

PharmaSUG 2021 – Paper SI-019

Advanced Project Management beyond Microsoft Project, Using PROC CPM, PROC GANTT, and Advanced Graphics

Stephen Sloan, Accenture; Lindsey Puryear, SAS Institute

ABSTRACT

The Challenge: Instead of managing a single project, we had to craft a solution that would manage hundreds of higher- and lower-priority projects, taking place in different locations and different parts of a large organization, all competing for common pools of resources.

Our Solution: Develop a Project Optimizer tool using the CPM procedure to schedule the projects and using the GANTT procedure to display the resulting schedule. The Project Optimizer harnesses the power of the delay analysis feature of PROC CPM and its coordination with PROC GANTT to resolve resource conflicts, improve throughput, clearly illustrate results and improvements, and more efficiently take advantage of available people and equipment.

INTRODUCTION

In this paper, we outline how we used the SAS/OR ® CPM, SAS/OR ® GANTT and PRINT procedures to create a Project Optimizer that allows projects sharing a single resource pool to be scheduled. Before delving into the specifics of the project, a brief introduction to the tools we used is in order. The CPM procedure is a project management tool based on the Critical Path Method of scheduling. Activities are assigned times based on their precedence relationships, starting with activities having no predecessor. The critical path is determined by making a backward pass from the end of the project to the beginning. A delay in a so-called critical activity will extend the project end date. PROC CPM is able to accommodate work shifts and holiday calendars and can schedule activities constrained by shared resources. This procedure uses a heuristic to find a good, feasible solution, so there is no guarantee that the span of resulting project schedules is at a minimum.

The Gantt procedure works seamlessly with PROC CPM to aid in visualizing the schedules produced. PROC GANTT can compress larger charts both horizontally and vertically to produce the desired number of output pages. The SAS Output Delivery System (ODS) supports both Portable Document Format (PDF) and Hypertext Markup Language (HTML), among other formats. PROC GANTT has extensive annotation capabilities for labeling charts. PROC GANTT can use a detailed annotation data set to draw primitives like text and lines directly. Alternatively, a label data set can similarly apply text to every activity. In addition, there are options that control precedence line drawing. PROC GANTT uses the same holiday data set provided to PROC CPM to differentiate holidays visually. Now we describe the project we tackled.

CHALLENGE

We tackled a scheduling task in which large numbers of otherwise unrelated projects shared common resource pools. Each project was previously managed via Microsoft Excel ® or Microsoft Project ®, and resources were assumed unlimited. Our mission was to help our users create realistic project schedules and to get an overall handle on the number of projects, the priorities of the projects, the deadlines, and the available resources. This was performed at both the division and corporate levels.

However, before using the CPM procedure to schedule the projects, we realized that we needed a global view of the human resource capabilities within the organizations. To this end, we obtained a list of projects under consideration over the next three years. We worked with the users to learn priorities, skills classifications and resource requirements. We grouped the resources into seven categories, and obtained availability estimates. Our first output was a graph showing workload against availability over time. The result for one of the resource categories appears in Figure 1 below.

