Schedule Your SAS® Jobs & Go WithThe Flow

Laura Liotus, Community Care Behavioral Health, Pittsburgh, Pennsylvania

ABSTRACT

Creating a SAS job flow with Platform Suite for SAS® in a Microsoft Windows environment, allows one to run processes unattended. SAS jobs are created and deployed in SAS® Data Integration Studio. Job flows are designed and scheduled in SAS® Management Console. These flows are run using Platform Suite for SAS which includes Platform Process Manager and LSF. The flows are created with dependencies that include a job completion status, file arrival or file existence. With a thoughtful plan, the user can get creative with the dependencies in SAS Management Console and the flow capabilities of Platform Process Manager. An added feature of all of the tools includes job status notification. With proper design and planning, one can use technology to automate their SAS processes.

INTRODUCTION

While using existing functionality in the SAS tools, with thoughtful planning and configuration, a system administrator or programmer can schedule SAS job flows to run in a smooth continuum. This automation allows for reduced monitoring and in the event of a process failure, an easier recovery. If jobs are resource intensive and have to be run during non-working hours, intelligent flow set-up allows for the user to be less attentive while the jobs run. With a segmented flow, if a system interruption or program error causes a job to fail, the process can be restarted from the point of failure. The SAS user can let technology do its job so that he or she can focus on larger tasks or simply enjoy their time away from work. There are multiple ways to monitor process flows including a graphical user interface of the flow progress, alerts and SAS job logs produced as the jobs run. An understanding of the sources and targets also aides one in job processing monitoring.

APPLICATION SET-UP FOR THE FLOW

There are three tools needed to configure your SAS jobs in a flow. The products needed for the examples and descriptions covered in this paper are SAS Data Integration Studio for job deployment and the SAS Management Console scheduling wizard. To take advantage of SAS scheduling, a third party product, Platform Suite for SAS will need to be included in your SAS contract. If Platform Suite is part of your contract, Process Manager and Platform LSF will be included in your SAS software deployment. The prerequisites for running the Platform Suite are in the installation manuals. The Windows system administrator at your site can configure the proper accounts and permissions in the operating system to allow SAS Data Integration Studio and Management Console to interface with Platform Suite.

A minor configuration is needed in SAS Management Console to assist with the job flow monitoring. Go to the Options tab of the SAS Application Batch Server properties. Click Advanced Options to edit The Rolling Log Options. Edit the text box to include year, month, day, minutes and seconds, Figure 1. This will append a date and time stamp to the log file name. Having the date and time included in the log file name prevents logs from being overwritten which can be useful for monitoring, identification and historical purposes.
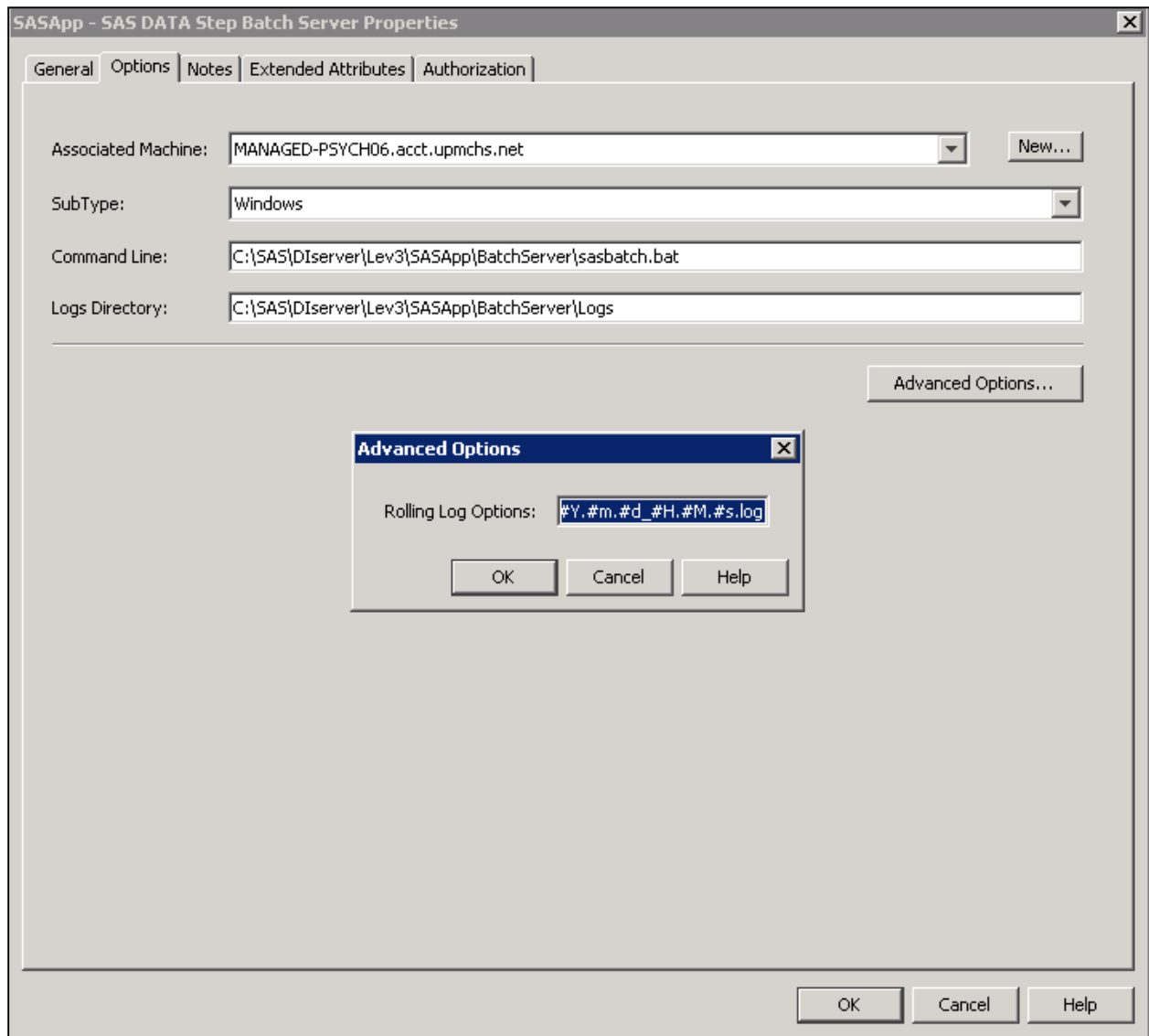
.

