Configuration of
SAS® Forecast Server 2.1:
Stored Processes

Chapter 1

SAS Forecast Server Stored Processes: Configuration

Contents

What Is a Stored Process?

A stored process is a SAS program that is stored centrally on a server. A client application can then execute the program, and can receive and process the results. Stored processes enable you to maintain and manage code centrally, give you better control over changes, enhance security and application integrity, and ensure that every client executes the latest version of code that is available. Stored processes are like other SAS programs, except that they have an additional feature that enables you to customize the program's execution. This feature enables the invoking application to supply parameters at the time that the stored process is invoked. For example, if you have a stored process that analyzes monthly sales data, you could create a MONTH variable in the stored process. At execution time, you would supply the parameter MONTH=MAY to analyze May sales data. For more information about how to create a stored process and to invoke it in a client application, see the stored process section of the *SAS Integration Technologies: Developer's Guide*.

Configure SAS Forecast Server Stored Processes

NOTE: Before you configure any of the stored processes, it is recommended that you make a copy of the stored process and store them in a different directory.

A SAS Forecast Server project stores information hierarchically in project directories, or folders, depending on the server's operating system. For an opened project, you can execute a stored process interactively by using the information stored in the project, and by using the level and node of the hierarchy that currently is selected.

To configure a SAS Forecast Server stored process, perform the following steps:

1. Save your SAS code in the **!SASROOT\ForecastStudio\Reports\Samples** directory. If the folder does not exist, then create a new folder named **Samples**.

2. Open SAS Management Console and connect to a metadata repository as the SAS Administrator (e.g., sasadm).

3. Expand the **Stored Process Manager** folder.

4. Expand the **Forecast Studio** folder.

5. Expand the **Stored Processes** folder.

6. If there is not a **Samples** folder, then right-click the **Stored Processes** folder, select **New Folder**, and create a new folder named **Samples**.

7. Right-click the **Samples** folder and select **New Stored Process**.

8. Enter the name of your SAS program and a description (optional), and click **Next**.

9. Select **SASMain** as the SAS server. You need to add a source repository, which is a directory where the programs can be found.

   a) Click **Manage**.
   b) Click **Add**.
   c) Enter **c:\SAS\ForecastStudio\Reports\Samples** as the location, which is the same location where you saved your .sas file, and a description (optional).
   d) Click **OK**.
   e) Click **OK** again.

10. Select or enter the directory where you saved your SAS code as the source repository, enter the name of your SAS program as the source file, select **Transient result package** as the output, and click **Next**.

11. Click **Finish** on the Parameter panel.

You can find your new report in SAS Forecast Server from the **Forecasting → Reports** menu option. If you have a project opened, then you can run the report.

Pre-defined Macro Variables to Use with Stored Processes

To assist you with creating stored processes, SAS provides pre-defined macro variables. These macro variables contain information about the project as well as the location in the hierarchy that you are viewing currently. In general, the pre-defined macro variables that are used by SAS Forecast Server can be grouped into two categories:

- project macro variables

- interactive macro variables

You can use the project macro variables outside of SAS Forecast Server in SAS programs and stored processes by including the &HPF_INCLUDE file. You cannot use interactive macro variables outside of SAS Forecast Server in SAS programs and stored processes unless you define these macro variables yourself.

Interactive macro variables depend on the node of the tree (table) that currently is selected. Project macro variables do not depend on the tree node.

NOTE: If the number of variables is very large (BY, dependent, independent, reporting, etc.), the stored process macro variables can exceed the default maximum macro variable value length of 4096. The maximum length can be increased to 65534 by using the MVARSIZE= system option.

Table 1.1 lists the macro variables and their descriptions. You can use these macro variables in any stored process which you create for a SAS Forecast Server project.

Table 1.1   Pre-defined Macro Variables

Name	Description	Format
Macro Variables for Project Information		
HPF_DESC	Description of the Forecast Studio Project.	SAS label
HPF_PROJECT	The name of the project.	SAS name
HPF_PROJECT_LOCATION	System path to the project directory or folder.	System path
HPF_PROJECT_SERVER	SAS Workspace Server name.	Host name
HPF_INCLUDE	Project include file. This macro variable specifies the system path and file name that contains the SAS code to assign SAS libraries and catalogs associated with the project. By default, all libraries and catalogs are assigned with Read Only access (ACCESS=READONLY). For example, the following SAS code assigns project library names with Read Only access: %include "&HPF_INCLUDE"; For example, the following SAS code assigns project library names with Read and Write access: %let HPF_READ_ONLY = 0; %include "&HPF_INCLUDE";	System file name
Macro Variables for Project Input Data Information		
HPF_INPUT_LIBNAME	SAS library reference for the input data set.	SAS name
HPF_INPUT_DATASET	SAS member for the input data set.	SAS name
Macro Variables for Project Variable Information		

