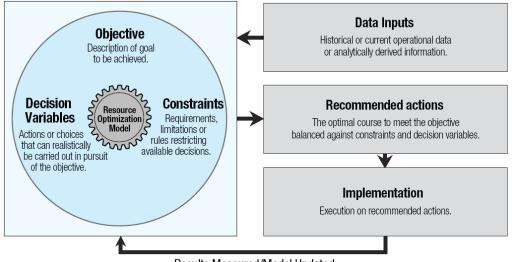
STATE THE BUSINESS CHALLENGE AND FORMULATE THE MATHEMATICAL MODEL

Optimization is at the top level on the data and analytic continuum.

Figure 1. Data and Analytic Continuum

Nothing can be optimized without first stating the business challenge. In other words, "What is <u>it</u> that we want to optimize?" Initially, stating the business challenge has little to do with math and even less to do with programming. It is a simple, plain language description of the business system to be modeled, including the objective, control factors, and constraints. It might at first seem simplistic to mention this as a key step in the optimization process. But, in fact, many optimization efforts fail because of a lack of consensus or completeness in defining the business system and the objective.

The key components to a complete business challenge statement have been described in the "Aligned Resource Optimization" white paper, available at http://www.sas.com/reg/wp/corp/4183. In this paper, the following model is presented, including a stated objective, decision variables, and constraints.



Results Measured/Model Updated

Figure 2. Resource Optimization Model

In the following table, examples are provided. In each of the specific business challenge statements, there is a clearly stated objective highlighted in green, with various decision variables highlighted in blue, and constraints highlighted in red.

Business Domain	High-Level Challenge Statement	Specific Business Challenge Statement
Supply Chain	Reduce the cost of lost or dumped stock.	Minimize waste by setting optimal stock levels to meet stated service levels.
	Increase product availability.	Maximize product availability by setting optimal stock levels while respecting inventory cost and infrastructure limitations.
	Reduce supplier costs.	Minimize supplier costs by using an optimal supplier mix based on cost and ability to deliver while respecting stated service levels.
Marketing	Increase return on marketing investment.	Maximize campaign response rate by setting optimal contact schedule (including channel, message, and timing) while respecting budget constraints and contact policies.
	Decrease marketing costs.	Minimize marketing costs by setting optimal contact schedule (including channel, message, and timing) while respecting established contact policies.
	Re-engineer contact policies to have a more customer-focused approach.	Minimize total number of contacts by setting optimal contact schedule (including channel, message, and timing) while respecting established contact policies and individual campaign objectives.
Production	Increase production line efficiency.	Minimize system down time due to production shifts by setting optimal production times while respecting projected demand amounts and operational constraints.
	Reduce production costs.	Reduce production costs by minimizing waste by defining optimal use of raw materials while respecting projected demand amounts and operational constraints.
Project Management	Reduce project time.	Minimize elapsed project time by allowing for concurrent project tasks in the scheduling while respecting required task completion order.

Table 1. Example Challenge Statements

Reflect on the examples in this table as you develop a business challenge statement and problem formulation for a single, real-life case.

Optimizing a Credit Marketing Campaign

The marketing department of a major bank is looking to expand its credit business.

This might not seem like an appropriate objective for our troubled economic climate. However, credit remains a very profitable business. Financial organizations need to continually secure safe and profitable credit business, without over-contacting high-potential customers. Getting the contact mix just right can make the difference between acquiring good customers and simply picking up the more risky scraps.

So, expand on this simple objective with a more specific objective.

BUSINESS CHALLENGE: Maximize the return on credit business by making a constrained number of offers to existing customers, optimized on default and incurred loss probabilities and propensity to respond.

That's a concrete business challenge with some key elements for the optimization. The objective is to maximize the return on new credit business. The control variables are the offer and customer pairs: the decision of which customer receives which offer. The constraints are the number of offers to make and the acceptable loss levels.

PROBLEM FORMULATION

Maximize Expected_Profit = Exp_Profit[customer,offer] * chosen_combinations[customer,offer], which is subject to maximum and minimum number of contacts, and limit the maximum amount of risk exposure.

The expected profit for each customer and each offer is derived from a series of predictive models.

This simple credit marketing campaign business example is used for the remainder of this paper.

PROGRAM THE MATHEMATICAL MODEL

Now that there is a clear problem formulation, use PROC OPTMODEL to drive the optimization.