Figure 1. Add Date and Time to Log File Name

PLAN THE FLOW

Once your SAS server is configured with Platform Process Suite and the log file names are set-up to include a date/time stamp, the next step to any job flow is determining the order of the jobs to be processed. This entails understanding source readiness and data dependencies. One needs to ensure that the most current data is being used in the SAS Data Integration job. Some job targets are sources for future jobs. Determining dependencies will require working with database administrators, data managers, analysts and programmers. You will need to communicate with your colleagues that understand what sources are needed and the various stages of the data. Your site will also need to determine how jobs should be segmented between flows. Perhaps all jobs will simply be in one flow or possibly segmented based on data definition or discipline. An understanding of the data and what jobs depend on predecessors gets you ready to create your flow that includes dependencies or triggers.

CREATE THE FLOW

Using the environment at Community Care Behavioral Health as an example, the first SAS Data Integration Studio job in the flow depends on a Microsoft SQL Server database refresh. The SQL Server job contains a step that produces a file on the SAS server. In our Windows environment, the syntax below is a step in the Microsoft SQL Server SQL Agent job. This produces the text file for the SAS server.

time /t > \\SAS_Server\Folder\SourceDataBaseReady.txt

As Platform Process Manager allows for a file arrival trigger, the final step of the source database process is the creation of this simple text file indicating the database is ready. This will be referred to as an 'end' file. In a Windows environment, to create the file is as simple as this one line of DOS code above. This gets the entire process flowing.

Upon the arrival of the text file generated from the database job, the entire SAS job flow begins. Using this same concept, SAS jobs can be grouped in streams that start with the arrival of a text file. The last job of each logically segmented stream can contain code to generate end files. The syntax from above can be placed in a simple DOS batch file. The call to the DOS batch file can be included in the SAS Data Integration Studio job in the Postcode section of the Precode and Postcode tab, Figure 2. Or this syntax can be called directly from the job postcode without a batch file. These commands, using either a batch file or issued directly, can be added to multiple SAS jobs as postcode processes. These commands will create the end files to be used as triggers for the next flow. Subsequent flows can be run based on the arrival or existence of these files. In the same manner, to clean up the files for the next run, the end files are deleted as a postcode entry of the final job in the triggered flow segment. Again, a batch file containing the delete file syntax can be included or the DOS syntax can be directly called from the postcode of the job.
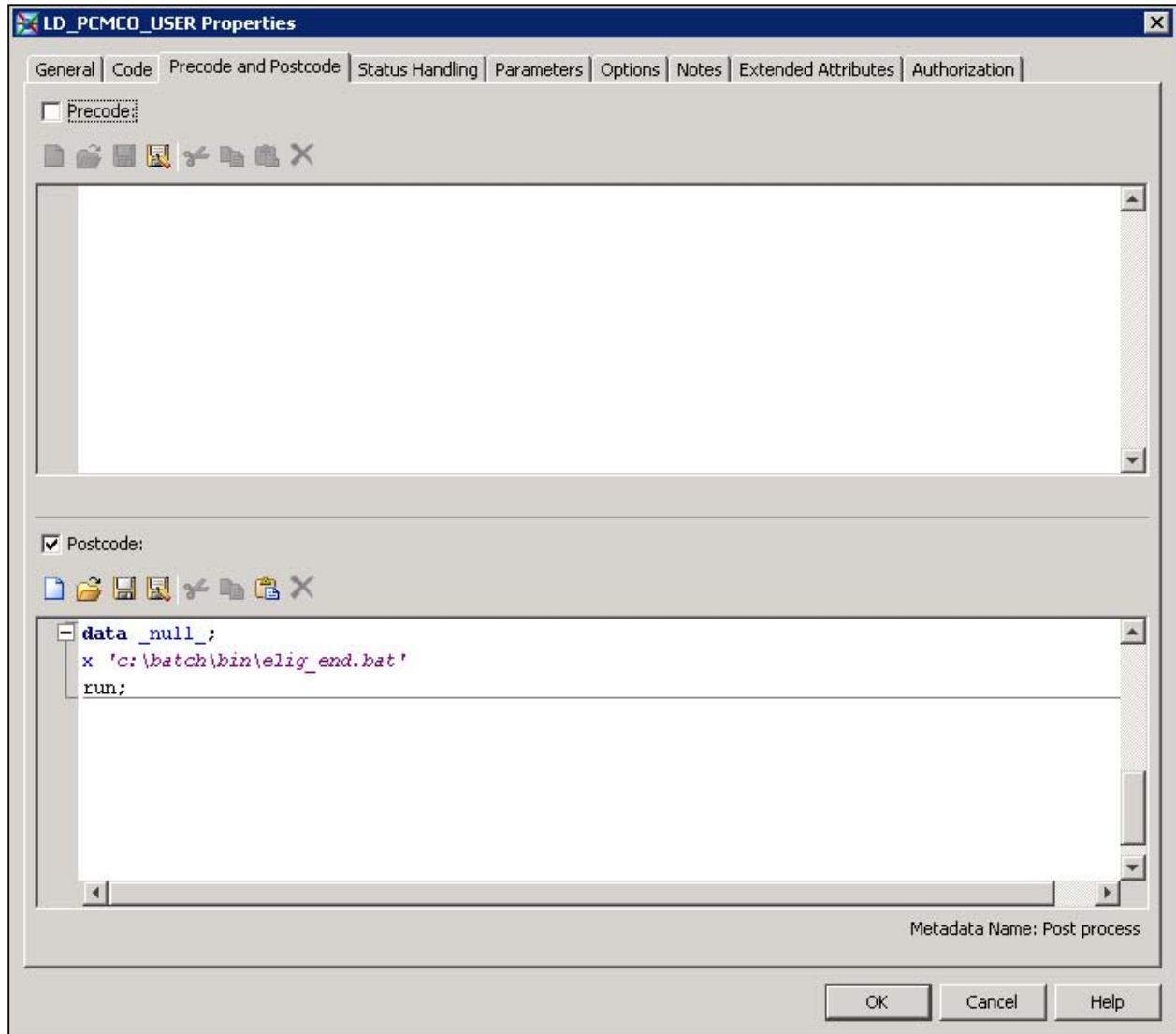
Figure 2. Call Batch File to Produce Text File in Postcode

