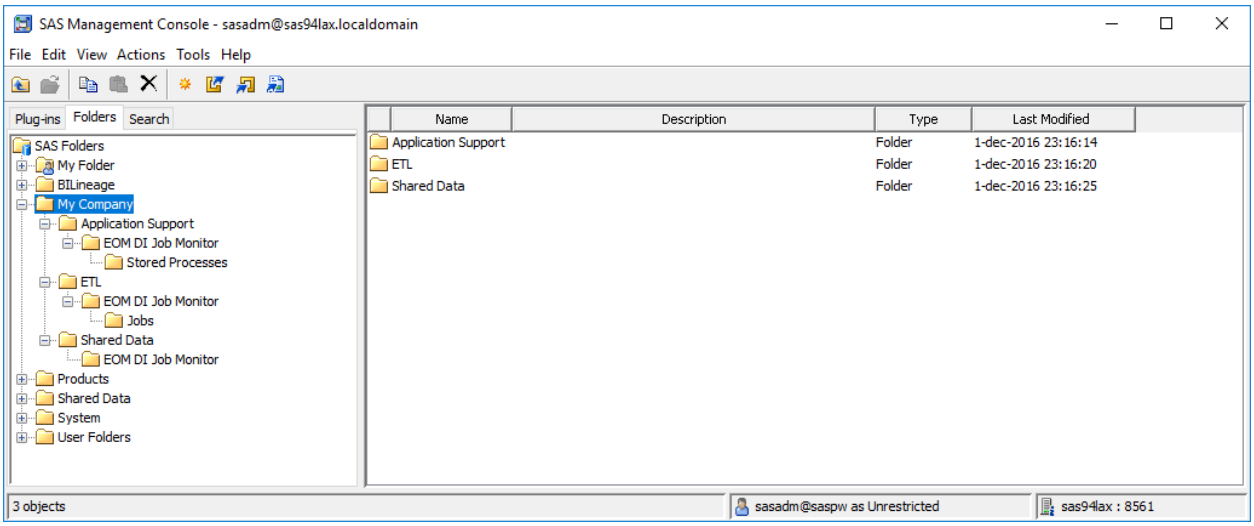
EOM DIMon 3.1 Installation Instructions for Linux

Bart Heinsius
May 21, 2017

Proposed metadata folder structure:

It is common practice to have separate SAS metadata folders for ETL programs, SAS Reports/SAS Stored Processes, and data. This document assumes installation in the SAS Metadata folder *My Company* shown here:



DIMon Batch Component Installation Instructions

Important note: When copying files from the installation package to the Linux file system, ensure that the files have the Linux file format on the Linux file system (LF as line termination string).

<table>
<tr><th>Nr</th><th>Instruction</th></tr>
<tr><td>1</td><td>Use SAS Management Console to create a SAS/SHARE or DBMS library with libref DIMON assigned to your SAS DI Application Server to store the DIMon tables. Your batch user needs UPDATE access to the tables in this library. Your SAS General Server User (e.g., sassrv) needs READ access to the tables in this library.

Notes:
- If you use a different libref than "DIMON" for your DIMon tables, add the following line to file "<sasappsrvcontextdir>/BatchServer/autoexec_usermods.sas":

<pre>libname dimon (<your libref>);</pre>
- For MySQL you need the following system variables in my.cnf:

<pre>sql_mode='ANSI_QUOTES' # allow " as an identifier quote character (next to backtick)
lower_case_table_names=1 # allow case-insensitive table names</pre>
- For MySQL you need to assign the SAS library to MySQL with PRESERVE_TAB_NAMES=NO

- If you experience slow performance when using the Postgres data store, please follow instructions for optimization at http://support.sas.com/kb/52/585.html</td></tr>
<tr><td>2</td><td>Create the required tables using the appropriate script for your database provided in the installation package folder "SASBatch\SQL":

<table>
<tr><th>Engine</th><th>Script</th></tr>
<tr><td>SAS/SHARE</td><td>dimon_create_tables_sas.sas</td></tr>
<tr><td>Postgres</td><td>dimon_create_tables_postgres.sql</td></tr>
<tr><td>MySQL</td><td>dimon_create_tables_mysql.sql</td></tr>
<tr><td>MS SQL Server</td><td>dimon_create_tables_sqlserver.sql</td></tr>
<tr><td>Oracle</td><td>dimon_create_tables_oracle.sql</td></tr>
</table>
</td></tr>
<tr><td>3</td><td>Register the tables that were created in step 2 in SAS metadata folder "/My Company/Shared Data/EOM DI Job Monitor".