First, read in the relevant data. The data should contain the problem parameters.

```
proc optmodel;
    /* declare sets and parameters */
    set Customers;
    set Offers = 1..3;
    num Credit_Loss_Limit init 100000 ;
    num Minimum_Offer_Level = 0;
    num Exp_Profit{Customers,Offers}, Credit_Loss{Customers,Offers};
    /* read data from SAS data sets */
    read data Opt.Bank_Offers into Customers=[ID]
    {j in Offers} <Exp_Profit[ID,j]=col('exprofit'||j)
        Credit_Loss[ID,j]=col('creditloss'||j)>;
```

bank_offers -

1	id 😡	creditloss1	creditloss2	creditloss3	exprofit1	exprofit2	exprofit3	decision
1	10001	125.84015439	62.920077194	106.96413123	-1.321316619	-1.035642274	0.1896791579	
2	10003	133.13669262	66.568346308	113.16618872	0.0243399037	-2.837544966	-0.136592398	•
3	10005	121.52743561	60.763717804	103.29832027	-0.11892676	-1.619434772	0.1264097883	
Ļ	10006	138.3747094	69.187354701	117.61850299	-0.442538855	-0.850556301	-0.071936622	•
5	10008	148.23540138	74.11770069	126.00009117	-0.51855832	-2.862581071	-0.079484402	
5	10009	127.02802833	63.514014167	107.97382408	-1.397183613	-1.848022056	0.1323977009	
7	10011	143.80010524	71.900052622	122.23008946	0.1987336163	-1.79391068	-0.371282671	
3	10014	140.60470369	70.302351847	119.51399814	-0.331607504	-0.448644439	0.0913721682	••••••••••••••••••••••••••••••••••••••
)	10019	123.76535827	61.882679137	105.20055453	0.1831902447	-1.037221612	-0.041343616	
0	10020	144.46968288	72.234841442	122.79923045	-1.53616152	-1.328196204	-0.030192304	•
1	10024	138.19941056	69.099705282	117.46949898	-1.208189544	0.4325687091	-0.354208986	
2	10028	149.70925433	74.854627164	127.25286618	0.3239471761	-1.444832742	-0.494162423	•
13	10029	151.42617437	75.713087187	128.71224822	0.2400992435	-1.741863044	-0.461551107	
4	10031	125.12382509	62.561912545	106.35525133	-1.206858013	-1.447645702	0.2776241932	
15	10032	155.79950167	77.899750837	132.42957642	-0.091966956	-1.489458516	-0.464689215	
6	10033	144.60348733	72.301743664	122.91296423	-1.282231073	-1.688703064	-0.050889686	¢
7	10040	124.45123895	62.225619475	105.78355311	-0.30659943	-1.734772534	-0.057120755	
8	10041	138.86907739	69.434538694	118.03871578	-1.330455058	-0.801388672	0.1497782995	0
9	10043	126.71846608	63.359233038	107.71069616	0.61412182	-0.474446431	-0.388632109	
20	10044	150.14535846	75.072679232	127.6235547	0.3624314827	-1.695022047	-0.35098978	•

Figure 3. Table View of Opt.Bank_Offers

The input table contains calculated loss given default (LGD) values and expected profits for each customer. These figures have been calculated using predictive models.

The control variables are next. In this case, the control variables are the decision variables for each customer and offer combination.

```
/* declare variables */
var x{Customers,Offers} binary;
```

It is useful to program the three constraints—the minimum and maximum number of offers for each customer to define the contact policies, and the total accepted project credit loss.

```
/* declare constraints */
con At Most One Offer{i in Customers}:
    sum{j in Offers} x[i,j] = 1;
con Minimum_Offers{j in Offers}:
    sum{i in Customers} x[i,j] >= Minimum_Offer_Level;
con Maximum_Credit_Loss:
    sum{i in Customers, j in Offers} Credit Loss[i,j] * x[i,j] <= Credit Loss Limit;</pre>
```

The numeric variables have already been created with the NUM statements. This makes the ensuing parameterization of these values much easier.

Next comes the mathematical expression of the objective. This contains both the mathematical relationship between the objective parameter and the indicative direction (MAX or MIN).

```
/* declare objective */
max Expected_Profit = sum{i in Customers, j in Offers}
    Exp_Profit[i,j] * x[i,j];
```

All that remains is to solve.

solve;

You can specify which solver PROC OPTMODEL should use. But, for now, the SOLVE statement is enough. The results of the optimization can be exported.

```
print Minimum_Offers.body;
print Maximum_Credit_Loss.body;
create data offers from [Customers Offers] x;
```

quit;

There is now a complete list of which customers receive which offers. This information can be used by marketing to execute the campaign.

PARAMETERIZE THE PROGRAM

It is essential to have the entire OPTMODEL program running and generating relevant results before you start to parameterize the program for users. Decide which parameters you want to give the user control over. In the current business example, perhaps the user wants to play with the number of offers or the acceptable level of credit risk.

The first step is making macro variables from the hardcoded values in the program.

For example, replace the numeric value assigned to Credit_Loss_Limit with the macro variable & Credit_Loss_Limit.

num Credit_Loss_Limit init &Credit_Loss_Limit ;

Replace the numeric value assigned to Minimum_Offer_Level with the macro variable & Minimum_Offer_Level.

num Minimum Offer Level = &Minimum Offer Level ;

Test the program again to make sure there are no mistakes with the replacement macro variables. For now, use %LET statements to assign values to the newly created macro variables.

%let Minimum Offer Level=0 ;

You now have a program from which you can create a stored process.