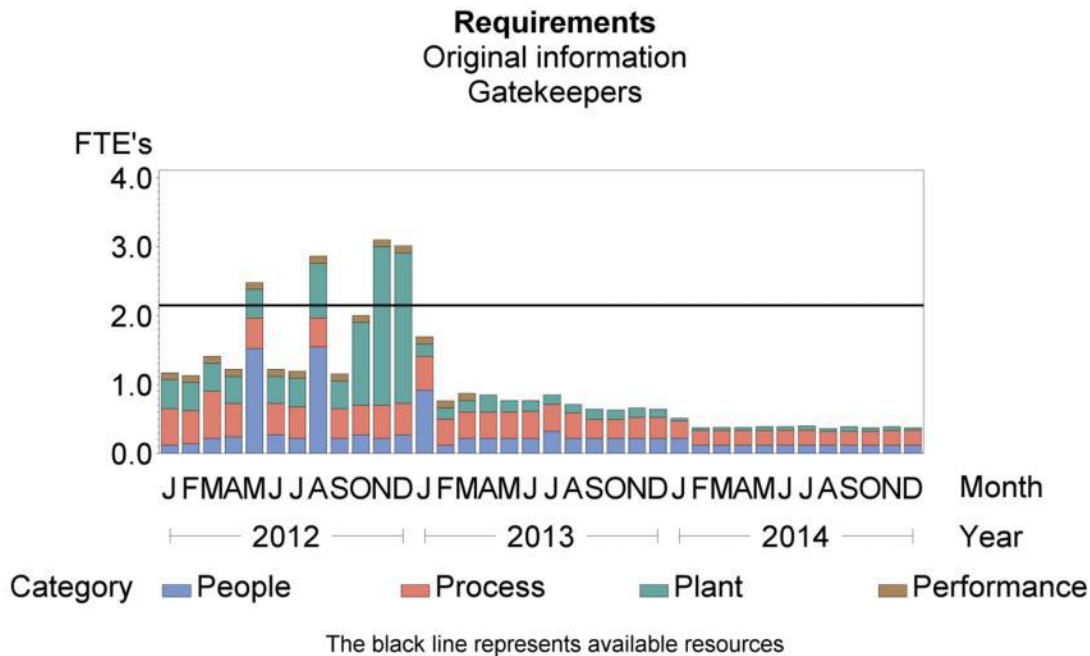


Figure 1. Resource Usage and Availability Level

FTE is Full-Time Equivalent, a normalized measure of one full-time employee's work. Please note that there is a resource capacity shortfall during some time periods, while at other times there is a surplus of resources.

SOLUTION

PROJECT OPTIMIZER

We developed a Project Optimizer tool using the CPM procedure for scheduling and the Gantt procedure for displaying the results. The Project Optimizer enables a user to resolve resource conflicts, improve throughput, and more efficiently take advantage of available people and equipment. The following describes the information accepted by the tool:

- Precedence relationships between tasks and projects
- Start and due dates of the projects (using alignment constraints, with the caveat that constrained resources can override deadlines)
- Resource availability (resource profile defined by the quantity available at the times(s) given)
- Resource driven durations (enable the user to add resources along the resource-constrained critical path for project crashing)
- Resource delay analysis (enables the user to identify and reduce or eliminate delays due to insufficient resources)

- Resource type (consumable or replenishable)
- Resource quantity required by task
- Costs associated with executing tasks (costs can be aggregated as a consumable resource)
- Activity splitting (maximum number or minimum duration of segments can be specified for each activity)
- Activity priority (can be specified for tie breaking)

The Project Optimizer can provide reports and Gantt charts for the resulting schedule, along with graphs and other visual aids. The tool follows the logic below:

1. At a minimum, the tool requires at least two tasks with durations, a start date for the project, and optionally precedence relationships between the tasks (i.e. which tasks need other tasks to complete before they can execute.)
2. The tool will start tasks subject to precedence constraints; i.e. a successor task will only be available to start after its predecessor tasks have finished.
3. In the presence of resource requirements and constraints, the tool will delay the start of tasks until resources are available and precedence constraints have been satisfied.
4. If a finish date is assigned for the project, and the finish date constraint cannot be satisfied, we have three choices for the tool:
 - a. Assign more resources than are currently available.
 - b. Allow the project to go past the defined end date.
 - c. Give a message and do not produce a schedule.

The tool also has the option of allowing activity splitting. It can schedule a minimum amount of time for someone to work on a task, and allow for interruptions after this minimum amount of time has elapsed, but before the end of the task. This will allow available resources to work on tasks along the resource critical path. The tool can also accommodate activity priorities. Higher-priority tasks will take precedence over lower-priority tasks, and can interrupt them with the activity splitting feature in effect.

CPM PROCEDURE

The CPM procedure is the scheduling engine used by the Project Optimizer. We chose the CPM procedure to more efficiently assign resources when scheduling the projects. While there is no guarantee of optimality, this procedure offers a multitude of options and features that used to guide the scheduling of the projects. In the case of resources, the CPM procedure allows the user to specify both the resource level profile over time, as well as the type (consumable or replenishable). The CPM procedure also has capabilities that will be useful in future stages of the implementation:

- The ability to associate a cost resource with tasks (this non-constraining resource type is consumable)
- The ability to measure progress against a base line schedule, and show the financial and chronological impact on the base line of different schedule updates

When identifying resources, PROC CPM allows the user to specify the available amount of different types of resources, and when they are available. Resource type is indicated in the resource data set by the OBSTYPE= keyword. A brief summary of some of the pertinent values for OBSTYPE= follows:

- RESTYPE allows the user to indicate whether a given resource is consumable (like raw materials), replenishable (like human resources) or consumable aggregate (costs)
- ALTPRTY/ALTRATE is used to define resource pools
- SUPLEVEL specifies supplemental resources to be used only when an activity stands to be delayed beyond the amount given by DELAY= and ACTDELAY=. The INFEASDIAGNOSTIC keyword allows infinite supplementary resources.
- AUXRES enables auxiliary resource relationships. An auxiliary resource is one that is required whenever the associated resource is used. A crane operator is usually an auxiliary resource for a crane.
- RESRCDUR specifies resource driven durations. This duration derives from the amount of work for the task and the work rate of the resource used for the task.

