



SAS® 9.2 Enhanced Logging Facilities

Session 308-2008

Wednesday, March 19, 2008

9:00 – 9:50, Room 212

SAS Global Forum 2008

**THE
POWER
TO KNOW.**

Copyright Notice

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., SAS® 9.2 Enhanced Logging Facilities, Cary, NC: SAS Institute Inc., 2008.

SAS® 9.2 Enhanced Logging Facilities

Copyright © 2008, SAS Institute Inc., Cary, NC, USA.

All rights reserved. Printed in the United States of America. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, by any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc. Limited permission is granted to store the copyrighted material in your system and display it on terminals, print only the number of copies required for use by those persons responsible for installing and supporting the SAS programming and licensed programs for which this material has been provided, and to modify the material to meet specific installation requirements. The SAS Institute copyright notice must appear on all printed versions of this material or extracts thereof and on the display medium when the material is displayed. Permission is not granted to reproduce or distribute the material except as stated above.

U.S. Government Restricted Rights Notice. Use, duplication, or disclosure of the software by the government is subject to restrictions as set forth in FAR 52.227-19 Commercial Computer Software-Restricted Rights (June 1987).

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

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.

Copyright © 2008, SAS Institute Inc. All rights reserved.

www.sas.com

® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

Table of Contents

Table of Contents	i
Abstract	ii
SAS 9.2 Enhanced Logging Facilities	ii
SAS 9.2 Enhanced Logging Overview	1
Introduction	1
SAS 9.2 Enhanced Logging Framework	2
SAS 9.2 Metadata Server Events	2
Log4SAS Implementation Details	3
Log4SAS Configuration and Namespaces.....	3
Log4SAS Appender Implementations	3
Log4SAS Appender Attributes	3
StoredprocessServer Appender Attributes.....	4
Log4SAS Implementation Examples	4
Windows System Facility Appender	4
Log4SAS in the SAS Language	5
Application Response Measurement (ARM)	5
SAS Management Console Updates.....	5
Conclusion	5
APPENDIX	7
SAS 9.2 Enhanced Logging Framework	7
Acknowledgments	7
Recommended Reading	8
Contact Information	8

Abstract

SAS® 9.2 Enhanced Logging Facilities

The SAS® 9.2 enterprise server environment contains an enhanced logging facility which provides the ability to finely tune SAS logging and auditing behavior. SAS administrators and Enterprise IT administrators now have the power and flexibility to classify messages according to a well-defined namespace and dynamically enable diagnostic logging levels. SAS programmers can also exploit the enhanced logging features through the use of SAS 4GL language statements. Enhanced logging features include classifying messages according to audit, performance, administration or application namespaces, and diagnostic level such as trace, information, error, warning, or fatal conditions.

The logging facility contains a suite of predefined patterns to customize the output format for diagnostic messages. Through the use of appenders, administrators can direct messages to customized output destinations such as the Windows Event Viewer, UNIX system logs, the z/OS system logger, or operator's console. Additionally, messages can be directed to an external database for real-time analysis of enterprise activity or to perform historical analysis on audit, performance, or diagnostic events.

Through the enhanced logging facility, SAS and IT administrators, as well as SAS application programmers are now armed with a rich collection of options to optimize their logging and auditing requirements within a SAS Enterprise environment.

SAS® 9.2 Enhanced Logging Overview

This session is titled SAS 9.2 Enhanced Logging Facilities; however, during the development life-cycle we started using “log4sas” as the internal name and Log4SAS is now common language to describe this feature. Therefore, for purposes of this session, we will continue to use this term.

SAS has received feedback from many customers that the SAS® LOG features were no longer adequate to meet the needs of the distributed computing infrastructure. Whether it is the traditional SAS application writer or the SAS or IT administrator trying to isolate a distributed application processing, the logging facilities needed to be more flexible and dynamic.

This session will review the enhanced logging framework for Log4SAS, the implementation, configuration, and exploitation of the Log4SAS framework within SAS 9.2 applications and SAS® Management Console.