CREATE A SAS STORED PROCESS

If you have not written your program in SAS Enterprise Guide, you need to copy or open your debugged program in SAS Enterprise Guide. Then, right-click on the program in a SAS Enterprise Guide diagram, and select **Create Stored Process**.

	Tools Help	🛛 - 🚘 - 🚳) 📇 sé Da da 🗙 (ini de i 🗁 - Equicant che dynamice -	1014
hogest Tree + H	Credit Offer C	marian +	
and Advance	in Bas + 10	y Synt + Schedde + Zoon + 🔂 Payettay 🔝 Payetles -	
Regions	(727)		
Teg Could Offer Optimization			
Pogami	Propuell		
Rappent Program	1000	Open +	
CodtOferOptionation2		fapet +	
		Sas Properti	
C Programs		Saw Programil Ac.	
A constant		Ran Program11 on SASAge	
		Rue Bireck true ProgramM	
		Select Server +	
		Condition >	
	1	Civits Stand Process	
		Sent Ta +	
		Maue Programili to Process Flow	
		Link Progravill Na.	
nerLid + H		Copy	
		Tata	
Tetresk Discoveret in Stap		Delete	
	1	Interine	
E II Seven			
CI 😏 Libraries 18 🥶 MAPS		Propeties	
III 🗃 MAPSGEK			
III III MAPESAS III III SASAve - SASDATA			
SASAgo - writerio SASAgo - writerio			
A SLEER			
STP Sargina			
DLAP Servers Private OLAP Servers			
D Develop Color Seven			
	Task Status		
	Task	Sana	Queue Server
	*		
хбу			R Connecters SAS Dense Uner, SRXDE
9 🛗 🍳 🌝	2 1	🛋 🛃 👑	

Figure 4. Start the Stored Process Creation Task

On the first page, specify a name and location for your stored process. For more information about configuring your environment for stored processes, see http://support.sas.com/documentation/cdl/en/stpug/62758/PDF/default/stpug.pdf.

1 of 6 Name and Description		sas
Save Stored Process as:		
Name		
Stored Process for Program11		
Location:		
/User Folders/sasdemo/My Folder		Browse
(Example: /8IP Tree/My Folder Name)		
Description:		
Keywords	Responsbilties:	
Keywords Keywords (one per line)	Responsbilities: Name	Role
		Role
	Nano	Role Responsibility.) Delete Responsibility.
Keywords (one per line)	Nano	

Figure 5. First Page—Name and Location

The next page provides a view of the code that you want to use for the stored process creation. It is important to drop the LIBNAME and macro variable assignments. These assignments will be handled by the parameters that you will create subsequently.



Figure 6. Second Page—Code View

The wizard asks you to set the run-time parameters. At this point, the defaults are acceptable.

Create New SAS Stored Process Wizar	d 📃
3 of 6 Execution Options	<u>S</u> .sas
Save SAS Stored Process Code	
Execution server	
SASApp - Logical Stored Process Server	
Location on Server:	
🧀 Source filepath:	
C:\SAS\Config\Lev1\SASApp\StoredPr	roceesServer'assuser 🔹
Source filename	
Stored_Process_for_Program11.ses	
SAS Read: Types this Stared Process car	
Streaming (only available on Stored Pro	
SAS Result Types this Sovied Process car Soviet Seasong (only invalidation of Stand Tro Prockage	
Streaming (only available on Stored Pro	

Figure 7. Third Page—Run-Time Parameters

The wizard asks whether you want to make parameters from the scanned program. Basically, with this option, the program is scanned for macro variables, and you can create parameters from these for use in your stored process. Here is an example of making a parameter for the macro variable value Credit_Loss_Limit:

4 of 6 Pro	mpts		<u>S</u> .sas	
nput Prompts:				
Displayed Text	Name	Туре	New -	
🖓 General		Standard group	New Prompt	7
			Prompt from SAS Code for	Multiple
			Project prompt copy	Credit_Loss_Limit
			🔊 New Group	Minimum_Offer_Leve
			Preview	
			Delete	
Output Parameters				
Name	Туре	Displayed Text	New	
			Edt	
< [,	Delete	
eates a new prom	pt or group. You can add a prompt in prompt from the project. You can als	three ways: by creating a new prom o organize these prompts into groups		
)		2 goop	More (F1)	

Figure 8. Fourth Page—Prompt for Parameters

After selecting the parameter that you want to use, the wizard enables you to set restrictions and types for the parameter values that will be used during the stored process execution. You should focus on the second tab, where you can change a few characteristics of the parameter vales.

- Make the **Prompt type Numeric**.
- Select User enters values for Method for populating prompt.
- Select values for Maximum number of decimal places allowed, Minimum value allowed, Maximum value allowed, and Default Value.