In the event of insufficient resources, the Project Optimizer tool lets the user determine whether lack of resource availability will be met by adding resources, extending due dates, or interrupting the process with an error message. At the same time, we can supply PROC CPM with information about the tasks in each project, the earliest times that they can start, the latest times that they can finish, task and project priority, and precedence relationships between tasks within projects. In the presence of resource requirements and constraints, the tool delays tasks until both resources are available and priority and precedence constraints are satisfied.

If necessary, PROC CPM can schedule at the shift level, and can schedule around holidays. For our purposes, start and end date were sufficient, as we were looking at monthly resource availability figures.

Finally, PROC CPM allows users to split activities so that there are idle gaps during the execution of the tasks. We can specify on a task-by-task basis which activities are subject to splitting. For a given activity, PROC CPM supports two ways of controlling the splitting: you must specify either a maximum number of segments, or a minimum duration for the segments. Activity splitting can facilitate more complete usage of available resources. This is possible either by interrupting lower priority activities or activities not on the critical path.

The CPM command we used appears below:

```
proc cpm data=resource_requirements /* Activity information */
  interval=month /* Time units for activity durations */
  date="&Project_StartDate."d /* align schedule to start date */
  out=schedule /* output schedule data set */
  resin=resource_availability_update /* resource availability */
  resout=resource_usage /* usage of each resource */
  resourcesched=resource_schedule /* when each resource is used */
  suppressobswarn; /* suppress missing value warnings */
  activity initiative; /* Identify activities to be scheduled */
  /* durations are resource-driven so no durations are provided */
  successor successor1-successor8; /* precedence relationships */
  /* control the scheduling of some tasks by manually */
  /* specifying alignment information */
  aligndate aligndate;
  aligntype aligntype;
  /* specify resources and options */
  resource Change_Leaders Gatekeepers Shift_Supervisors Ops_Chiefs
  Ops_SME Environmental_SME PS_SME Other /
  maxobs=99999 period=per routinterval=day
```

```

obstype=otype /* Identifies type of resources */
work=usage /* Work in standard units like FTE */
/* infeasdiagnostic enables unlimited resources */
/* infeasdiagnostic delay=0 */
rcs /* give rate of resource usage */
avl /* give rate of resource availability */
totusage /* give total usage of resources */
append /* append total usage to rate of usage in output */
delayanalysis /* request delay analysis to show bottlenecks*/
minsegmtdur=minseg /* enable activity splitting */
schedrule=actprty /* activity priority tie-breaking */
actprty=newpriority; /* specify activity priority */
run;

```

Using PROC CPM, we were able to schedule the projects within resource limitations, though the time span of the multi-project was extended by 6 months, as shown in Figure 2 below:

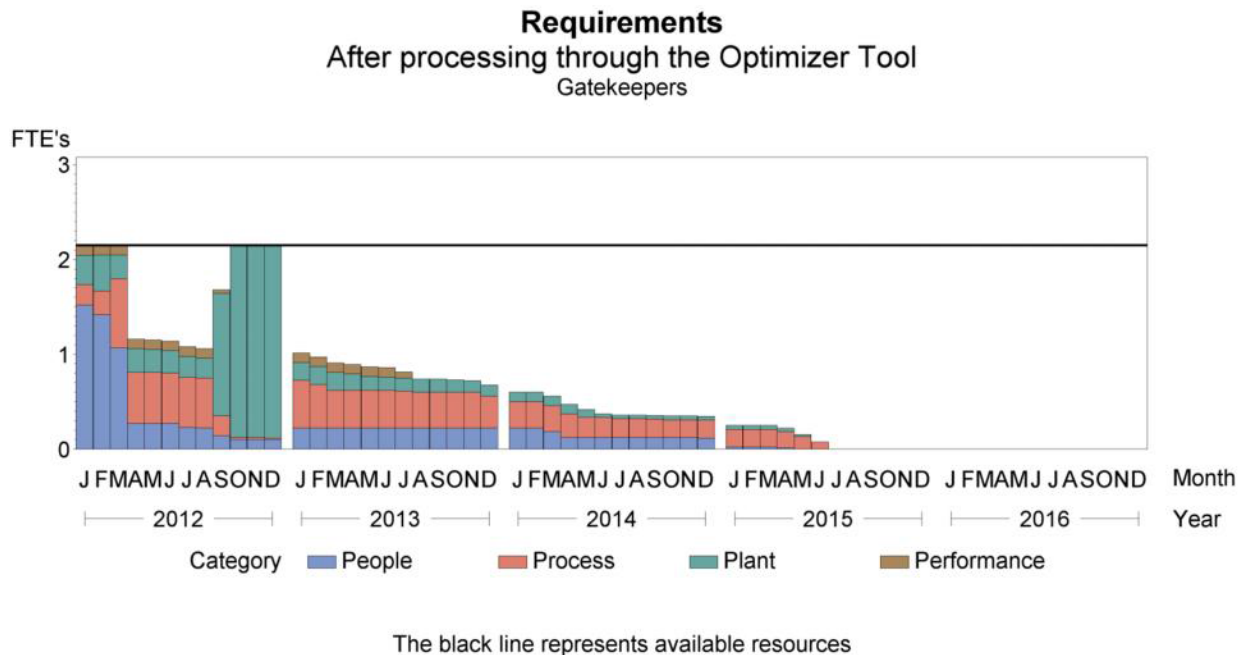


Figure 2. Improved Resource Usage with Availability Level

GANTT PROCEDURE

We used PROC GANTT to display the schedule. As we allowed task splitting in order to avoid idle resources, the task segments are labeled "Part 1", "Part 2", etc. PROC CPM supplies a reserved segment number with the variable name segmt_no. This variable was not included in the Gantt chart so that it could focus on the categories of concern to the user: People, Process, Plant, and Equipment. For the same reason, the placement of the project and task number, the task name, and the resource name take advantage of the annotation capabilities and appear above the bars instead of in the column to the right.

PROC GANTT provides a wide array of options in defining charts. It allows the user to specify the time periods, the intervals (we chose quarters, and PROC GANTT used the FORMAT we defined), the reference lines, and the colors and size of the charts. We used a parameter file as part of the input to specify the different graphical details on the Gantt chart. The PROC GANTT command appears below:

```

proc gantt data=gantt_data(where=(e_start^=.)
  rename=(start_date=e_start end_date=e_finish))
  labdata=label_E; /* labdata data set controls annotation */
  chart
    e_start=' ' e_finish=' ' /* plot e_start and e_finish */
  /
  mininterval=qtr /* time interval for labeling */
  nojobnum nolegend mindate="&Project_StartDate"d
  maxdate="&Project_EndDate"d
  /* specify key for linking label and activity data sets */
  labvar=initiative
  labsplit='|' /* specify split character */
  scale=10 /* adjust the horizontal scale of the chart */
  skip=4 /* add vertical space between bars */
  barht=.1 /* specify bar height */
  chcon=black /* specify black for horizontal connector color */
  height=1.5 /* adjust text height */
  /* collate tasks by zone */
  zone=Category onezoneeval zoneoff=-1 czone=black
  /* specify reference lines */
  ref="&quarter_graph_start"d to "&quarter_graph_end"d by qtr
  cref=black lref=1 lwidth=1 idpages
  useformat; /* apply proc format to time axis */
run;

```

Figure 3 below shows the graphic produced by the code above:

Gantt chart after processing through the Optimizer tool

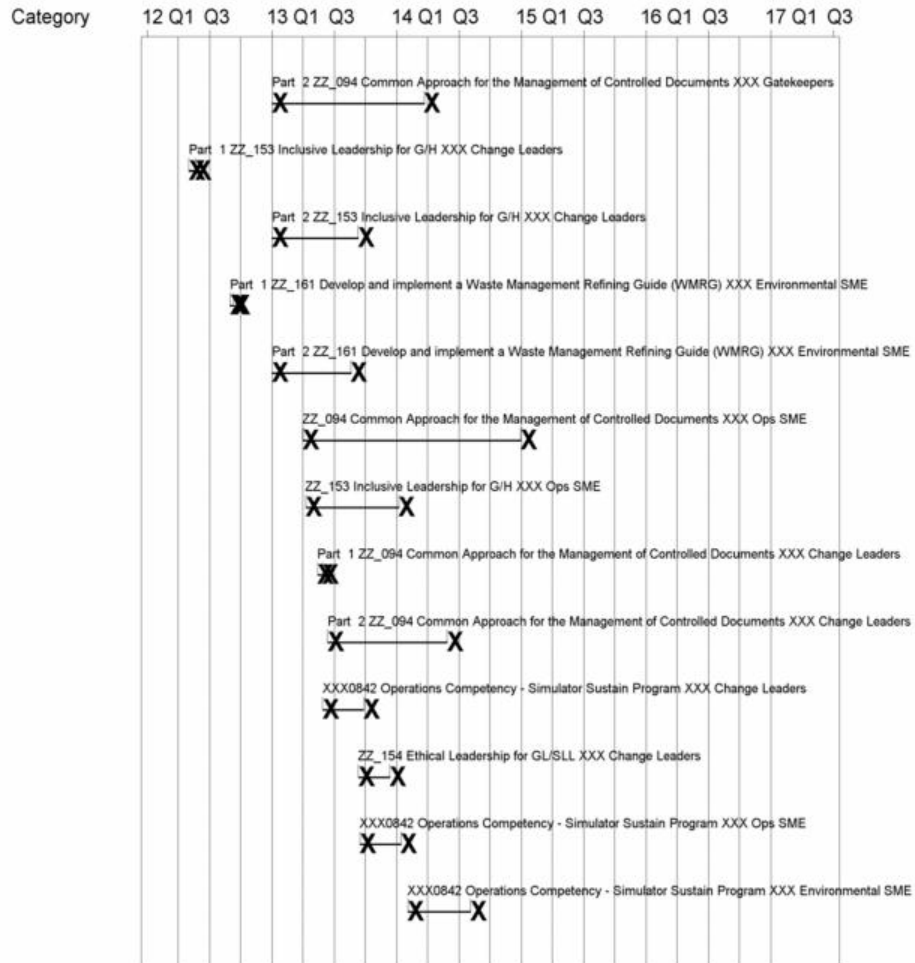


Figure 3. Improved Schedule with Gantt

In order to provide the users with a summary of the changes suggested by PROC CPM, we used PROC PRINT and ODS to produce spreadsheets comparing the start and end dates before and after PROC CPM was run. We generated the spreadsheets at the overall activity level and at the activity-resource level. Finally, we set up a front-end to the system using Microsoft Excel and Microsoft Visual Basic. The input screen allows the user to point to spreadsheets that contain the resource requirements, the resource availability, and the options chosen by the user. Display 1 below shows the input screen:

Scheduling Tool						
<p>INPUTS</p> <p>INPUT (RELATIVE PATH) <input type="text" value="\Input"/></p>	<p>INSTRUCTIONS</p> <ol style="list-style-type: none"> 1. Update the relative path of input file and output files, if necessary. 2. Download all the necessary input files to the "Input" folder. 3. Run the tool, as instructed stepwise 4. See output. 					
<p>ACTIONS</p> <p>STEP-1: DOWNLOAD INPUT FILES & SAVE TO SPECIFIED DIRECTORIES</p> <p>STEP-2: SPECIFY THE UNIT OF INTERVAL <input type="text" value="MONTH"/></p> <p>STEP-3: SPECIFY PROJECT START DATE</p> <table border="1"> <tr> <td>1</td> <td>JAN</td> <td>2012</td> </tr> </table> <p>STEP-4: SPECIFY PROJECT COMPLETION DATE (Optional)</p> <table border="1"> <tr> <td>31</td> <td>DEC</td> <td>2014</td> </tr> </table> <p>STEP-5: COPY PARAMETERS & DELETE EXISTING OUTPUT FILES <input type="button" value="↻"/></p> <p>STEP-6: GENERATE OUTPUT <input type="button" value="➡"/></p> <hr/> <p>If the project can not be completed with the available resources by project completion date, shall we increase the <input type="button" value="Extend the completion date"/></p>		1	JAN	2012	31	DEC
1	JAN	2012				
31	DEC	2014				
<p>OUTPUT</p> <p style="text-align: center;">PICTORIAL REPRESENTATION</p>						

Display 1. Project Optimizer Input Screen

FUTURE WORK

While the Project Optimizer tool will likely provide the best results through manual trial and error, it is possible to automate the process of adding resources to achieve a given end date by performing a delay analysis. To this end, we use two features of the CPM procedure: the DELAYANALYSIS option and its ability to identify the critical path. A description of the process follows.