IMPLEMENT THE FLOW

Implementing job flow in SAS involves a few simple pieces besides the SAS tools, including DOS batch files or calling operating system commands from a SAS Data Integration Studio job. Jobs are created and deployed in SAS Data Integration Studio. Once the jobs include the postcode commands, the next step is to deploy the jobs, Figure 3.
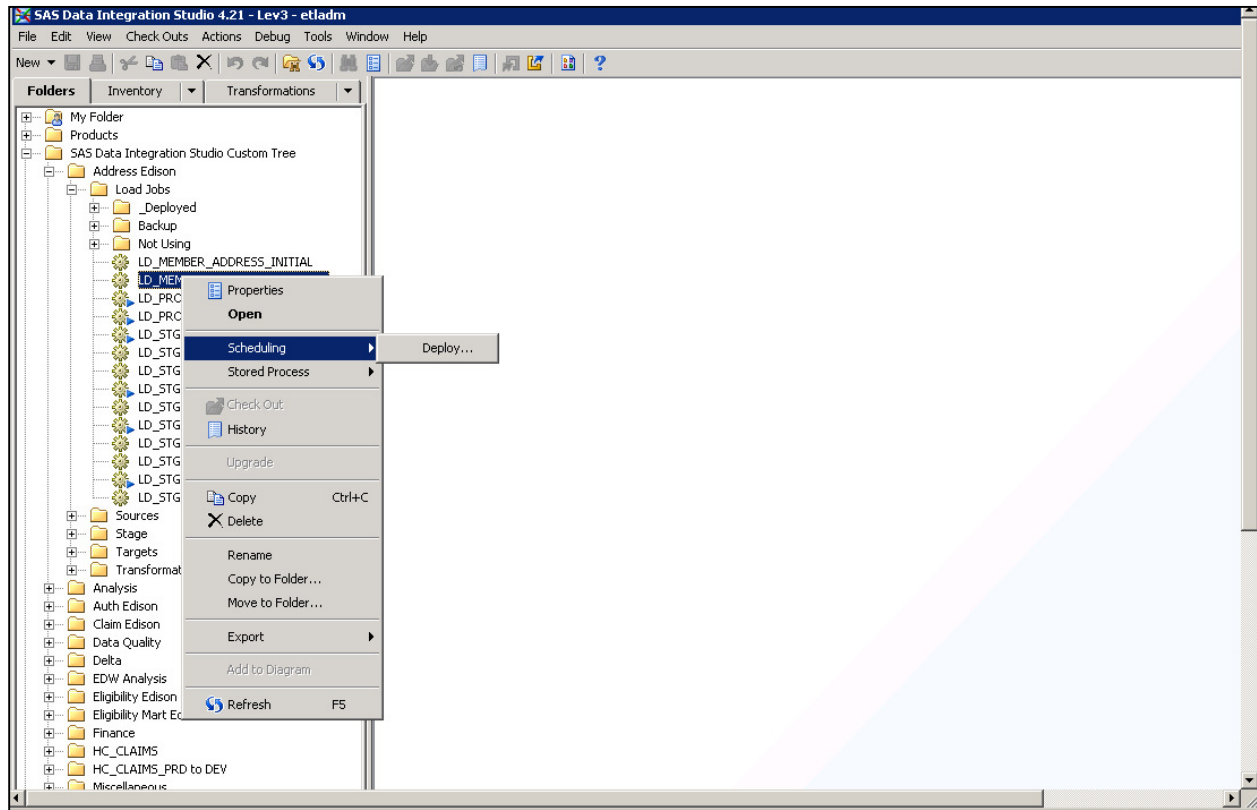