Through the maturity and evolution of the SAS® 8 and SAS® 9 life cycle, there’s been a shift toward increased regulatory and compliance requirements. Similarly, SAS products and solutions have moved from individual or department usage into larger organizations or across an enterprise environment. SAS products are integrating into executive-level dashboards and business critical processes; therefore, the pressing need from our customers for greater accountability requirements for what is occurring within the infrastructure related to SAS applications. In broad terms, we use the names, audit, regulatory, performance, and capacity management but, fundamentally, the questions are:

- Who is logged into the system?
- What are they doing?
- How many resources are they consuming?
- What went wrong?

SAS 9.2 Enhanced Logging (Log4SAS) provides excellent methods to answer those questions and allow administrators to easily extend the features for local site requirements. A key element to highlight regarding Log4SAS is that the features described in the session apply to all of SAS foundation products, including Base SAS®, EBI servers, and performance metrics for distributed workloads.

Introduction

The features and enhancements implemented for logging within SAS 9.2 were motivated by a number of factors, where the primary consideration is to provide a more **robust and customer-focused method for logging and monitoring** the SAS environment. The environment may consist of a single SAS user whose desire is to customize SAS log behavior for personal preferences, or an enterprise-wide system administrator that has a regulatory requirement to audit SAS usage and user ID access across the breadth of hosts and operating systems within the network.

The key features that comprise the SAS enhanced logging facilities include:

- classification of events based on a well-defined hierarchical namespace; for example, Audit, Application, or Performance
- classification of events based on a diagnostics level; for example trace, debug, and information type messages
- ability to direct messages to a single or multiple output destinations simultaneously, with customized formatting attributes, based on a well-defined set of patterns

Naturally, with these enhanced features, there are some cognitive and linguistic challenges that have to be overcome to fully understand and exploit these features. This session will introduce the fundamental concepts associated with

the enhanced logging facility, review implementation specifics for a number of the logging features, review implementation examples, and finally offer a few suggestions and recommendations on how to extend the features into an enterprise environment.

SAS 9.2 Enhanced Logging Framework

Customers familiar with open source facilities or web application may recognize that Log4SAS shares a heritage with Apache Logging Services, a feature more commonly known as LogJ. Log4J is a Java-based logging feature, along with associated ports to other languages, Log4cxx, Log4net, Log4php—all of which share a common set of logging semantics. This explains our (SAS) usage of the name Log4SAS. As much as possible, we have faithfully implemented the same processing architecture as version 1.3 of Log4J implementation, with extensions that are tailored for SAS customers.

An [illustration](#) of the Log4SAS infrastructure denotes the various components within the architecture and the interrelationships. For illustration purposes, the components are represented as unique symbols; however, for the most part, the log4sas framework is resident within a single instance of the application or process.

The framework is initialized during instantiation of a SAS process, Batch, DMS, or a Server, represented by the process symbol on the upper left of the image. At this point, we are focusing on the key concepts of the framework, and the configuration will be discussed in subsequent pages. Earlier in the session, the term “event” was briefly introduced; here, the event takes shape and importance. Associated with an event are properties, the namespace, diagnostic level, and message contents. These properties are used to determine if an event will ultimately result in a message rendered to an output destination.

The event includes the namespace property, which is linked to a configured “logger.” Loggers consume the event and, based on the configuration the event is routed to one or more configured “appender” definitions. Then, based on the appender configuration, the event is emitted to an output destination as a fully formatted message.

In principle, the definitions and concepts are very straightforward, where an event is issued with properties, the logger consumes the event and routes the event to an appender, with an output definition such as a file or other defined destination.

It is important to emphasize at this point, if that was all there was to this process, that we could conclude the session and proceed merrily on our way. However, some very critical features need to be expanded to fully develop the capabilities of the framework.

SAS 9.2 Metadata Server Events

The SAS 9.2 Metadata Server, along with all the SAS servers and Base SAS, participate in the Log4SAS framework as an introduction into the characteristics of a logging event. The following three messages represent the elements of an event. For example the namespaces depicted include:

- Admin.Operations
- App.Meta.IO.IMDB.Wheres
- Audit.Meta.Security.UserAdm

Additionally, observe the diagnostic levels for each of the events, INFO and TRACE, and then the contents of the original message. Note that the fully formatted messages contain a number of additional patterns that will be reviewed in the next section.