First, you can invoke the CPM procedure with the DELAYANALYSIS option of the RESOURCE statement. The procedure adds two variables to the output schedule data set: DELAY_R, which gives the duration of any activity delay, and R_DELAY, which gives the delaying resource. Next, you can identify activities that caused delays as those immediately preceding delayed activities that required the same resource. Finally, you can create an artificial precedence relationship between each of these pairs of activities. Using the same data set and variable names as before, the following SQL creates a table with the new precedence relationships:

```
proc create table precedence as
  select a.Process1, a.Process2, a.Process3, a.Process4, a.Process5,
         a.Process6, a.Process7, a.Process8, a.Process9,
         b.initiative, a.initiative as successor1, '' as successor2,
         '' as successor3, '' as successor4, '' as successor5,
         '' as successor6, '' as successor7, '' as successor8,
         b.duration as duration
  from resource_schedule b, schedule a
  where a.r_delay > 0 and a.delay_r = b.resource
        and b.dur_type='RDRIVEN' and b.s_finish=a.s_start;
```

Updating the durations in the input data set with the corresponding resource driven values from the schedule, and appending the observations as created above, you can invoke PROC CPM without resources. Activities that are on the so-called resource critical path will have a T_FLOAT value of zero, meaning they cannot float further along in time without affecting the multi-project finish date.

Next, you can choose the resource whose available quantity is to be incremented. The first choice is the resource corresponding to the critical activity with the largest resource delay. Assuming OUT is the name of the output schedule obtained by invoking PROC CPM with no resources, the following SQL will assign the resource name associated with the maximum delay to macro variable RESOURCE:

```
select max(b.r_delay) into :maxdly from out a left join schedule b
  on a.initiative=b.initiative
  where a.t_float=0 and b.r_delay>0;
select b.delay_r into :resource from out a left join schedule b
  on a.initiative=b.initiative
  where a.t_float=0 and b.r_delay=max(&maxdly.,1);
```

If no critical activities were delayed, then we choose the resource that will shorten the resource critical path the most. Consider that the activities in this case have resource driven duration given by: $(duration / resource_quantity)$. Incrementing the resource quantity by one unit gives this change in resource driven duration: $(duration/resource_quantity) - (duration/(resource_quantity + 1)) = duration/(resource_quantity * (resource_quantity + 1))$.

We want to find the resource that maximizes this expression and increment its quantity by one unit. (Note that duration in this case is the total duration of the activities on the resource critical path that require the same resource. Also, note that this is only a heuristic because other bottleneck resources can be uncovered at each step, limiting the calculated speedup.) The following SAS code will assign the chosen resource name to macro variable name RESOURCE as above:

```
proc sql noprint;
  select sum(a.duration), b.resource into :sum_1-:sum_9, :res_1-:res_9
    from out a left join resource_schedule b
    on a.initiative=b.initiative
    where a.t_float=0 and b.resource is not null
    group by resource;
quit;
data _null_;
  retain total decrement 0;
  set resource_availability_update;
  if otype='reslevel' then do;
    %do i = 1 %to 7;
      %if %symexist(sum_&i.) %then %do;
        /* determine effect of increasing resources */
        decrement = &&sum_&i../(&&res_&i. * (&&res_&i. + 1));
        if (decrement > total) then do;
          total = decrement;
          call symput('resource', "&&res_&i..");
        end;
      %end;
    %end;
  end;
run;
```

You can repeat the above steps until you arrive at the desired makespan or resources limit further progress.

CONCLUSION

We have demonstrated how the Project Optimizer uses the CPM and Gantt procedures to schedule a multi-project and display the corresponding Gantt chart. It also allowed for a graphical representation of the “before and after” progress. By making adjustments through the Project Optimizer, we were able to bring the resource usage down to acceptable limits. This did however extend the schedule an additional six months.

REFERENCES

SAS Institute Inc. 2011. *SAS/OR 9.3 User’s Guide: Project Management*. Cary, NC, USA: SAS Institute Inc.

CONTACT INFORMATION

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

Stephen Sloan
Accenture
917-375-2937
Stephen.B.Sloan@accenture.com

Lindsey Puryear
SAS Institute Inc.
Lindsey.Puryear@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.