Table 1.1 Pre-defined Macro Variables (continued)

Name	Description	Format
HPF_NUM_BYVARS	Number of BY variables. If there are no BY variables, HPF_NUM_BYVARS is set to zero.	Nonnegative integer
HPF_BYVARS	List of BY variable names. The order of the BY variable names is the same as specified in the project. The macro variable is always defined; but if there are no BY variables, HPF_BYVARS is set to NULL.	List of SAS names separated by a single space
HPF_BYVAR&n	BY variable name listed in the nth position of the ordered list of BY variables (HPF_BYVARS). The first BY variable name is stored in HPF_BYVAR1, the second in HPF_BYVAR2, and the last is stored in HPF_BYVAR&HPF_NUM_BYVARS. If there are no BY variables (&HPF_NUM_BYVARS is zero), these macro variables are not defined.	SAS name
HPF_NUM_DEPVARS	Number of dependent variables. There is always at least one dependent variable.	Positive integer
HPF_DEPVARS	List of all dependent variable names. The order of the dependent variable names is the same as specified in the project.	List of SAS names separated by a single space
HPF_DEPVAR&n	Dependent variable name listed in the nth position of the ordered list of dependent variables (HPF_DEPVARS). The first dependent variable name is stored in HPF_DEPVAR1, the second in HPF_DEPVAR2, and the last is stored in HPF_DEPVAR&HPF_NUM_DEPVARS. Since there is always at least one dependent variable associated with a project, HPF_DEPVAR1 is always defined.	SAS name
HPF_NUM_INDEPVARS	Number of independent variables. If there are no independent variables, HPF_NUM_INDEPVARS is set to zero.	Nonnegative integer

Table 1.1 Pre-defined Macro Variables (continued)

Name	Description	Format
HPF_INDEPVARS	List of all independent variable names. The order of the independent variable names is the same as specified in the project. The macro variable is always defined; but if there are no independent variables, HPF_INDVARS is set to NULL.	List of SAS names separated by a single space
HPF_INDEPVAR&n	Independent variable name listed in the nth position of the ordered list of independent variables (HPF_INDEPVARS). The first independent variable name is stored in HPF_INDEPVAR1, the second in HPF_INDEPVAR2, and the last is stored in HPF_INDEPVAR&HPF_NUM_INDEPVARS. If there are no independent variables (&HPF_NUM_INDVARS is zero), these macro variables are not defined.	SAS name
HPF_NUM_REPORTVARS	Number of reporting variables. If there are no reporting variables, then HPF_NUM_REPORTVARS is set to zero.	Nonnegative integer
HPF_REPORTVARS	List of all reporting variable names. The order of the reporting variable names is the same order as specified in the project. The macro variable is always defined; but if there are no reporting variables, then HPF_REPORTVARS is set to NULL.	List of SAS names separated by a single space
HPF_REPORTVAR&n	Report variable name listed in the nth position of the ordered list of report variables (HPF_REPORTVARS). The first report variable name is stored in HPF_REPORTVAR1, the second in HPF_REPORTVAR2, and the last is stored in HPF_REPORTVAR&HPF_NUM_REPORTVARS. If there are no reporting variables (&HPF_NUM_REPORTVARS is zero), then these macro variables are not defined.	SAS name
Macro Variables for Project Time ID Information		
HPF_TIMEID	Time ID variable name	SAS name
HPF_TIMEID_FORMAT	Time ID format name	SAS format name
HPF_SEASONALITY	Length of the seasonal cycle A seasonality of 1 implies no seasonality.	Positive integer

Table 1.1 Pre-defined Macro Variables (continued)