Conceivably, a configuration may create one or more loggers that consume each level in the namespace hierarchy and redirect events to multiple output destinations. In practice, a much more practical usage is that while working on a Metadata tracking entry in technical support, you can configure a logger to only consume App.Meta.IO.IMDB events.

Log4SAS Implementation Details

Log4SAS Configuration and Namespaces

The SAS publications team is developing a SAS 9.2 administration guide which fully describes the features and exploitation of Log4SAS. The following information describes some of the fundamental features related to Log4SAS. Log4SAS is enabled and configured through the use of a `-LOGCONFIGLOC` command line options specification. If this option is not enabled, SAS logging behaves as it did in SAS 9.1 and earlier releases. The option takes as input an XML file that contains the logger and appender definitions. Base SAS applications and SAS servers may exploit the feature, and a comprehensive set of samples is being provided for the more common configurations. An excerpt of a SAS StoredprocessServer `logconfig.xml` file includes:

```
<logger name="Audit" additivity="false">
  <level value="Info"/>
</logger>

<!-- Root logger -->
<root>
  <level value="Error"/>
  <appender-ref ref="TimeBasedRollingFile"/>
</root>
```

The key logger definition features are

- Audit logger, Additivity set to false; additivity controls how far through the hierarchy an event is percolated.
- Diagnostic level set to INFO: info, warn, error, and fatal events are processed. Absent a specific diagnostic level, the level is inherited based on the top-level logger definition.
- Root logger sends error and fatal events to a TimeBasedRollingFile appender. Appenders are discussed in the next section.

Log4SAS Appender Implementations

The primary goal of Log4SAS is to enhance the logging and monitoring capability of the SAS environment for both the SAS and IT administrator. This is accomplished in Log4SAS by providing a number of methods to customize the output destination and output format of a SAS log. Each appender has a unique set of configuration options as well as a common set of output format patterns and conversion specifiers. Appender features include writing to an output file, or time-based rolling file, writing to operating system-specific destinations for Windows, UNIX, and z/OS environments, performance (ARM appender), external database appender or SAS internal-use appenders for workspace or SAS/CONNECT servers.

Log4SAS Appender Attributes

Each individual appender has some unique configuration properties that are described in the administration guide, and there are also a common set of attributes for all appenders. The attributes include conversion specifiers which control the output format of the resulting messages, similar to a C language `sprint()` function call. Conversion

formats include date-time format, user ID, source file, and line number as well as the actual message contents that were specified on the log4sas event.

In the logger, the diagnostic level controls which messages are consumed by the logger, the appender may also be configured to process certain diagnostic levels. In the appender, the levels are controlled with a “threshold” definition. Additionally, the appender includes filters, which are used to finely tune the messages that are routed to the output location.

StoredProcessServer Appender Attributes

To augment the previous logconfig.xml definition for the server, below is an excerpt of the appender definition.

```
<appender class="RollingFileAppender" name="TimeBasedRollingFile">
<param name="Append" value="true"/>
<param name="ImmediateFlush" value="true"/>
<rollingPolicy class="TimeBasedRollingPolicy">
<param name="fileNamePattern"
value="C:\SS\FS\Lv\SASAp\StoredProcessServer\Logs\STPServer_%d_%S{pid}.log"/>
</rollingPolicy>
<layout>
<param name="HeaderPattern" value="Host: '%S{hostname}', OS:
'%S{os_family}', Release: '%S{os_release}', SAS Version: '%S{sup_ver_long2}',
Command: '%S{startup_cmd}'"/>
<param name="ConversionPattern" value="%d %-5p [%t] %u - %m"/>
</layout>
</appender>
```

Key attributes for the appender include:

- class="RollingFileAppender" and rollingPolicy definition
- "fileNamePattern" parameter which include dynamic substitution parameters for the date-time as well as the process PID
- "HeaderPattern" syntax which defines what information will be written to the log at process initialization and file rollover time
- "ConversionPattern", in this case, date-time, diagnostic level, thread ID, user ID, and message

Log4SAS Implementation Examples

Windows System Facility Appender

An example of the information that is written by the system facility appender for Windows includes the Admin.Session namespace (logger). SAS 9.2 applications have been enhanced to issue Log4SAS events to the Admin namespace, the events include the conversion patterns for the appender, a unique message ID for each server and server state, and the command line. In this instance, a SAS/CONNECT session was started and stopped, with corresponding messages being emitted to the Windows Event Viewer. The same type of messages can be directed to the UNIX syslogd or z/OS console or z/OS System Logger facility.

