



Audit Messages for SAS Library Access

Peter Hobart

Auditing

Audit Messages for SAS Library Access

- Introduced in SAS 9.2 M3
- Track any contact with a table by interactive browsing or running code
- The logging facility provides three loggers to audit access to SAS libraries,

Logger	Action
Audit.Data.Dataset.Delete	writes a message to an audit log when a SAS data set is deleted
Audit.Data.Dataset.Open	writes a message to an audit log when a SAS data set is opened
Audit.Data.Dataset.Rename	writes a message to an audit log when a SAS data set is renamed

Auditing

information available from a SAS library

- Action taken on the SAS data set: OPEN, DELETE, or RENAME.
- Status of the action taken on the SAS data set: SUCCESS or FAILED.
- Return code from the action and the associated message (if any).
 - Warning and note messages have a status of SUCCESS and negative return codes.
 - Error messages have a status of FAILED and positive return codes.
- The libref.
- The engine associated with the library.
- The library member name (Table name).
- If a SAS data set was renamed, the new member name.
- The library member type, such as catalog or data set.
- The mode that the library was opened for: INPUT, OUTPUT, or UPDATE.
- The path to the library or to a database table.

Auditing

Notes

- Audit messages are not available for in-database tables, filerefs, or SAS OLAP cubes.
- Based on the XML configuration files that SAS provides, auditing messages appear when the logging threshold is TRACE

Auditing Setup

- Edit the logconfig.xml file for the relevant server(s)
 - Typically, workspace servers and stored process servers

Auditing

Sample Logger

- `<!-- Audit.Data.Dataset.Open logger definition -->`
- `<logger name="Audit.Data.Dataset.Open" additivity="false">`
- `<appender-ref ref="AuditLibraryFile"/>`
- `<level value="Trace"/>`
- `</logger>`

Auditing

Sample Appender

```
<appender name="AuditLibraryFile" class="FileAppender">
  <param name="Append" value="true"/>
  <param name="ImmediateFlush" value="true"/>
  <param name="fileNamePattern"
value="C:\SAS\Config\Levl\SASApp\WorkspaceServer\Logs\Audit.Library_serv
er_%d_%S{hostname}_%S{pid}.log"/>
  <layout>
    <param name="ConversionPattern"
      value="DateTime=%d Userid=%u Libref=%E{Audit.Dataset.Libref}
          Engine=%E{Audit.Dataset.Engine}
Member=%E{Audit.Dataset.Member}
          MemberType=%E{Audit.Dataset.Memtype}
          OpenMode=%E{Audit.Dataset.Openmode}
Path=%E{Audit.Dataset.Path}"/>
    </layout>
  </appender>
```

Auditing

Conversion Patterns

- Conversion patterns on the "layout" tag of the appender provide the format for log messages.

Pattern	Information	Example values
%d	Event date & time	2016-09-09T14:40:28,788
%-5p	Logging level	TRACE
%t	Thread identifier	00034120
%u	User ID	sasdemo@SUKPJH
%c	Logger	Audit.Data.Dataset.Open
%m	Message	Libref=ORIONSRC Engine=BASE Member=PRODUCT_DIM MemberType=DATA OpenMode=INPUT Path=C:\OrionStar\Source Engine=META Member=PRODUCT_DIM

Auditing

Example log

```
DateTime=2016-09-09T14:40:01,919 Userid=sasdemo@SUKPJH Libref=WORK  
Engine=V9 Member=_PRODSAVAIL MemberType=DATA  
OpenMode=OUTPUT Path=C:\Users\sasdemo\AppData\Local\Temp\SAS Temporary  
Files\_TD22320_SUKPJH\_Prc2
```

```
DateTime=2016-09-09T14:40:08,604 Userid=sasdemo@SUKPJH Libref=ORIONSRC  
Engine=BASE Member=CUSTOMER_DIM MemberType=DATA  
OpenMode=INPUT Path=C:\OrionStar\Source
```

```
DateTime=2016-09-09T14:40:08,604 Userid=sasdemo@SUKPJH Libref=ORIONSRC  
Engine=META Member=CUSTOMER_DIM MemberType=DATA  
OpenMode=INPUT Path=C:\OrionStar\Source
```

Auditing

Example code to parse an audit log

```
data auditlog ;
  infile "C:\SAS\Config\Lev1\SASApp\WorkspaceServer\Logs\Audit.Library_server*.log";
  length TimeStamp 8;
  input
  DateTime=      : $100.
  Userid=        : $50.
  Libref=        : $8.
  Engine=        : $20.
  Member=        : $32.
  MemberType=    : $8.
  OpenMode=      : $8.
  Path=          : $200.
  ;
  TimeStamp = input(DateTime,E8601DT19.6);
  format TimeStamp datetime21.;
  drop DateTime;
run;
proc print data= auditlog;
run;
```

Wild cards can be used in the infile specification

SAS Named Input searches for name/value pairs on each line.

Auditing

Resulting SAS data set

Obs	DateTime	Userid	Libref	Engine	Member	MemberType	OpenMode	Path
1	2016-09-09T14:40:01,919	sasdemo@SUKPJH	WORK	V9	_PRODSAVAIL	DATA	OUTPUT	C:\Users\sasdemo\AppData\Local\Temp\SAS Temporary Files_TD22320_SUKPJH_Prc2
2	2016-09-09T14:40:08,604	sasdemo@SUKPJH	ORIONSRC	BASE	CUSTOMER_DIM	DATA	INPUT	C:\OrionStar\Source
3	2016-09-09T14:40:08,604	sasdemo@SUKPJH	ORIONSRC	META	CUSTOMER_DIM	DATA	INPUT	C:\OrionStar\Source
4	2016-09-09T14:40:25,219	sasdemo@SUKPJH	ORIONSRC	BASE	CUSTOMER_ORDERS	DATA	INPUT	C:\OrionStar\Source
5	2016-09-09T14:40:25,219	sasdemo@SUKPJH	ORIONSRC	META	CUSTOMER_ORDERS	DATA	INPUT	C:\OrionStar\Source
6	2016-09-09T14:40:28,788	sasdemo@SUKPJH	ORIONSRC	BASE	PRODUCT_DIM	DATA	INPUT	C:\OrionStar\Source
7	2016-09-09T14:40:28,788	sasdemo@SUKPJH	ORIONSRC	META	PRODUCT_DIM	DATA	INPUT	C:\OrionStar\Source
8	2016-09-09T14:43:41,704	sasdemo@SUKPJH	ORIONSRC	BASE	PRODUCT_DIM	DATA	UPDATE	C:\OrionStar\Source
9	2016-09-09T14:43:41,704	sasdemo@SUKPJH	ORIONSRC	META	PRODUCT_DIM	DATA	UPDATE	C:\OrionStar\Source
10	2016-09-09T14:44:13,352	sasdemo@SUKPJH	ORIONSRC	BASE	PRODUCT_DIM	DATA	INPUT	C:\OrionStar\Source
11	2016-09-09T14:44:13,352	sasdemo@SUKPJH	ORIONSRC	META	PRODUCT_DIM	DATA	INPUT	C:\OrionStar\Source

Auditing References

- SAS(R) 9.4 Logging: Configuration and Programming Reference, Second Edition
- "Audit Messages for SAS Library Access"
- <http://blogs.sas.com/content/sgf/2015/09/30/part-1-auditing-data-access-who-did-what-and-when/>

Appendix:

Complete logconfig.xml

- There is a complete working logconfig.xml in the notes for this page, taken from my local Windows install of SAS 9.4M3



**WordPad
Document**