

One Check Box to Happiness: Enabling and Analyzing SAS® LASR™ Analytic Server Logs in SAS® Visual Analytics

Chris Vincent, Western Kentucky University

ABSTRACT

EBI administrators who are new to SAS® Visual Analytics and used to the logging capability of the SAS® OLAP Server might be wondering how they can get their SAS® LASR™ Analytic Server to produce verbose log files. While the SAS LASR Analytic Server logs differ from those produced by the SAS OLAP Server, the SAS LASR Analytic Server log contains information about each request made to LASR tables and can be a great data source for administrators looking to learn more about how their SAS Visual Analytics deployments are being used. This session will discuss how to quickly enable logging for your SAS LASR Analytic Server in SAS Visual Analytics 6.4. You will see what information is available to a SAS administrator in these logs, how they can be parsed into data sets with SAS code, then loaded back into the SAS LASR Analytic Server to create SAS Visual Analytics explorations and reports.

INTRODUCTION

Many SAS administrators are unaware of the SAS LASR Analytic server's logging capability simply because this feature is turned off by default. These administrators may also be unaware of how easy it is to enable SAS LASR Analytic server logging in SAS Visual Analytics 6.4. This paper will detail the process of enabling logging for your SAS LASR Analytic server, look for interesting data points in the log file, and demonstrate a technique for parsing the log file. This paper also features a sample report created in SAS Visual Analytics Designer from a sample log dataset.

ENABLING LOGGING FOR THE SAS LASR ANALYTIC SERVER

To enable logging for your SAS LASR Analytic Server, open up SAS Management Console and expand the Server Manager plugin. Find your desired SAS LASR Analytic server, right click its object in the Server Manager plugin and choose Properties. Once the properties window has appeared, choose the Options tab, then click the Advanced Options button. Once the advanced options window appears, choose the Logging Options tab. From here, change the Enable logging: dropdown to 'Yes'.

Note the directory printed in the Path to log files: text input box. This is where the SAS LASR Analytic server will begin storing your log files. The default path for a Windows deployment is '/tmp/', which translates to the directory in which the SAS LASR Analytic server stores its signature files. By default, this location is: \Lev1\AppData\SASVisualAnalytics\<LASR Analytic server name>\sigfiles. Check the documentation for your deployment version and operating system, as this location varies.

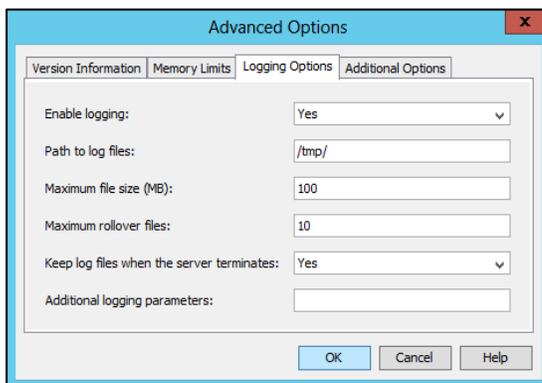


Figure 1. Advanced Options Window in SAS Management Console

STRUCTURE OF SAS LASR ANALYTIC SERVER LOG ENTRIES

SAS EBI administrators familiar with output from the SAS OLAP Server logger will notice that the the log entries output by the SAS LASR Analytic Server are brief, are more general in the information they provide, and are much easier to read at a glance:

```
ID=970,PID=12121,SASTime=1741595103.5,Time=Tue Mar 10 08:25:03
2015,User=ADMIN_USER,Host=VA_2,LASRServer=C:\VA64\Lev1\AppData\SASVisualAnalytics\Dev
LASR\sigfilesV_T_7A35B98F_5827EC2/LASR.2100077534.0.saslasr,Port=21212,RawCmd=action=TableInfo
name=DEV.ENR_LEVEL "Flags=16777248"comment= Dean, WKU ",ExeCmd=action=TableInfo
name= DEV.ENR_LEVEL "Flags=16777248" comment= Dean, WKU",JnlMsg=,StatusMsg=Command
successfully completed.,RunTime= 0.00
```

Table 1 shows the relationships between some of the more interesting variables found in the SAS LASR Analytic Server log and SAS Visual Analytics system metadata.

Log Variable	Description
ID=	Unique ID for this action/log entry
PID=	Process ID for the SAS LASR Analytic Server
SASTime=	DATETIME value of the time at which the action was performed
User=	The ID of the user who uploaded the table being acted upon
Host=	The name of the host on which the SAS LASR Analytic Server resides
LASRServer=	The physical path to the SAS LASR Analytic Server's signature file that currently being accessed by this action
Port=	This SAS LASR Analytic Server's port
RawCmd=	The command issued to the SAS LASR Analytic Server by this action
Name=	The name of the SAS LASR Analytic table currently being accessed, in the format <server tag>.<table name>
Comment=	The name property of the person/identity performing the action on the SAS LASR Analytic server
StatusMsg=	Message returned by the SAS LASR Analytic server. This is the result of the action that was just performed
RunTime=	The action's execution time expressed in seconds

Table 1. SAS LASR Analytic Log Variables and What They Represent

PARSING THE LOG FILE INTO A SAS DATASET

Getting the SAS LASR Analytic Server logs' raw text into Base SAS can be tricky, due to the occasional occurrence of back to back semicolons (;;) in the log text. Once you've processed the raw log text, we can begin the process of extracting the desired variables from the log data.