Deselect the following options when registering the tables:

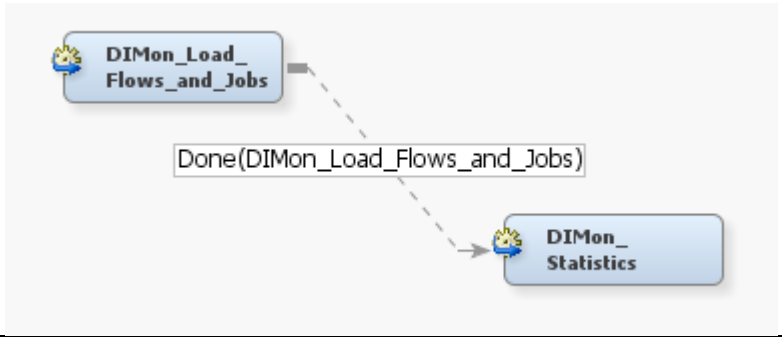
- Enable case-sensitive DBMS object names
- Enable special characters within table or column object name</td></tr>
</table>

4	Import SAS metadata package "SASBatch\SASPackages\dimon-batch.spk" from the installation package to SAS metadata folder "/My Company/ETL/EOM DI Job Monitor/Jobs". Map the tables to the tables you registered in step 3.
5	Copy all files from installation package folder "SASBatch\SASSteps" to folder "<sasappsrvcontextdir>/SASEnvironment/SASCode/Steps" on your SAS DI Application Server.
6	Create directory "<sasappsrvcontextdir>/SASEnvironment/SASCode/dimon" on your SAS DI Application Server. Copy all files from installation package folder "SASBatch\SASCode" to this directory.
7	Copy all files from installation package folder "SASBatch\BatchServer\Linux" to "<sasappsrvcontextdir>/BatchServer" on your SAS DI Application Server. By default, your DI jobs will be submitted with a customized -log option, possibly ignoring options you may have set yourself. Please read Appendix A. Batch Logging to see if this affects your installation and how to change it if you wish. To facilitate debugging you can set `DIMONDEBUG=YES` in dimon_usermods.sh, which creates the file /tmp/dimon-debug-$(USER).txt containing a list of environment variables.
8	Make a backup copy of file "<sasappsrvcontextdir>/BatchServer/sasbatch.sh" on your SAS DI Application Server.

9	Edit <sasappsrvcontextdir>/BatchServer/sasbatch.sh on your SAS DI Application Server:
	Insert before line:
	`exec "$SAS_COMMAND" -noxcmd -lrecl 32767 "$@" "${USERMODS_OPTIONS[@]}"`
	the following lines:
	`# EOM DI Monitor - prolog -- begin`
	`. $APPSERVER_ROOT/BatchServer/dimon_pre.sh`
	`# EOM DI Monitor - prolog – end`
	
	Insert after line:
	`exec "$SAS_COMMAND" -noxcmd -lrecl 32767 "$@" "${USERMODS_OPTIONS[@]}"`
	the following lines:
	`# EOM DI Monitor - epilog -- begin`
	`DIMON_JOBRC=$?`
	`. $APPSERVER_ROOT/BatchServer/dimon_post.sh`
	`exit $DIMON_JOBRC`
	`# EOM DI Monitor - epilog – end`
	
	replace line:
	`exec "$SAS_COMMAND" -noxcmd -lrecl 32767 "$@" "${USERMODS_OPTIONS[@]}"`
	