Figure 3. Deploy Jobs in SAS Data Integration Studio

Once jobs are deployed, they are added to a flow in SAS Management Console using the Schedule Manger plug-in wizard. The dependencies and triggers whether job status or file existence, are also set-up in SAS Management Console.

In SAS Management Console, there are two levels of creating a flow, dependencies and triggers. There are dependencies within a flow and triggers between flow streams. The dependencies within a flow are determined when creating a flow stream in SAS Management Console. As jobs are added to a flow, part of the definition includes building dependencies. These dependencies can consist of time, job or file events with an associated completion status, Figure 4. There are various completion event types including Completes successfully or Ends with any exit code for jobs with warnings, Figure 5.
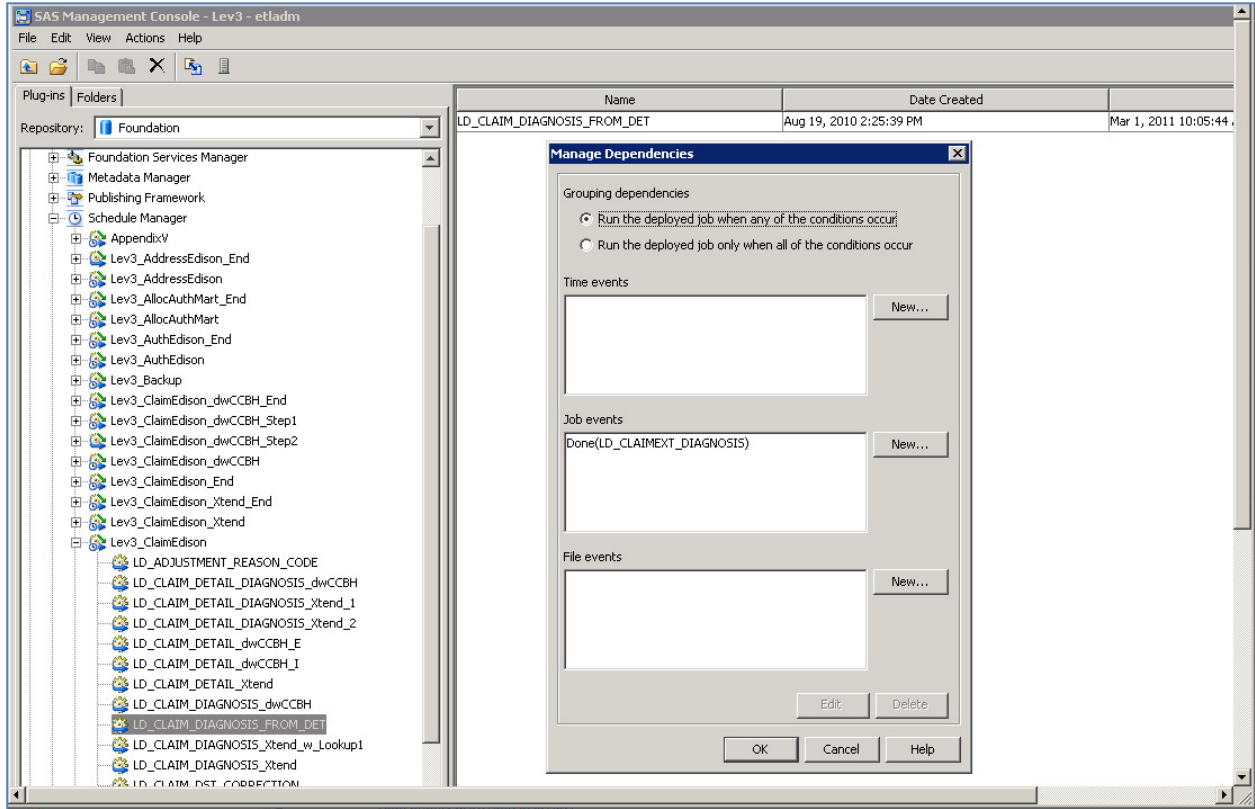
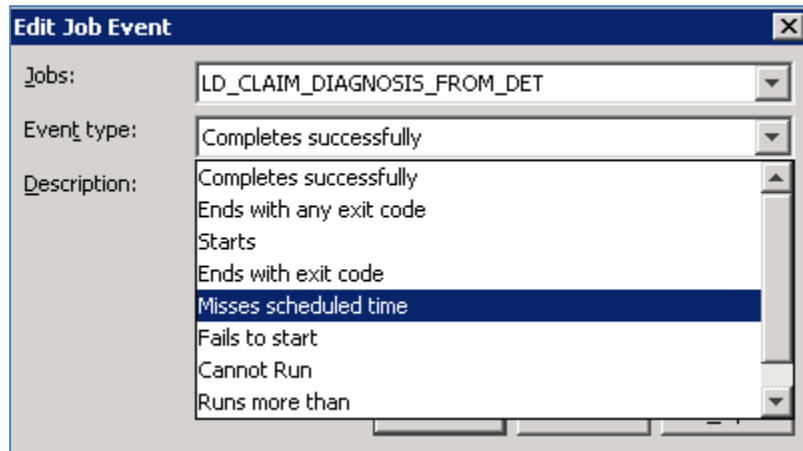Figure 4. Flow Dependencies Between Jobs

Figure 5. Completion Event Type

The file triggers built at the flow level are specified at the time of job scheduling in SAS Management Console. This is performed by issuing the following sequence of steps in SAS Management Console.