General	Prompt Type and Values	Dependencies	
Prompt	type:		
Numeri	c	•	
Method	for populating prompt:	Number of values:	
User er	nters values	✓ Single value	
Allon	w only integer values		
Minimur	n number of decimal places	allowed: Maximum number of decimal places al	lowed:
		1	
Minimur	n value allowed:	Maximum value allowed:	
50000		500000	
Default 150000	107071010		

Figure 9. Specify Parameter Characteristics

Repeat this process for the second parameter, ${\tt Minimum_Offer_Level}.$

For the last two pages, select the defaults, and click **Finish**.

Data Source	es (input streams to a s	tored process):		
Fileref	Content	Label	Description	New
)ata Tarnet	s (output streams from	a stored process):		Delete
Fileref	Content	Label	Description	New
				Edit
ts any data eams.	targets where you wa	nt to send output when ti	ne stord process runs. These data ta	Delete

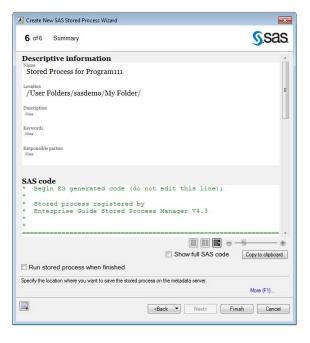


Figure 10. Last Pages of Stored Process Creation

You have created a stored process. You can test its execution by right-clicking on the stored process icon in SAS Enterprise Guide, and selecting **Run Credit Offer Optimization**.

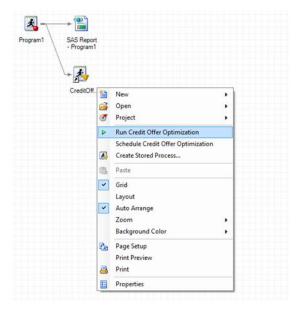


Figure 11. Test the Stored Process

The execution begins with a request for the parameter values that need to be specified.

General	Reset group defaults
Credit Loss Limit Maximum allowed credit loss per offer.	
150000.0	
* Minimum Offer Level	
200.0	

Figure 12. Specify Parameter Values

If you simply click **Run**, then the stored process executes with the default values. In a few seconds, you will see the output of the optimization process and the data set containing individual customer and offer combinations.

Problem Sumn	агу
Objective Sense	Maximization
Objective Function	Expected_Profit
Objective Type	Linear
Number of Variables	5136
Bounded Above	0
Bounded Below	0
Bounded Below and Above	5136
Free	0
Fixed	0
Binary	5136
Integer	0
Number of Constraints	1716
Linear LE (<=)	1
Linear EQ (-)	1712
Linear GE (>=)	3
Linear Range	0
Constraint Coefficients	15408

Credit	OfferOptimization2	•										
Ξ.	og 🛃 Output Data	😫 Results										
6	Modify Task 🐺	Filter and Sort	Query Builder	Data - Describe -	Graph - Analyze	•	Export - Send	To -				
	id 🕡	creditloss1	i creditioss2	creditloss3	exprofit1	3	exprofit2	😟 exprofit3	0	decision	1	Offers
1	10001	125.8401543	9 62.920077194	4 106.96413123	-1.321316619		-1.035642274	0.1896791579		and the second second	1	2
2	10003	133.1366926	2 66.568346308	8 113.16618872	0.0243399037		-2.837544966	-0.136592398				3
3	10005	121.5274356	1 60.763717804	4 103.29832027	-0.11892676	5	-1.619434772	0.1264097883				3
4	10006	138.374709	4 69.187354701	1 117.61850299	-0.442538855		-0.850556301	-0.071936622		(2
5	10008	148.2354013	8 74,11770065	9 126.00009117	-0.51855832	2	-2.862581071	-0.079484402		(1	3
6	10009	127.0280283	63.514014163	7 107.97382408	-1.397183613	1	-1.848022056	0.1323977005		-	1	3
7	10011	143.8001052	4 71.900052622	2 122 23008946	0.1987336163		-1.79391068	-0.371282671				2
8	10014	140.6047036	9 70.302351841	7 119.51399814	-0.331607504		-0.448644439	0.0913721682			1	2
9	10019	123.7653582	61.882679133	7 105 20055453	0.1831902447		-1.037221612	-0.041343616				2
10	10020	144.4696828	8 72.234841442	2 122.79923045	-1.53616152	2	-1.328196204	-0.030192304		(2
11	10024	138.1994105	69.099705282	2 117.46949898	-1.208189544	1	0.4325687091	-0.354208986		1		2
12	10028	149.7092543	3 74.854627164	4 127.25286618	0.3239471761		-1,444832742	-0.494162423				2
13	10029	151.4261743	7 75.713087187	7 128.71224822	0.2400992435		-1.741863044	-0.461551107				2
14	10031	125.1238250	62.561912545	5 106.35525133	-1.206858013	1	-1.447645702	0.2776241932		1	1	3
15	10032	155.7995016	77.899750837	7 132.42957642	-0.091966956		-1.489458516	-0.464689215		(1	2
16	10033	144.6034873	3 72.301743664	4 122.91296423	-1.282231073	1	-1.688703064	-0.050889688		(1	2
17	10040	124.4512389	62.225619475	5 105.78355311	-0.30659943	8	-1.734772534	-0.057120758		(3
18	10041	138.8690773	9 69.434538694	4 118.03871578	-1.330455058	1	-0.801388672	0.1497782995			1	2
19	10043	126.7184660	63.359233038	8 107.71069616	0.61412182		-0.474446431	-0.388632109				2
20	10044	150.1453584	6 75.072679232	2 127.6235547	0.3524314827		-1.695022047	-0.35098978	3			2
21	10047	143.3880728	9 71.694036447	7 121.87986196	-0.254516106	1	0.3359695248	-0.483433503				2