This feature allows a SAS or IT administrator to very easily ascertain all the SAS activity that is being launched throughout the enterprise.

Log4SAS in the SAS Language

The SAS language has been enhanced to support calling into the Log4SAS framework through a series of SAS macro, data step, and call function Log4SAS calls. Support includes the ability to create user-defined loggers, appender and conversion patterns, as well as write log4sas events.

The sample below emits an Audit.Login event, diagnostic level set to error with a customized message. The resulting output message contains the conversion pattern for date-time, namespace, diagnostic level, thread ID, and the original message contents.

```
%log4sas(); data _null_;
rc = log4sas_logevent("Audit.Login", "INFO", "Userid: sasjhg"); run;
...
2007-11-27T13:49:54,722 Audit.Login INFO 00000016 Userid: sasjhg
```

Application Response Measurement (ARM)

An existing feature for SAS 9 applications was the ability to instrument SAS language code to produce application response measurement records to ascertain performance of the application. ARM support included SAS language macros as well as exploitation on various SAS foundation servers. In SAS 9.2, it is important to emphasize that existing SAS 9 applications will continue to execute unmodified. This allows customers who have customized their instrumentation to continue to use ARM without perturbing their environment.

SAS 9.2 enhancements include a number of refinements to the ARM support and include the following features:

- integrated into the Log4SAS framework
- supports both ARM 2.0 and ARM 4.0 specifications
- cooperates with vendor ARM 2.0 and ARM 4.0 libraries
- default transaction correlation for nested transactions
- new SAS language PERF macros with additional metrics

SAS 9.2 ARM support is implemented using the Perf.ARM Log4SAS namespace; it includes an ARM appender and a comprehensive set of metrics for transaction performance information. Examples of using the ARM features in Log4SAS include SAS/CONNECT, SAS® Data Integration Studio, SAS® OLAP Server, and distributed transaction processing across the SAS server tier and SAS mid-tier.

SAS Management Console Updates

SAS 9.2 Management Console contains a vast array of new features for the SAS or IT administrator. SASMC also includes features that directly integrate into the Log4SAS framework. Each of these features was implemented to enhance the monitoring, management, and administration of the EBI environment. Enhancements include:

- server management functions to start, stop, pause and quiesce servers
- active client session status
- assign Log4SAS diagnostic levels
- direct log monitoring for connected servers

Conclusion

The SAS 9.2 Enhanced Logging Framework was designed to significantly enhance the ability of the SAS or IT administrator to understand, log, and monitor the SAS execution activity within the enterprise environment. The