After the flow is created with dependencies, right click on the flow name and click Schedule Flow. When prompted, log in with the proper credentials for your system. The login used will need permissions to run the jobs. Click the drop-down list to select Manually in Scheduling Server, Figure 6.
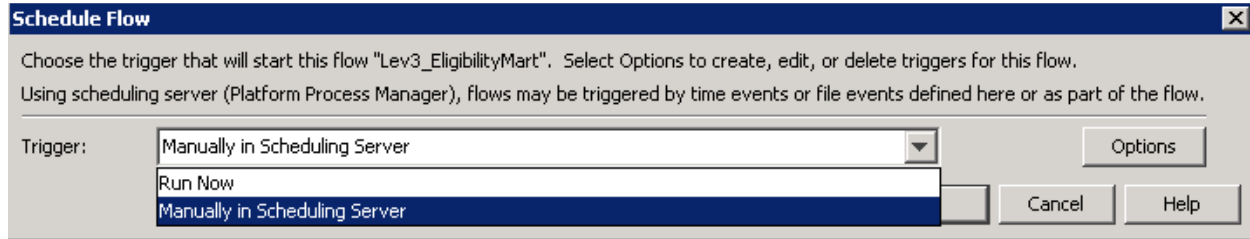

Figure 6. Set Schedule Flow to Manual

The triggers between flow streams are built at the time of submitting the job to the Platform Process Manager. Again, there are multiple options for building triggers. These options include time and file events and the ability to create custom triggers. Pick Options button and New File Event. Browse to specify the file that will be delivered. Pick the appropriate condition, in our case arrives, for the end file, Figure 7.
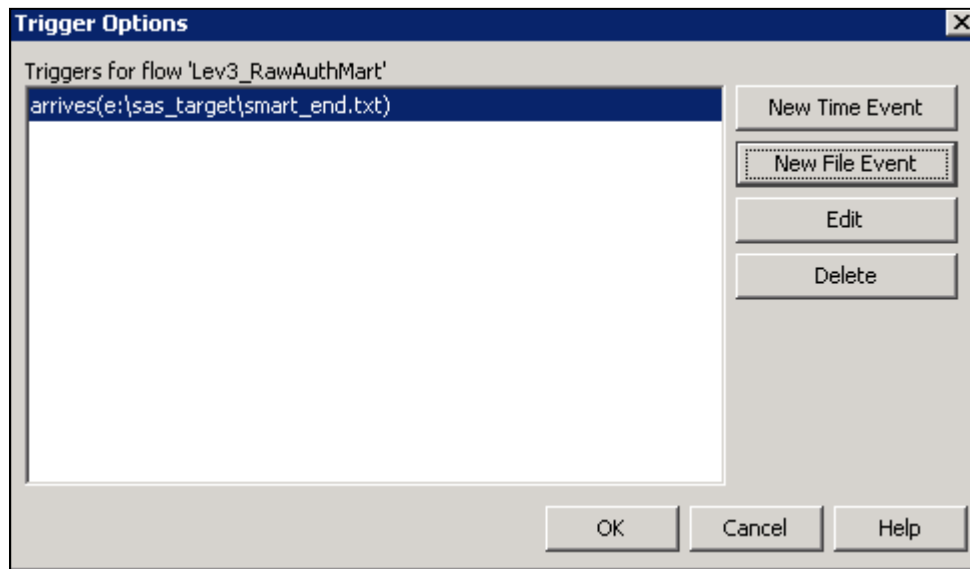