Figure 13. View Optimization Output

MAKING IT ACCESSIBLE WITH SAS ENTERPRISE GUIDE OR MICROSOFT EXCEL

Users can access the optimization stored process with their choice of client. Because you have already tested it using SAS Enterprise Guide, you can see how it looks in Microsoft Excel.

In an Excel spreadsheet, select the $\ensuremath{\mathsf{SAS}}$ tab, and then select $\ensuremath{\mathsf{Reports}}$.

File	Hon	ne Ins	ert Page	Layout	Formulas	Data	Review	View	SAS
2			*	Ø	Modify Properties		1	2	
SAS Data	Tasks	Reports	SAS Favorites *	Refresh	E Propercies	Manage Content	Tools	Help	
		sert	5	S	election		Tools		

In the menu, select CreditOfferOptimization2.

owse Search SAS Fo	olders				
Look in:	My Folder		▼ (n ▼ 💽	× 🍃 🖬 • 65	
SAS Folders Servers Servers My Documents My Computer My Network Places	Stored Proc	ptimization2 ess for Progra			
	File name: Files of type:	CreditOfferOptin All File Types	iization2		
					pen Can

Figure 14. Select Stored Process

When you run the stored process, it executes just as it did in SAS Enterprise Guide.

A1	- (~	fu											
A	В	C D	E	F	G	н	1	J	К	L	M	N	C
		-											
		CreditOff	erOptimizati	on2								23	
		IT Change	ly required its		(herea)								
		I Show on	ity requires in	una (denotes	(by w)								
		General								В	east group de	efaults	
		Credit	Loss_Limit										
			um allowed or	redit loss per	offer.								
		15000	10										
		★ Mir	imum_Offer_	Level									
		200.0											
		_											
		-											

Figure 15. Run the Stored Process

The same results are displayed.

	Invoit Formular Data	David	View	CAC						Book1 -	Microsoft E	xcel		
Image Layout Formulas Data Review View SA3 Image Layout Formulas Data Review View SA3 A3 Image Layout Solution Summay Image Layout Solution Summay Image Layout N N N O A B C D E F G H I J K L M N O The OPTMODEL Procedure Image Layout Imag														
A	В	С	D	E	F	G	н	1	L	K	L	M	N	0
The OPTMODEL	Procedure													
			SAS Stat	us for Micro	soft Excel									• 💌
			🗎 55 m	X	2									
Problem Sur	Home Insert Page Layout Formulas Data 30 		Job		Status					Time Starte	ed 🗸	Result	s Location	
Objective Sense	Maximization													
Objective Function	Expected_Profit													
Objective Type	Linear													
			Total Jobs: () Comple	ted Jobs: 0	Running Jo	bs:0 Pe	ending Jobs:	0					1
Number of Variables	5136		1	1	1		1999) - 1993) I	-		-	-		-	
Bounded Above	0													
Bounded Below	0													
Bounded Below and Above	5136													
Free	0													
Fixed	0													
Binary	5136													
Integer	0													
Number of Constraints	1716													
Linear LE (<=)	1													
Linear EQ (=)	1712													
Linear GE (>=)	3													
Linear Range	0													
Constraint Coefficients	15408													

Figure 16. View the Results

PROJECT PLANNING WITH PROC OPTMODEL AND MICROSOFT PROJECT

Until now, this paper has focused exclusively on PROC OPTMODEL as an optimization engine under the hood of user-friendly stored processes. In addition, SAS/OR offers a robust suite of project planning capabilities.

- PROC CPM can be used for planning, controlling, and monitoring a project.
- PROC PM is another interactive procedure that can be used for planning, controlling, and monitoring a project.
- PROC DTREE is an interactive procedure for decision analysis.
- PROC GANNT is a graphical scheduling tool for planning and controlling a project.
- PROC NETDRAW is a procedure that draws a network diagram of the activities in a project.

 Earned Value Management (EVM) macros compare an earned value to the work that was planned to measure project performance and to estimate future costs and project completion.

These capabilities become ready for business when project data is read directly from Microsoft Project files using the %MSPTOSAS macro and processed in SAS/OR. Then, the results are converted into Microsoft Project files using the %SASTOMSP macro.