Name	Description	Format
HPF_INTERVAL	Time interval name	SAS time interval
HPF_DATASTART	Start date/date-time/time value of the project The starting time ID value of the project input data set (&HPF_LIBNAME.&HPF_ DATASET).	SAS date/ date-time/ time value
HPF_DATAEND	End date/date-time/time value of the project The ending time ID value of the project input data set (&HPF_LIBNAME..&HPF_DATASET).	SAS date/ date-time/ time value
Macro Variables for Project Data Options		
HPF_SETMISSING	Missing value interpretation	
HPF_TRIMMISS	Missing value trimming	
HPF_ZEROMISS	Zero value interpretation	
Macro Variables for Project Diagnostic Options		
HPF_DIAGNOSE_ INTERMITTENT	Intermittency threshold value	Positive number
HPF_DIAGNOSE_SEASONTEST	Seasonality significance level	P-value
Macro Variables for Project Selection Options		
HPF_SELECT_CRITERION	Model selection criterion	
HPF_SELECT_HOLDOUT	Holdout sample absolute size Zero implies that the model fit is used for selection.	
HPF_SELECT_HOLDOUTPCT	Holdout sample percent size	Positive integer
HPF_SELECT_MINOBS_NON_MEAN	Minimum number of observations required before a non-mean model is selected	Positive integer
HPF_SELECT_MINOBS_TREND	Minimum number of observations required before a trend model is selected	Positive integer
Macro Variables for Project Forecast Options		
HPF_LEAD	Length of the forecast horizon or lead	Nonnegative integer
HPF_BACK	Number of time periods to omit	Positive integer
HPF_FORECAST_ALPHA	Confidence level size	P-value
Macro Variables for Project Hierarchy Information		
HPF_NUM_LEVELS	Number of levels in the hierarchy The levels of the hierarchy are numbered from 1 (the top of the hierarchy) to &HPF_NUM_LEVELS (the leaves of the hierarchy). If there is no hierarchy, then the number of levels is 1.	Positive integer

Table 1.1 Pre-defined Macro Variables (continued)

Name	Description	Format
HPF_LEVEL_DATAWHERE&n	Input data WHERE clause for the nth level, where *n* ranges from 1 to &HPF_NUM_LEVELS These WHERE clauses can be used to subset the input data sets for each level in the hierarchy to obtain information about the currently selected node. The WHERE clause at level 1 (the top) is stored in HPF_LEVEL_DATAWHERE1, the WHERE clause at the lowest level (the leaves) is stored in HPF_LEVEL_DATAWHERE&HPF_NUM_LEVELS. **NOTE:** You must unquote this macro variable. For example, %unquote (&&HPF_CURRENT_DATAWHERE&n);	SAS WHERE clause
HPF_LEVEL_OUTWHERE&n	Output data WHERE clause for the nth level where n ranges from 1 to &HPF_NUM_LEVELS These WHERE clauses can be used to subset the input data sets for each level in the hierarchy to obtain information about the currently selected node. The WHERE clause at level 1 (the top) is stored in HPF_LEVEL_OUTWHERE1, the WHERE clause at the lowest level (the leaves) is stored in HPF_LEVEL_OUTWHERE&HPF_NUM_LEVELS. **NOTE:** You must unquote this macro variable. For example, %unquote (&&HPF_LEVEL_OUTWHERE&n);	SAS WHERE clause
HPF_LEVEL_RECONCILE_DATASET&n	Reconciled forecast data set for the nth level	SAS name
HPF_LEVEL_RECONCILE_STATISTICS&n	Reconciled statistics data set for the nth level	SAS name
HPF_LEVEL_RECONCILE_SUMMARY&n	Reconciled summary data set for the nth level	SAS name

Table 1.1 Pre-defined Macro Variables (continued)

Name	Description	Format
HPF_LEVEL_BYVARS&n	List of BY variable names associated with the nth level, where n ranges from 1 to &HPF_NUM_LEVELS The variables names are separated by a single space. The BY variables at level 1 (the top) are stored in HPF_LEVEL_BYVARS1, the BY variables for the lowest level (the leaves) are stored in HPF_LEVEL_BYVARS&HPF_NUM_LEVELS.	SAS name
HPF_LEVEL_LIBNAME&n	SAS library reference for the nth level, where n ranges from 1 to &HPF_NUM_LEVELS The library reference at level 1 (the top) is stored in HPF_LEVEL_LIBNAME1, the library reference at the lowest level (the leaves) is stored in HPF_LEVEL_LIBNAME&HPF_NUM_LEVELS.	SAS LIBNAME
HPF_LEVEL_NSERIES&n	Number of series associated with the nth level, where n ranges from 1 to &HPF_NUM_LEVELS The number of series at level 1 (the top) is stored in HPF_LEVEL_NSERIES1, the number of series at the lowest level (the leaves) is stored in HPF_LEVEL_NSERIES&HPF_NUM_LEVELS.	Positive integer
Macro Variables for Project Event Information		
HPF_EVENTS	List of all event names	SAS name
HPF_EVENT_n	Name of the nth event	SAS name
HPF_NUM_EVENTS	Number of events	Positive integer
Macro Variables for Project Reconcile Information		
HPF_RECONCILE_LEVEL	Number of the reconciliation level The reconciliation level number ranges from 1 to &HPF_NUM_LEVELS, depending on the level of reconciliation.	Positive integer
HPF_RECONCILE_BYVAR	BY variable of reconciliation level	SAS name
HPF_RECONCILE_METHOD	Reconciliation method	SAS name
Macro Variables for Interactive Current Level Information		