Figure 7. Add File Arrival Event Trigger

DOCUMENT THE FLOW

Once the job flow is designed, a visual flow chart with each process is helpful. The first, last or first and last job name can be included in each process box. Also included are the batch files that create the text file to be used as a trigger. By indicating the beginning and end jobs of each flow with the correlating triggers, whether they be preceding jobs or the existence of a file, a useful guide will be available. This flow chart provides a high level view of the entire process, Figure 8. This aids in the event of a job failure by providing a quick way to determine the restart point. A separate low level diagram that documents the dependencies within a job stream is also useful. Keep in mind that these flow diagrams are a work in progress and will change as the flow is updated.
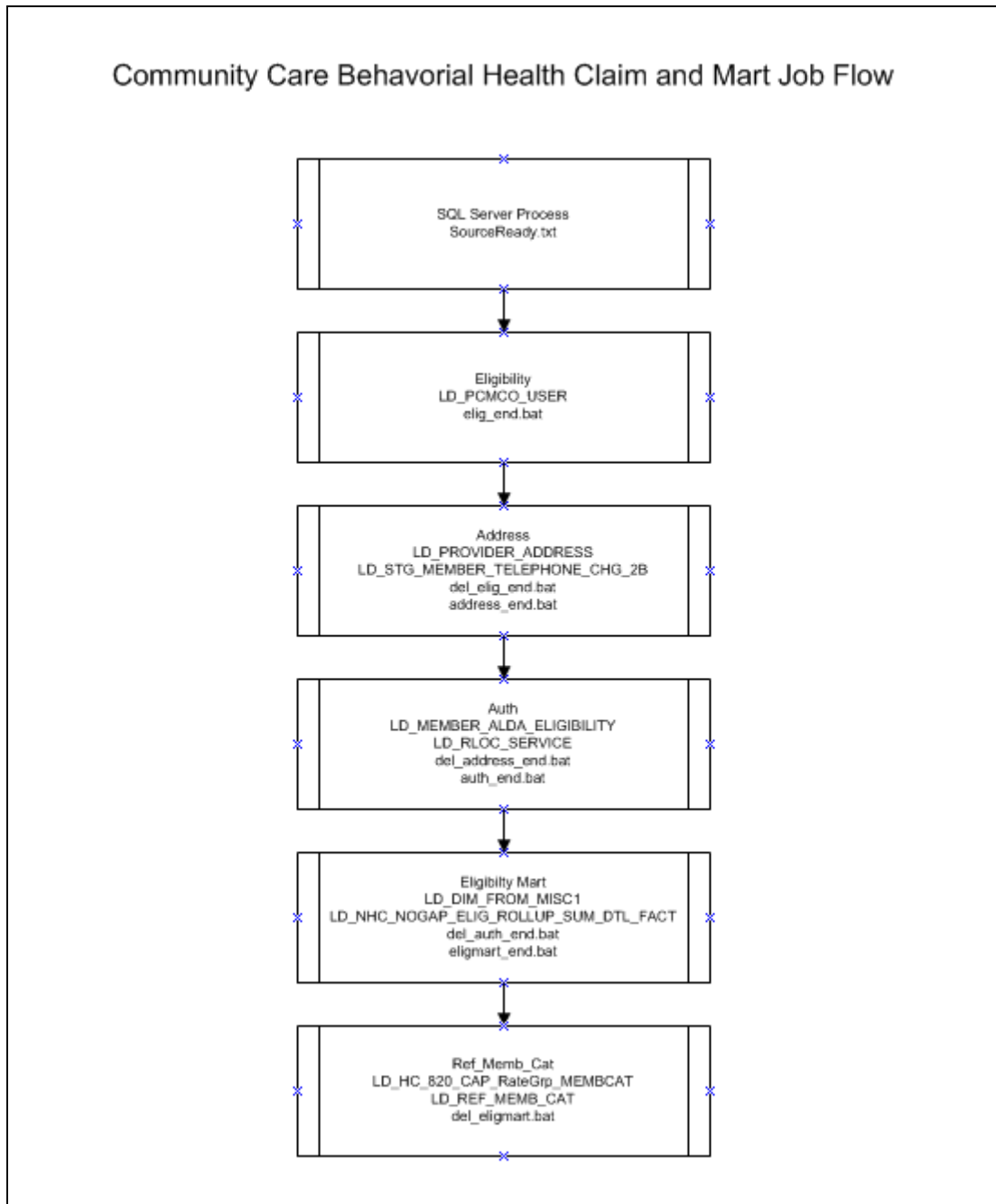

Figure 8. Flow Trigger Diagram

EXECUTE THE FLOW

There are multiple ways to execute the job flow. Besides using triggers, the flow can be run manually or run immediately. The scheduler also allows for one to run jobs at a specific time, day of the week or day of the month. SAS Management Console and Platform Suite allow for a lot of flexibility in scheduling the initiation of a flow to suite your environment.

MONITOR THE FLOW

With the rolling log option set, using the logs generated by each job in the SAS Data Integration Server batch log folder, the administrator is able to monitor the flow progress, Figure 9.

Lev2_AllocAuthMart_LD_DIM_MONTH_2012.01.25_08.31.00	1/25/2012 8:31 AM	LOG File	43 KB
Lev2_AllocAuthMart_LD_AUTH_DETAIL_RAW_2012.01.25_08.31.14	1/25/2012 8:40 AM	LOG File	145 KB
Lev2_AllocAuthMart_LD_AUTH_DETAIL_EDISON_2012.01.25_08.40.16	1/25/2012 1:29 PM	LOG File	213 KB
Lev2_SEHybridMart_LD_RLOC_DESC_2012.01.25_13.30.15	1/25/2012 1:30 PM	LOG File	44 KB
Lev2_SEHybridMart_LD_PAID_SERVICE_EVENTS_2012.01.25_13.30.33	1/25/2012 5:34 PM	LOG File	263 KB
Lev2_Ref_Memb_Cat_LD_HC_820_CAP_RateGrp_MEMBCAT_2012.01.25_17.35.16	1/25/2012 5:35 PM	LOG File	79 KB
Lev2_Ref_Memb_Cat_LD_HC_820_CAP_All_MEMBCAT_2012.01.25_17.35.35	1/25/2012 5:35 PM	LOG File	70 KB
Lev2_Ref_Memb_Cat_LD_REF_MEMB_CAT_2012.01.25_17.35.50	1/25/2012 5:37 PM	LOG File	130 KB