Here is a standard Microsoft Project plan for a market research project:

0	Task Name	Duration	Start +	Finish	29-03 05-04 12		03-05 10-05 17	21 May 7-05 24-05 31-05 07	11 June 06 14-06 21-06	01 July 28-06 05-07 12-07	21 July 19-07 26-07 02-0
	Primary Market Research Schedule	79.5 days	Mon 12-84-84	Fri 30-07-							Primary II
C	- Initiation Phase	13 days	Mon 12-04-04	Wed 28-04-			itiation Phase				
	Requirements Gathering & Analysis	4 days	Mon 12-04-04	Thu 15-04-		Requirements Gathering	& Analysis				
	Define process for gathering requirements	1 day	Mon 12-04-04	Mon 12-04-	CE-D	fine process for gathering	requirements				
	Document project-stakeholder interviews	1 day	Tue 13-04-04	Tue 13-04-	400	Document project-stakehol	der interviews				
	Analyze requirements					Analyze requirements					
	Create requirements document	1 day	Thu 15-04-04	Thu 15-04-	-	Create requirements do	cument				
0	Project Charter Development	10 days	Thu 15-04-04	Wed 28-04-			roject Charter Development				
-	Business Case	2 days	Thu 15-04-04	Fri 16-04-		Business Case					
	Document summary of business purpose justification	2 days	Thu 15-04-04	Fri 16-04-		Document summary of	business purpose justification				
	Define expected benefits	2 days	Thu 15-04-04	Fri 16-04-	*	Define expected bene	fts				
	Project Scope Definition (High Level)	5 days	Mon 19-04-04	Fri 23-04-		Project Se	ope Definition (High Level)				
	Define primary market research objectives	3 days	Mon 19-04-04	Wed 21-04-		Define primar	y market research objectives				
	Define primary market research deliverables	2 days	Thu 22-04-04	Fri 23-04-		40 O Define pre	mary market research deliverable	es .			
	Identify specific exclusions to scope	2 days	Thu 22-04-04	Fri 23-04-		HE Childentify sp	ecific exclusions to scope				
	Establish high-level time, cost, and resource estimates	2 days	Mon 26-04-04	Tue 27-04-			ablish high-level time, cost, and i	resource estimates			
	Establish list of key stakeholders	2 days	Mon 26-04-04	Tue 27-04-			ablish list of key stakeholders				
	Document project charter		Wed 28-04-04				ocument project charter				
	Project Charter Approval		*****				roject Charter Approval				
	Initiation Phase Approval	0 days	*****	****		40.0	itiation Phase Approval				
						<u>ت</u>					
	Planning Phase	32.5 days	Thu 29-04-04	Mon 14-06-		-			Planning Phase		
C	- Project Plan Development	32.5 days	Thu 29-04-04	Mon 14-06-		-			Project Plan Developm	ent	
	= Scope Statement	4 days	Thu 29-04-04	Tue 04-05-		-	Scope Statement				
	Create scope description (based on business objectives)	2 days	Thu 29-04-04	Fri 30-04-		+G 6	Create scope description (bas	ed on business objectives)		1	
	Define scope boundaries (both in and out of scope)	2 days	Thu 29-04-04	Fri 30-04-		×03 6	- Define scope boundaries (both	h n and out of scope)		1	
	Define key project deliverables (including user-acceptable criteria)	2 days	Mon 03-05-04	Tue 04-05-			Define key project deliv	erables (including user-acceptable c	iteria)		
	- Work Breakdown Structure (WBS)	4 days	Wed 05-05-04	Mon 10-05-			Work Breakd	own Structure (WBS)			
Ø	Create WBS to work-level execution	2 days	Wed 05-05-04	Thu 06-05-			Create WBS to work	k-evel execution			
	Define task dependencies (including predecessors/successors)	2 days	Fri 07-05-04	Mon 10-05-			Defne task d	ependencies (including predecessor	visuccessors)		
	Performance Baseline Measurement	2 days	Tue 11-05-04	Wed 12-05-			Carlo Performan	nde Baseline Measurement		1	
0	Create schedule baseline (with expected resource effort)	t day	Tue 11-05-04	Tue 11-05-			Create sch	equie baseline (with expected resou	de effort)		
Ø	Define budget baseline (with schedule and cost assumptions)	1 day	Wed 12-05-04	Wed 12-05-			HOD Define bu	dget baseline (with schedule and co	assumptions)		
	Establish baseline tolerance thresholds (trigger change control)	1 day	Wed 12-05-04	Wed 12-05-				baseline tolerance thresholds (trigge			
	Procurement Planning (Research Vendor Selection)		Wed 05-05-04					rement Planning (Research Vendor S	election)		
	Analyze market conditions	1 day	Wed 05-05-04	Wed 05-05-			Analyze market condi	tions			1
	Analyze make-or-buy findings		Thu 06-05-04				HO O-Analyze make-or-	bey findings			
	Select contract type	1 day	Mon 10-05-04	Mon 10-05-			Select contra				
	Develop procurement management plan							précurement management plan			
	Develop statement of work	2 days	Thu 13-05-04	Fri 14-05-				op statement of work.	t		1
	Project Schedule Plan	2.5 days	Thu 13-05-04	Mon 17-05-				roject Schedule Plan			
12	Project Milestones (PMs)	0.5 days	Thu 13-05-04	Thu 13-06-			Project M	filestones (PMs)			
	Define major milestones (alignment with deliverables)	0.5 days	Thu 13-05-04	Thu 13-05-			Define m	ajor milestones (alignment with delive	(ables)		
	Project Scheduling Work Plan	2 days	Thu 13-05-04	Mon 17-05-			Provide Provid	roject Scheduling Work Plan			1
	Define standard PM and approval gates	0.5 days	Thu 13-05-04	Thu 13-05-				tendard PM and approval gates			
	Document WBS, milestones, and deliverables (dates and resourcing)	2 days	Thu 13-05-04	Mon 17-05-			HO O DI	ocument WBS, milestones, and delive	tables (dates and resourcing)		
	Document schedule baseline	t day	Thu 13-05-04	Fri 14-05-				est schedule baseline	E		
	- Resource Management	2 days	Fri 14-05-04	Tue 18-05-				Resource Management			
	Name project learn members (with roles and responsibilities)	2 days	Fri 14-05-04	Tue 18-05-			40 0	Name project team members (with ro	es and responsibilities)		