Table 1.1 Pre-defined Macro Variables (continued)

Name	Description	Format
HPF_CURRENT_LEVEL	The level number associated with the current level. The current level number ranges from 1 to &HPF_NUM_LEVELS, depending on the currently selected level of the hierarchy.	Positive integer
HPF_CURRENT_LIBNAME	SAS library reference for the currently selected level of the hierarchy	SAS LIBNAME
HPF_CURRENT_LEVEL_ START	Start date/date-time/time value of the current level The starting time ID value of the input data set for the currently selected level of the hierarchy.	SAS date/ date-time/ time value
HPF_CURRENT_LEVEL_END	End date/date-time/time value of the current level The ending time ID value of the input data set for the currently selected level of the hierarchy.	SAS date/ date-time/ time value
HPF_CURRENT_LEVEL_NSERIES	Number of series (or nodes) associated with the currently selected level of the hierarchy	Positive integer
HPF_CURRENT_DATAWHERE	Input data WHERE clause for the currently selected node This WHERE clause can be used to subset the input data set to obtain information about the currently selected node of the hierarchy. **NOTE:** You must unquote this macro variable. For example, %unquote (&HPF_CURRENT_DATAWHERE);	SAS WHERE clause
HPF_CURRENT_OUTWHERE	Output data WHERE clause for the currently selected node This WHERE clause can be used to subset the output data sets to obtain information about the currently selected node. **NOTE:** You must unquote this macro variable. For example, %unquote (&HPF_CURRENT_OUTWHERE);	SAS WHERE clause
HPF_NUM_CURRENT_BYVARS	Number of BY variable names for the currently selected level of the hierarchy	Nonnegative integer

Table 1.1 Pre-defined Macro Variables (continued)

Name	Description	Format
HPF_CURRENT_BYVARS	List of BY variable names for the currently selected level of the hierarchy The macro variable is always defined; but if there are no BY variables, HPF_CURRENT_BYVARS is set to NULL.	List of SAS names separated by a single space
HPF_CURRENT_BYVARS&n	n^{th} BY variable name for the current level	SAS name
HPF_CURRENT_DEPVAR	Dependent variable name associated with the currently selected node of the hierarchy. This variable is contained in the list of dependent variables (HPF_DEPVARS).	SAS name
HPF_CURRENT_SERIESSTART	Start date/date-time/time value of the current node The starting time ID value of the series for the currently selected node of the hierarchy.	SAS date/ date-time/ time value
HPF_CURRENT_SERIESEND	End date/date-time/time value of the current node The ending time ID value of the series for the currently selected node of the hierarchy.	SAS date/ date-time/ time value
HPF_CURRENT_HORIZON	Horizon date/date-time/time value of the current node The time ID value of the start of the multi-step ahead forecast for the currently selected node of the hierarchy.	SAS date/ date-time/ time value
Macro Variables for Metadata Repository Information		
HPF_METADATA_HOST	Metadata server host name	Host name
HPF_METADATA_PORT	Metadata server port number	Port number
HPF_METADATA_REPNAME	Metadata repository name	Repository name
Macro Variables for Other Information		
HPF_READ_ONLY	Project access Read Only flag By default, stored processes have Read Only access with respect to the project libraries. This macro variable changes this access to Write. For example, the following SAS code assigns project library names with Read Only access: %include "&HPF_INCLUDE"; For example, the following SAS code assigns project library names with Write access: %let HPF_READ_ONLY = 0; %include "&HPF_INCLUDE";	Boolean

Table 1.1 Pre-defined Macro Variables (continued)

Name	Description	Format
HPF_DEFAULT_LOCATION	Default path to the system directory where the SAS Forecast Server projects are currently stored	System path name
HPF_ODSDEST	Used inside a stored process to control the output destination of ODS By default, HPF_ODSDEST is set to HTML, which means that the ODS output is in HTML format. You can change this value by adding a stored process parameter with the macro variable name of HPF_ODSDEST, which provides a choice at runtime. The valid values are the following: HTML (default) PDF RTF XML	ODS destinations