Figure 9. Log Files with Date and Time for Monitoring

One can also add job status notifications to the jobs in SAS Data Integration Studio. This is done by adding individuals to receive email alerts indicating job success or failure. If multiple support personnel are to be alerted of a job status, an alert for each person must be configured. Another option to assure the programmer or administrator the processes are progressing is to monitor the files created in the SAS WORK and UTILITY areas. Temporary SAS files are generated in this area. Once one studies the behavior of jobs and the associated temporary files, it is obvious if things are proceeding normally. If one has access to view database activity using the respective RDBMS monitoring tool, take advantage of this resource to assist in monitoring process progress. Or simply query the database activity looking for SELECTs, INSERTs or any other operation being performed in the source or target database. If the flow would fail for some reason, after investigating the cause, the SAS user is able to start a flow in the middle of the stream with the Platform Process Manager Rerun Now command, Figure 10.
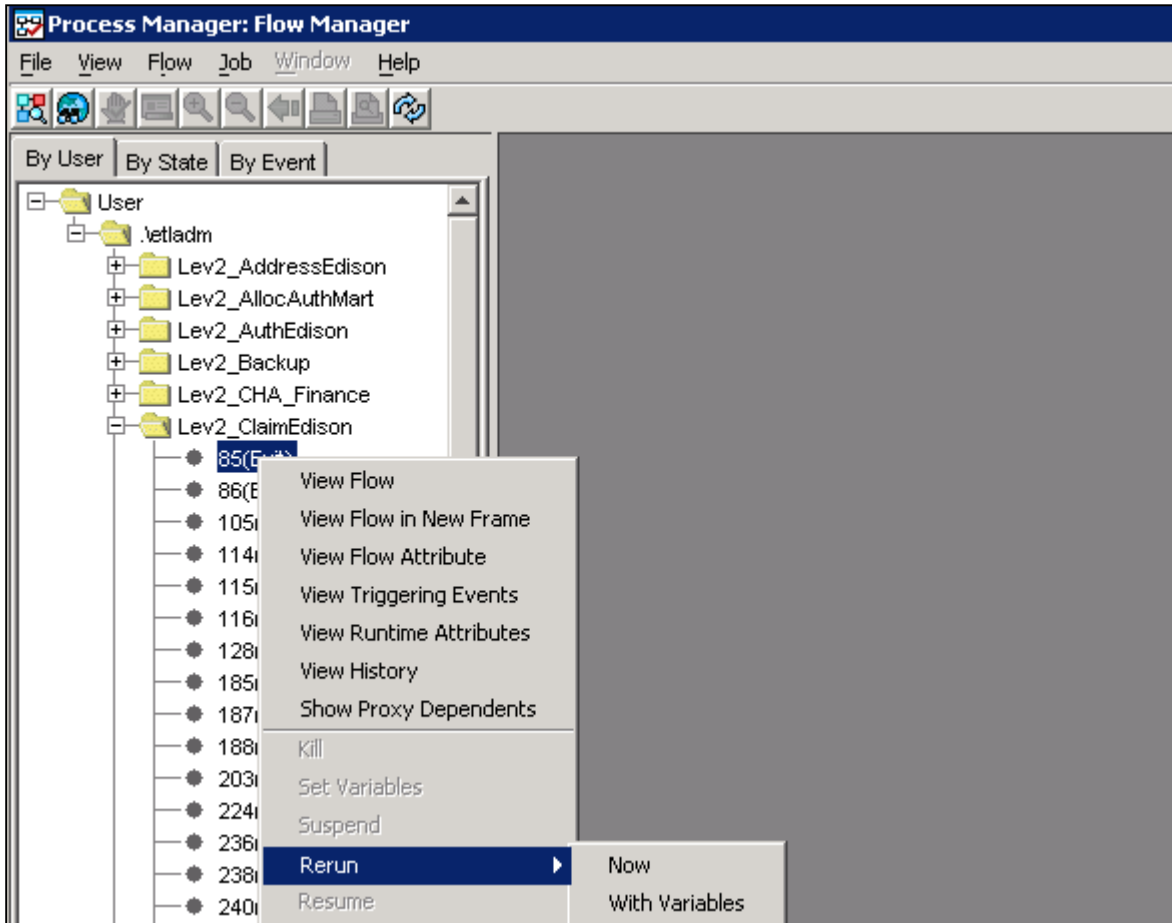
Figure 10. Rerun Job in Process Manager from Point of Failure.