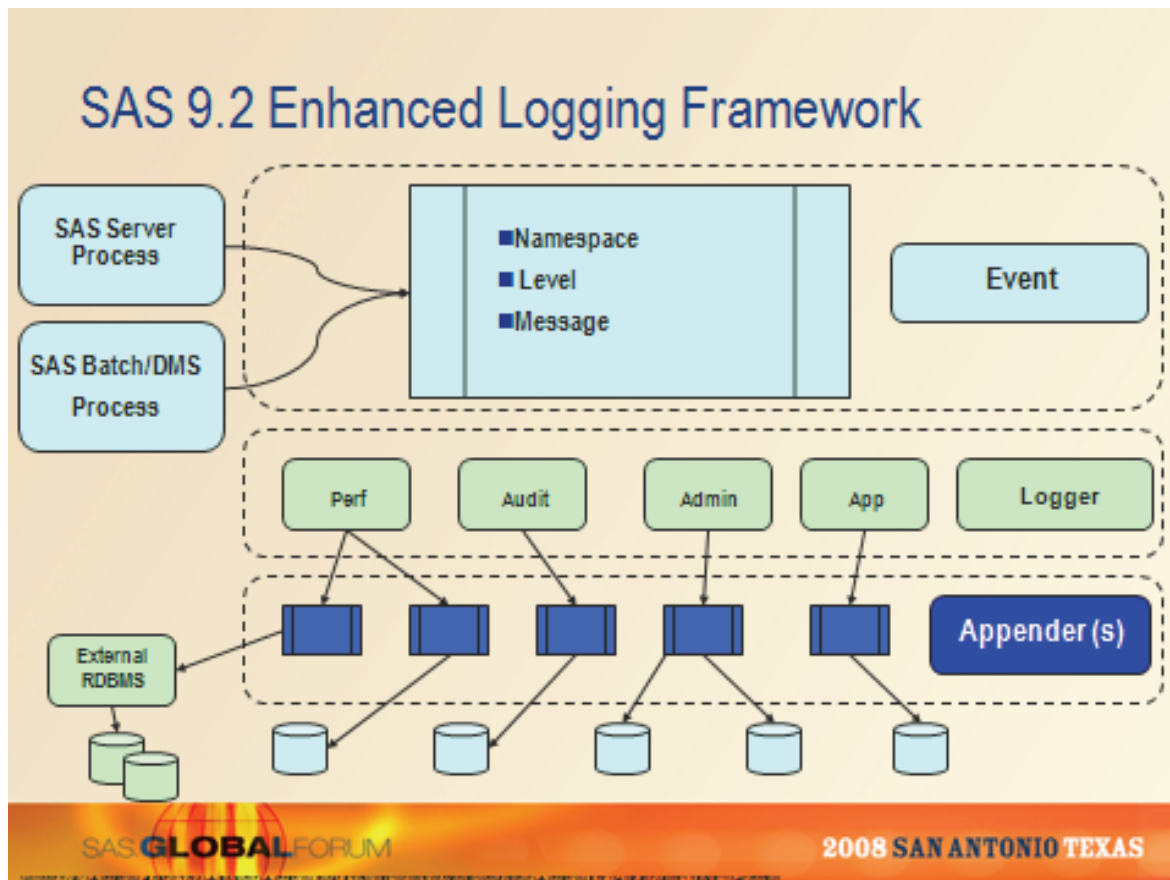
interfaces are flexible and easily customized for site installation requirements. Additionally, through the use of a few key features, IT and business folks can optimize the SAS framework for their business requirements. Examples include:

- Capture start and stop events for every instance of a SAS application, and then write this information to an operating system specific location using the Admin namespace.
- Generate compliance, auditing, or regulatory reports using the Audit namespace.
- Generate accounting and resource utilization reports with the Perf.ARM namespace.
- Utilize improved diagnostic features with SAS Management Console.

For a complete description of the features of the enhanced logging features, refer to the SAS 9.2 Enhanced Logging Administrator guide.

APPENDIX

SAS 9.2 Enhanced Logging Framework



Acknowledgements

The SAS 9.2 Enhanced Logging Facilities feature required the creative talents and collaboration of a significant number of folks from SAS development, testing and documentation teams. Special thanks and acknowledgement to a few key individuals from across the Institute for their contributions towards design and implementation of the logging facility:

- Platform Research and Development: Beth Ayres, Biff Beers, Darren Key, Doug Haigh, Joe Mudd
- Platform Research and Development: John Brown, John Gillikin, Larry Noe, Linda Hamm
- Platform Research and Development: Mary Beth Ward, Pat Garrett, Quan Zhou
- Business Intelligence Clients: Donna DeCapite, Nancy Rausch, Paul Polo
- Publications: Elizabeth Maldonado, Nancy Pipes

Recommended Reading

- Apache Logging Services: <http://logging.apache.org/>
- The complete manual, Log4j: <https://www.gos.ch/shop/products/log4j/>
- SAS 9.2 Enhanced Logging Administration Guide: <http://tinyurl.com/2w29l5> (2Q 2008)

Contact Information

Please contact the author if you have any additional questions, comments or require clarification on information presented in this paper. Your comments and suggestions are gratefully appreciated.

Gary T. Ciampa, Software Manager
Platform Research and Development
SAS Institute Inc.
SAS Campus Drive
Cary, NC 27513
Work Phone: 919.531.6140
Fax Number: 919.677.4444
Gary.Ciampa@sas.com
<http://support.sas.com/rnd/emi>

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.