	with

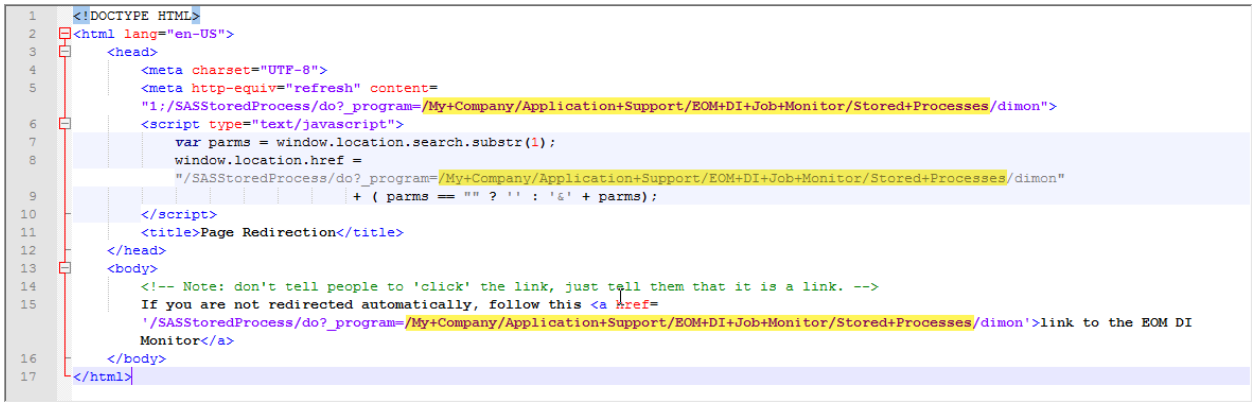
	`"$SAS_COMMAND" -noxcmd -lrecl 32767 ${DIMON_CMDLINEARGS} "${USERMODS_OPTIONS[@]}"`
10	Add the following line to file "<sasappsrvcontextdir>/BatchServer/autoexec_usermods.sas": `options fullstimer;`

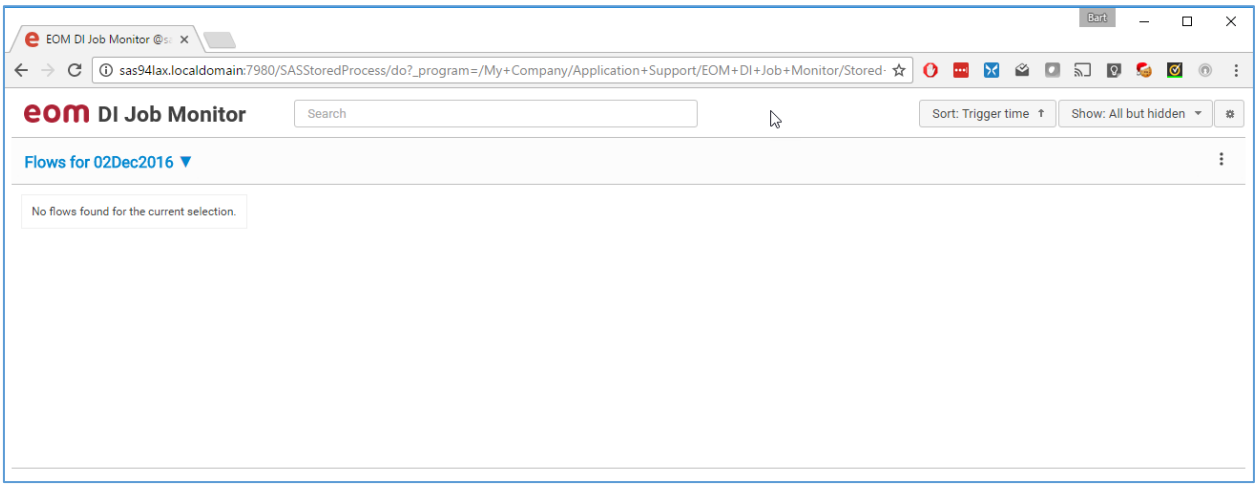
11	Check to see whether the APPSERVER_ROOT environment variable is available in your SAS batch programs. You can do this by submitting the following SAS code on your SAS DI Application Server:
	
	`%put %sysget(APPSERVER_ROOT);`
	
	If you see a valid path in the log, you're done with this step.
	If you see the following message in the log:
	
	`WARNING: The argument to macro function %SYSGET is not defined as a system variable.`
	
	Then add the following line to file "<sasappsrvcontextdir>/appservercontext_env_usermods.sh":
	