MIGRATE THE FLOW

Once developed and tested, migrating flows between environments needs to occur in a sequence of steps. If done in the proper order, using the export and import wizard of SAS Data Integration Studio, jobs can be seamlessly migrated with nothing more than a name change, if necessary in your environment. All objects from the source will need to be exported to a SAS package including metadata, objects and jobs that are contained in the job stream. The actual job flows are also exported to a package in the same fashion. In the target environment, import the metadata, objects and jobs from the first package created. In the target environment, once the jobs are imported, be sure to deploy each job in SAS Data Integration Studio. Once all objects and jobs are imported and the jobs have been deployed, the job stream can be imported. Upon importing the job flow, verify that the Original deployed job from the source matches the Target deployed job, Figure 11.
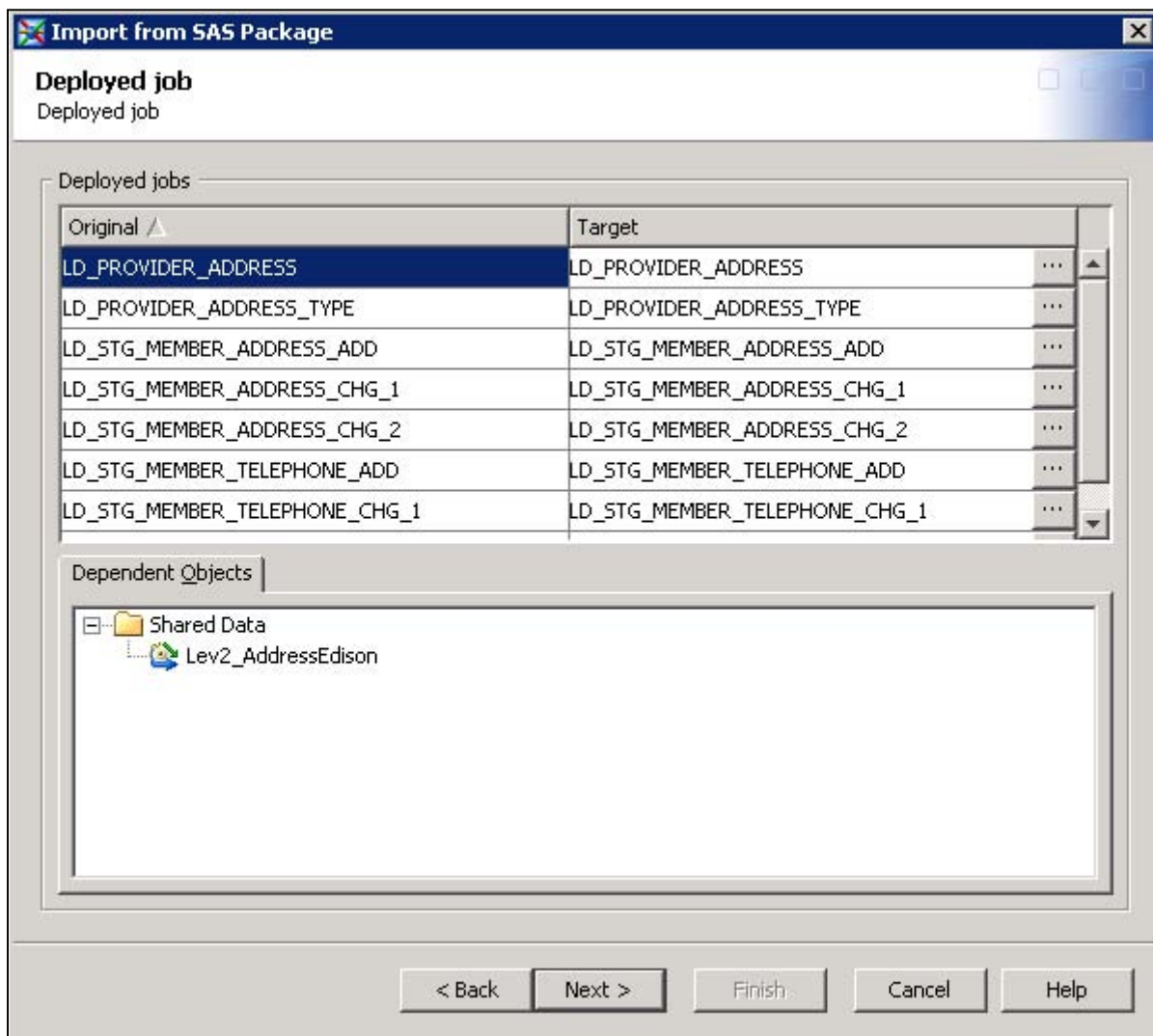
Figure 11. Deployed Jobs in Target Environment for Job Migration

ADJUST AND MAINTAIN THE FLOW

As mentioned previously, the flow will be a work in progress. Perhaps processes are added or removed from the flow or the order may have to be changed. As it will be referenced often, it is important to keep the visual diagram of the flow up-to-date to reflect the most current situation. Although having date and time included in the log name is useful, since logs will not be over-written, a log pruning script is recommended. This pruning script can be written in Perl or any programming language of choice therefore retaining only the most current logs on disk.

CONCLUSION

The savvy SAS user is always looking for ways to reduce their workload. Oftentimes, long-running processes occur off-hours, thus justifying the investment to create job flows. Using SAS Data Integration Studio, SAS Management Console and Platform Suite for SAS, one has the ability to allow SAS technology to handle job flow automatically. By scheduling flows and implementing monitoring processes, the user can spend less time performing menial tasks and more time learning new SAS features. Well written SAS jobs and programs are essential. Creating a smooth running flow using SAS tools allows the SAS Administrator to execute the programming tasks in an efficient manner.

CONTACT INFORMATION

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

Laura Liotus
Community Care Behavioral Health
One Chatham Center
Suite 700
Pittsburgh, PA 15219
412-402-8713
liotuslg@upmc.edu
http://www.ccbh.com/