An XML file of this project plan can be saved and then later identified in SAS with the following FILENAME statement:

filename check 'C:\Documents and Settings\sbxdemol\My Documents\Primary market research schedule 2007.xml' ;

This XML file can be read into SAS using the following simple macro call:

```
libname plan 'D:\Business Development\Events\2012\SAS Global Forum - Orlando\sasdata';
%MSPTOSAS (
    LIBRARY=plan,
    VERSION=2007,
    XMLFILE=check );
```

This creates the following data sets, which are already structured for use with the SAS/OR project planning capabilities:

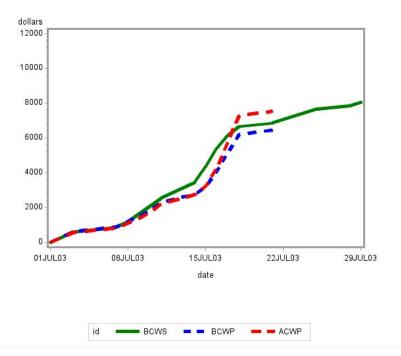
Contents of 'Plan'		
Name	Size	Туре
Activity	81.0KB	Table
Calendar	9.0KB	Table
📑 Holiday	5.0KB	Table
Prefs	33.0KB	Table
Resource	5.0KB	Table
Schedule	241.0KB	Table
Task_attributes	337.0KB	Table
📑 Workday	5.0KB	Table

The SAS interactive project management window immediately opens:

L on Example

Job	APTIMITY	ues:	28	MPRO	4 1	INPR	104	14	APRO	4 1	170	1104	20	MPRO	4 3	23AP	R04	26	APRO	04	290	1804		mato	4 1	1 In	104	014	INY04	-	1 March 1	04	148	1104	17	maro	4 2	OMAY	104	23110	AYO
ibr.	ACTIVITY	Code 0,0	1	F	3			1	1.1		8 P			- P	1	1	1	1	1		1.1	1	8	.C.,		1	t.,	3	3 /			-t		S	1	P	4.1		25	-	-
	Prinary Market Research Schedule	10.00	-				****				****				****																										
2	Initiation Phase	0.0.0		· · · · ·				-						-}					·} · · ·		}		į	· · · · · ·								·	·}								
	Requirements Gathering & Analysis	0.0.0.0		ļ;			-2							ş	ļ				ļ			· · · · ·						ļ;			4.,	ą.,,			4		here in				
00004	Define process for gathering requires	0.0.0.0		1			1	i	1.4				4	4					1				.i.,					į			4.	4			4.				1		
00005	Document project-stakeholder intervie	0.0.0.1		11	_		٠.	I						.i									.i.,					1			. i	.i									
00006	Analyze requirements	0.0.0.2		1			1				!			.i						l			.i.,					1				.i			1						
20007	Create requirements document	0.0.0.0.3		1			1						4.						1				1.	1.1				į		1		J	ļ;		1.			4			
	Project Charter Development	0.0.0.1		ļļ	L.			÷						.,														į				.i	····				in the second				
	Business Case	0.0.0.1.0		11		1.	1	1	_														1.					1				1			1.						
00000	Document sunnary of business purpose	0.0.0.1.0.0				1		1						1			1		1	0							1					1			1			1			
	Define expected benefits	0.0.0.1.0.1					1	1		h			1.	1.		. 1			1			1	1	1 1	1			1		1	1	1			1	1					
	Project Scope Definition (High Level)	0.0.0.1.1					-	-	10								1	1	1				1				-								-			120			
	Define primary market research object	0.0.0.1.1.0						-		Ļ	-	-1		h					1				1	1						1								1			
00000	Define primary market research delive	0.0.0.1.1.1					1	1						1					1				1				1			1	1	1			1						
	Identify specific exclusions to scope	0.0.0.1.1.2					1	1	1					1 L		h			1				1	1		1	1			1	1	1			1	1				1	
100016	Establish high-level time, cost, and	9.0.0.1.2	1				1	1						-		-		-					1			1	1					1						1			
100017	Establish list of key stakeholders	0.0.0.1.3	11	11		1	1	1	11		1	1	1	1		÷	-	-	h	1			1	1 1	1	T	1	1		1	1	1		1	1	11		1	1	1	
100018	Document project charter	0.0.0.1.4		11			1	1	1	1	1	1	1	1		1	1	-	ť	h		1	1	11	1	1	1			T	1	1			1	1	1	1	1	1	
100019	Project Charter Approval	0.0.0.1.5	11			1	1	T	1				1	1			1	1	1	The second		1	1.	11			1	1			1	T			1	1			1	1	
20020	Initiation Phase Approval	0.0.0.2		1-1			1	1	1							1			1	++-				11			1	1		-	1	1				1			1		
21		0.1		1.1		1		?	1		}		17	-}		-		1						· · · · ·			·.	<u></u>	····)·		1	·!···	· · · · · ·								
22	Planning Phase	0.0.1	1	11		T	1	1	11		-1	1	17	1		-1	-1-	1	1.	1			-24+	-22				2					1	1112					-22		
	Project Plan Development	0.0.1.0	1	11	1	111	1	1	11					1			···?··	12		1	301			(\$2.2.2)	10.4							-									
	Scope Statement	0.0.1.0.0		1-1			+	÷	1				1	1					+						h		1	1		1		1			1	1	1		1		
	Create scope description (based on bu	0.0.1.0.0.0	-	1-1			1	i	111			÷	1.	1		-i	÷		t***	T	-	1	1.		1		1	1				1-			1-			-1-	1	***	
	Define scope boundaries (both in and	0.0.1.0.0.1		1-1			****		1										1	C		-	· · · ·	1-1		-	1			t	+	·				+	11.1.1				
	Define key project deliverables (incl	0.0.1.0.0.2	-	++				1	1										+	t		-	-1	1			-+	· ····		-+-	-+	+	· \$			+			-11		
and the second second	Hork Breakdown Structure (HDS)	0.0.1.0.1	-	·····			-?	·?···	1			···	·····	·!···					·?···				·?··				*****	· · · ·				· ····		···?··	·····						
	Create HBS to work-level execution	0.0.1.0.1.0	-	1			-						-										····		-t			1			******	1-				-					
TIT	Define task dependencies (including p	0.0.1.0.1.1	-				+	÷					÷	÷					÷				÷				£	å	i.			÷			+	+					
31		0.0.1.0.2		++				· ····						+					÷				····				-	1.1		-		+			+						
10000		0.0.1.0.2.0	-	·				÷					4										.ş							1	÷.,	4			4						***
	Create schedule baseline (with expect		+		}		÷	· ····					· · · ·	· · · · ·				1					.j	·			·····	j		-		· • • • •	1 2		·				-1		
00000	Define budget baseline (with schedule	0.0.1.0.2.1	-	·····			· ····	·						· ····					· ····					· · · · ·				· ····			Į.	· ····							· · · · ·		
00000	Establish baseline tolerance threshol		-					Ş	ļ										.;									ł				l									
	Procurement Planning (Research Vendor	0.0.1.0.3	4	1			1	i	1				1						1				.i.,																		
	Analuze narket conditions	001030		1 1	10.4	1	1	1	: 1	1	- 1	1	1	: :	1	1		1	1		-22	1.18	2.1	2 1		L 1	2.1	2 3	1	1	1	11		1	1	1 1	1	1	1 1	1	1

You can now run various analyses. For example, here are the results of a cost analyses:



Comparison of Costs

After you have analyzed the project and modified the project flow if needed, you can complete the task by creating an .mdb file for further use in Microsoft Project.

filename mdbfile 'C:\Documents and Settings\sbxdemol\My Documents\revised\Revised market research schedule 2007.mdb' ;

```
%SASTOMSP(
    LIBRARY=plan,
    MDBFILE=mdbfile,
    _DUR=duration,
    actds=activity, calds=calendar, workds=workday,
    scheduleds=schedule, interval=dtday,
    _date="12APR04:08:00:00"dt,
    _activity=ACTUID, _successor=SUCCUID,
    _lag=LAG, _project=PNTUID,
    _id=ACTIVITY ACTUID
    );
```

CONCLUSION

For SAS/OR and the optimization possibilities that it offers to reach a business, analysts need to be creative in making SAS/OR more business-ready for business users. This paper discusses a few simple approaches toward this goal.

CONTACT INFORMATION

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

Andrew Pease SAS Institute Inc. E-mail: Andrew.Pease@sas.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.