	`export APPSERVER_ROOT`

12	Using SAS DI Studio, run DI Studio job "/My Company/ETL/EOM DI Job Monitor/Jobs/DIMon_Load_Flows_and_Jobs" that you imported in step 4, on your SAS DI Application Server. You can ignore the warning that there are transformations that may be out of order in the job.
13	Deploy the SAS DI Studio jobs imported in step 4 for scheduling on your SAS DI Application Server. Use the SAS Management Console Schedule Manager plug-in to create a flow with the following deployed jobs: 1. DIMon_Load_Flows_and_Jobs 2. DIMon_Statistics 

	Schedule the flow to run daily, as the first step in your nightly batch.
-- END OF INSTRUCTIONS DIMON BATCH COMPONENT	

DIMon Web Application Installation Instructions

<table>
<tr><th>Nr</th><th>Instruction</th></tr>
<tr><td>1</td><td>Import SAS metadata package "Webapp\SASPackages\dimon-webapp.spk" into SAS metadata folder "/My Company/Application Support/EOM DI Job Monitor/Stored Processes". Assign the Stored Processes to run on your SAS Web Application Server (if you have that).</td></tr>
<tr><td>2</td><td>Copy the content of folder "Webapp\Webapps" to directory "<SASConfigDir>/Web/WebServer/htdocs/" on your SAS Web Application Server.</td></tr>
<tr><td>3</td><td>Copy the content of folder "Webapp\SASMacro" to directory "<sasappsrvcontextdir>/SASEnvironment/SASMacro" on your SAS Web Application Server.</td></tr>
<tr><td>4</td><td>Edit file "<sasappsrvcontextdir>\SASEnvironment\SASMacro\dimon_usermods.sas" on your SAS Web Application Server and review all settings, especially:
<table>
<tr><th>Setting</th><th>Description</th><th>Default value</th></tr>
<tr><td>libname</td><td>Optional alternative allocation of dimon library</td><td>none</td></tr>
<tr><td>sproot</td><td>Folder where dimon-webapp.spk was imported to</td><td>/My Company/Application Support/EOM DI Job Monitor/Stored Processes</td></tr>
<tr><td>webroot</td><td>Relative URL path to where the webapps components were copied to in step 2</td><td>/eom/dimon</td></tr>
</table>
</td></tr>
<tr><td>5</td><td>If you chose a different metadata location in Step 1 than the default ("/My Company/Application Support/EOM DI Job Monitor/Stored Processes"), update file "<SASConfigDir>/Web/WebServer/htdocs/eom/dimon/index.html" to reflect that in the sections marked yellow below:

</td></tr>
</table>

6	Start the EOM DI Job Monitor web application by navigating your browser to http://your-sasweb-server:7980/eom/dimon/ . If you don't have any flows scheduled yet you should see the following: 
-- END OF INSTRUCTIONS DIMON WEB APPLICATION COMPONENT	

Appendix A. Batch Logging

To facilitate real-time monitoring, the dimon_pre.sh script stores the fully qualified name of the SAS log file that your SAS batch job will be submitted with, in its tables (DIMON_JOB_RUNS). This requires that dimon_pre.sh knows the log file name that your SAS batch program will use before your SAS batch program actually executes. It turns out that the log file name is not always clear at this point in time in the batch execution process.

The following section explains how the batch SAS log file name is determined. After that, dimon options are given to manipulate the logfile name for your situation.

How the batch SAS log file name is determined

The log file name is specified in the -log option in the command that SAS Management Console's Schedule Manager plugin composes when a job is scheduled, for example:

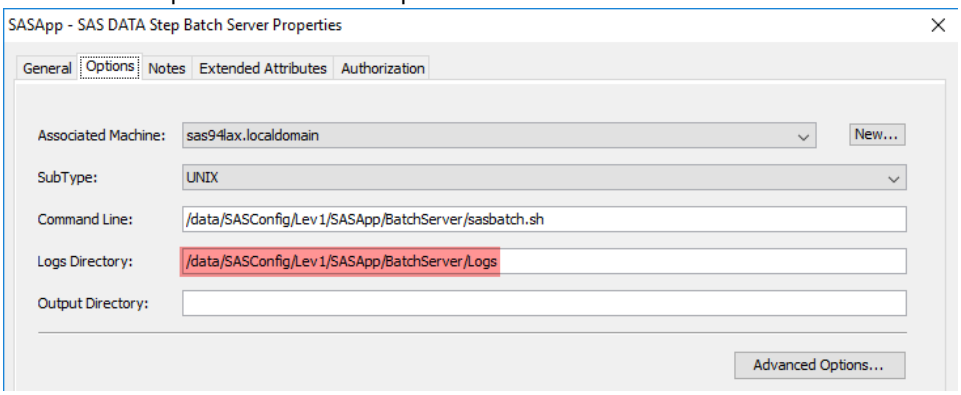
```
/data/SASConfig/Lev1/SASApp/BatchServer/sasbatch.sh -log
/data/SASConfig/Lev1/SASApp/BatchServer/Logs/DIMon_DIMon_Load_Flows_and_Jobs_#Y.#m.#d_#H.#M.#s.log
-batch -noterminal -logparm "rollover=session"  -sysin
/data/SASConfig/Lev1/SASApp/SASEnvironment/SASCode/Jobs/DIMon_Load_Flows_and_Jobs.sas
```

Schedule Manager composes the log file name from:

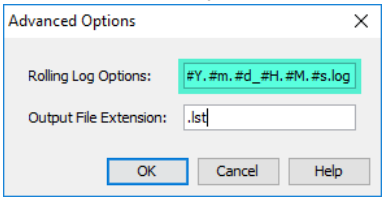
- SAS DATA Step Batch Server Properties
- SAS Management Console's Schedule Manager settings

In a default SAS installation with SAS configuration directory /data/SASConfig/Lev1 these are:

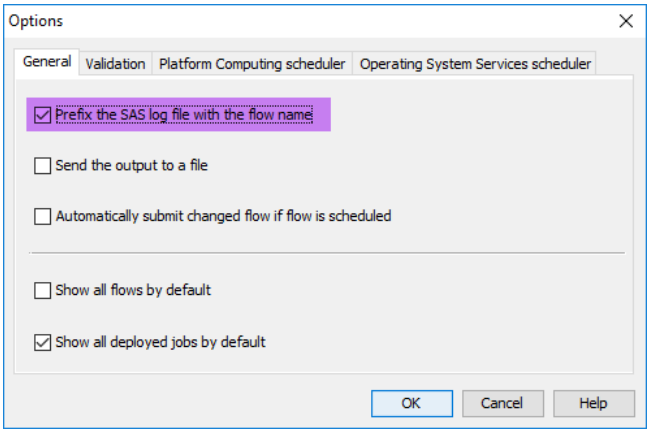
- SAS DATA Step Batch Server Properties:



with advanced options:



- SAS Management Console's Schedule Manager settings:

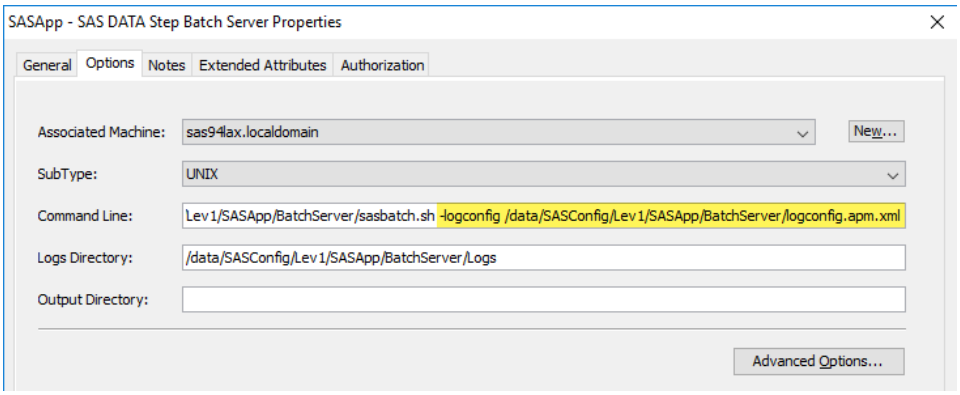


, to compose the following log file name for deployed job DIMon_Load_Flows_and_Jobs job in the DIMon flow:

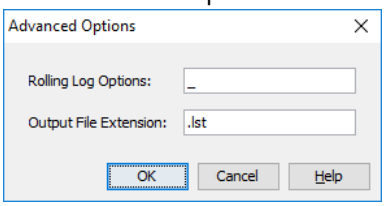
```
/data/SASConfig/Lev1/SASApp/BatchServer/sasbatch.sh -log
/data/SASConfig/Lev1/SASApp/BatchServer/Logs/DIMon_DIMon_Load_Flows_and_Jobs_#Y.#m.#d_#H.#M.#s.log
-batch -noterminal -logparm "rollover=session"  -sysin
/data/SASConfig/Lev1/SASApp/SASEnvironment/SASCode/Jobs/DIMon_Load_Flows_and_Jobs.sas
```

Together with the `-logparm "rollover=session"` option, the SAS executable resolves `#Y.#m.#d_#H.#M.#s` to an actual year, month, day, hour, minute, and second when executed.

If you configured your SAS DATA Step Batch Server to use the SAS Logging Facility, following the official SAS instructions you will have changed the SAS DATA Step Batch Server Command Line property to something like:



with advanced options:



Now the command that Schedule Manager composes is:

```
/data/SASConfig/Lev1/SASApp/BatchServer/sasbatch.sh -logconfig
/data/SASConfig/Lev1/SASApp/BatchServer/logconfig.apm.xml -log
/data/SASConfig/Lev1/SASApp/BatchServer/Logs/DIMon_DIMon_Load_Flows_and_Jobs_  -batch -noterminal
-logparm "rollover=session"  -sysin
/data/SASConfig/Lev1/SASApp/SASEnvironment/SASCode/Jobs/DIMon_Load_Flows_and_Jobs.sas
```

The actual log file name is now determined by the following section in logconfig.apm.xml:

```
<!-- Batch File Appender, match the output log file name to the input -log statement -->
<appender class="FileAppender" name="BatchFileAppender">
   <param name="Append" value="false"/>
   <param name="Unique" value="true"/>
   <param name="FileNamePattern" value="%S{App.Log}%d{yyyy.MM.dd_hh.mm.ss}.log"/>
   <layout>
      <param name="ConversionPattern" value="%m"/>
   </layout>
</appender>
```

where `%S{App.Log}` is resolved to the -log command line option (`/data/SASConfig/Lev1/SASApp/BatchServer/Logs/DIMon_DIMon_Load_Flows_and_Jobs_` in this example) and `%d{yyyy.MM.dd_hh.mm.ss}` is resolved to an actual year, month, day, hour, minute, and second.

Options for dimon

Dimon supports the following options for manipulating the logfile name.

Option name	**DIMON_SASLOGFILE_RESOLVE_YMDHMS**
Valid values	YES, NO
Default value	YES
Description	When this option is set to YES and the rolling log options for the SAS DATA Step Batch Server contain the string "`#Y.#m.#d_#H.#M.#s`", then "`#Y.#m.#d_#H.#M.#s`" is replaced by an actual YYYYMMDD_HHMMSS value.

Option name	**DIMON_SASLOGFILE_PREPEND_JOBID_FLOWID_USER**
Valid values	YES, NO
Default value	YES
Description	When this option is set to YES, the SAS batch logfile name is prefixed by the LSF Job Id, LSF Flow ID, and the user that executes the SAS batch job. Example: **1074_52_sasdemo**_DIMon_DIMon_Statistics_20170129_230456.log

Option name	**DIMON_SASLOGFILE_APPEND_DATETIME**
Valid values	YES, NO

Default value	YES

Description	When this option is set to YES and the rolling log option for the SAS DATA Step Batch Server ends with an underscore, <YYYYMMDD_HHMMSS>.log is appended to the SAS batch log file name. This is typical when ARM logging is configured and the logfile name is set in the logconfig.xml file. In addition to setting this option to YES you need to remove `%d{yyyy.MM.dd_hh.mm.ss}` from the FileNamePattern in your logconfig.xml as shown here:
	
	`<!-- Batch File Appender, match the output log file name to the input -log statement -->`
	`<appender class="FileAppender" name="BatchFileAppender">`
	`  <param name="Append" value="false"/>`
	`  <param name="Unique" value="true"/>`
	`  <param name="FileNamePattern" value="%S{App.Log}"/>`
	`  <layout>`
	`  <param name="ConversionPattern" value="%m"/>`
	`  </layout>`
	`</appender>`