



Ottawa Platform User Society

Running a Stored Process by
Querying Metadata Server

Ian J. Ghent, Credit Risk Manager – Bell Canada

June 3, 2009

Stored Processes Overview

- Standard interface to SAS code which allows execution/output to be triggered and delivered in a variety of formats
- Exist in the Metadata Repository
- Execution is handled by an instance of SAS called a “Stored Process Server”
 - Utilizes ODS to package output to various destinations or streams
 - There is a Stored Process Web Application that makes call backs to deliver images to the browser, which can also be used like SAS/IntrNet for web applications

One Problem

- Stored processes can only be triggered through “newer” client interfaces such as (Enterprise Guide, Add-in for MS Office, etc.)
- What if you want to use stored processes from the SAS DMS or batch interface
- This presentation goes through the steps to manually read a stored process definition from the Metadata Server, to get the source code location, enabling you to %include stored process code in your batch or DMS code

Metadata Server Overview

The Metadata Server functions as a:

- User Authentication, Access Control, and Profile Provider
- Server Controller
- Systems Configuration Tracking
- Library, Dataset, and OLAP Metadata storage
- Information Map Metadata Storage
- Stored Process Metadata Storage

Metadata Repository

- Is a database that contains metadata necessary for many of the SAS BI suite functions
- In SAS 9.1.3, it is just a series of related SAS tables
- Instead of doing queries against the tables, an interface of method calls is used, called the SAS Open Metadata Interface (OMI)
- OMI can be accessed via Java/.Net libraries or directly through SAS via Datastep/SCL functions (standard interface), and Proc Metadata (doRequest Method, preferred)

Proc Metadata

- Utilizes option declarations to set the server details (i.e. options metaserver= metauser=)
- Accepts and OMI XML query as input (can be inline or fileref)
- Returns an XML string which contains the results of the query
- XML syntax contains methods for adding/deleting metadata objects, querying to find metadata objects, and checking in/out of metadata objects
- Avoid manually altering metadata

About Metadata Objects

- Every Metadata object is assigned a unique 17 character ID in the form (*repositoryId.objectId*)
- In SAS 9.1.3 there are 165 metadata object types which store information about processes, servers, files and directories, SAS datasets and libraries, OLAP cubes, etc.
- Metabrowse is a utility available through the SAS DMS interface which allows you to view the various metadata objects

Stored Process Metadata

Stored Processes are stored in the BI Manager tree and contain the following metadata:

- Server specified to run the stored process
- Input streams
- Output format (none, streaming, transient package, permanent package)
- Prompts
- Location of source code

The Macros

- `%getTreeNodeInfo(location, treeType):`
 - Generic Macro that gets the node id and name of the node for a given path and tree type
 - For stored processes the tree type is “BIP Folder”
 - Returns the tree node id as a global macro variable “treeNodeId”
- `%getStoredProcessFileId(treeNodeId):`
 - Gets the source file id for a given tree node
 - Returns the source file id as a global macro variable “sourceFileId”
- `%getStoredProcessFilePath(sourceFileId):`
 - Gets the file path from the file id
 - Returns the path as a global macro variable “sourceFilePath”

The Macros Continued

- `%executeStoredProcess(location)`:
 - Is the wrapper macro for the whole process
 - Example of a stored process location is “/BIP Tree/Sample”
 - Performs an `%include` on the source file path (assumes macro running on same server)
- `%stpBegin()/%stpEnd()`:
 - Are empty filler macros since they are normally expected to be available when using an “official” stored process execution environment
 - Would normally trap output to an ODS destination or to a stream
- Prior to running the macros, metadata server options must have be submitted (i.e. `metaserver`, `metaport`, `metauser`, `metapass`, `metarepository`).

Limitations

- Stored processes have many client interfaces in which not all features are supported, the macros are no different
- Lots of functionality contained within the %stpBegin/%stpEnd macros are not reproduced
- Some uses may need customized code to fill functionality gaps
- Prompts are not handled but can be coded easily (macro variables)
- Execution not on the same machine as stored process code requires code linking/downloading mechanism (i.e. Desktop SAS DMS session)

Questions?

Ian J. Ghent
Credit Risk Manager
Bell Canada
ian.ghent@bell.ca

Additional Resources

- **%stpBegin/%stpEnd Macro Documentation**
 - http://support.sas.com/rnd/itech/doc9/dev_guide/stprocess/stpmacro.html
- **SAS Open Metadata Interface Documentation**
 - http://support.sas.com/documentation/onlinedoc/91pdf/sasdoc_913/omd_ref_10177.pdf
- **SAS OnlineDoc**
 - <http://support.sas.com/onlinedoc/913/docMainpage.jsp>