EXTRACTING LOG VARIABLE VALUES USING PERL REGULAR EXPRESSIONS

One way to identify and capture interesting variables in the SAS LASR Analytic Server logs is to use the SAS Perl regular expression (regex) functions.

The following code shows the structure of a DATA step purposed for identifying and outputting several of the interesting values noted in Table 1:

```

DATA VALUES_CAPTURED (DROP=LOG_ENTRY i REGEX:);
SET LOGFILE_ENTRIES;
IF _N_=1 THEN
DO;
    RETAIN
        REGEX_PID REGEX_ACTION_ID REGEX_LOADED_BY REGEX_ERROR
        REGEX_HOST REGEX_PORT REGEX_ACTION REGEX_TABLE
        REGEX_SERVER_TAG REGEX_USER REGEX_STATUS;

    REGEX_ACTION_ID = PRXPARSE('/ID=(\d+),PID/');
    REGEX_PID = PRXPARSE('/PID=(\d+),/');
    REGEX_ERROR = PRXPARSE('/:ERROR:\s(.+)\.\.,Status/');
    REGEX_LOADED_BY = PRXPARSE('/,User=(. {8}),Host/');
    REGEX_HOST = PRXPARSE('/,Host=(.+),LASRServer=/');
    REGEX_PORT = PRXPARSE('/,Port=(. {5}),RawCmd/');
    REGEX_ACTION = PRXPARSE('/RawCmd=action=(\w+)\s*/');
    REGEX_SERVER_TAG = PRXPARSE('/name=(\w+)\.\.\w+\s/');
    REGEX_TABLE = PRXPARSE('/\sname=\w+\.\.(\w+)\s/');
    REGEX_USER = PRXPARSE('/comment=(.+)",Exe/');
    REGEX_STATUS = PRXPARSE('/StatusMsg=(.+),RunTime=/');

END;

ARRAY EXPRESSIONS {*} REGEX:;
ARRAY OUT_FIELDS {*} $ 512 PID ACTION_ID LOADED_BY_USER ERROR HOST
    PORT ACTION TABLE SERVER_TAG USER STATUS;
DO i=1 to DIM(OUT_FIELDS);
    IF PRXMATCH(EXPRESSIONS{i}, LOG_ENTRY) THEN
        DO;
            OUT_FIELDS{i}= UPCASE(PRXPOSN(EXPRESSIONS{i}, 1, LOG_ENTRY));
        END;
    END;
END;
RUN;

```

The above DATA step code uses the PRXPARSE and PRXMATCH functions to check and match Perl regular expressions, allowing you to identify and output our variables of interest based on their predictable surroundings within the text of the log file.

The above code produces a dataset similar to the one pictured in Figure 2.

PID	ACTION_ID	LOADED_BY_USER	ERROR	HOST	PORT	ACTION	TABLE	SERVE	USER	STATUS
12121	395	LASR_ADMIN		VA-DEV	21212	SUMMARY	ENROLLED_LVL	DEV	DEAN, WKU	COMMAND SUCCESSFULLY COMPLETED.
12121	396	LASR_ADMIN		VA-DEV	21212	TABLEINFO	ENROLLED_LVL	DEV	DEAN, WKU	COMMAND SUCCESSFULLY COMPLETED.
12121	397	LASR_ADMIN		VA-DEV	21212	SUMMARY	ENROLLED_LVL	DEV	DEAN, WKU	COMMAND SUCCESSFULLY COMPLETED.
12121	398	LASR_ADMIN		VA-DEV	21212	TABLEINFO	ENROLLED_LVL	DEV	DEAN, WKU	COMMAND SUCCESSFULLY COMPLETED.
12121	399	LASR_ADMIN		VA-DEV	21212	SUMMARY	ENROLLED_LVL	DEV	DEPT HEAD, WKU	COMMAND SUCCESSFULLY COMPLETED.

Figure 2. A Sample of the Log Dataset

LOADING THE FINISHED DATASET INTO SAS VISUAL ANALYTICS

Once the manicured dataset is written out, it can be loaded into SAS Visual Analytics through the self-service import feature that is accessible to users of the SAS Visual Analytics web applications. Alternatively, the table can be loaded in Base SAS if your SAS LASR Analytic Server library is pre-assigned or manually assigned via a LIBNAME statement using the SASIOLA engine. However, users uploading tables to a SAS LASR Analytic Server in Base SAS will need to run the METALIB procedure to update library and table metadata for their SAS LASR Analytic Server in order for SAS Visual Analytics to recognize a table has been added.

VISUALIZING THE DATA IN YOUR SAS LASR ANALYTIC SERVER LOG

Once you've imported your dataset into the SAS Visual Analytics system, you can employ different controls and report objects to visualize the log data.

Figure 3 shows a sample report produced from the dataset sampled in Figure 2. This report contains several linked report objects, including treemaps (used to show frequency only), a bar chart, and a list table to display detailed data. Importing and formatting the SASTime or Time variables from the log allow you to view system events over time. The report in Figure 3 shows the time report variable, read in from the SASTime log variable, assigned to a horizontal slider. This provides report viewers an adjustable time window through which the log data may be viewed.

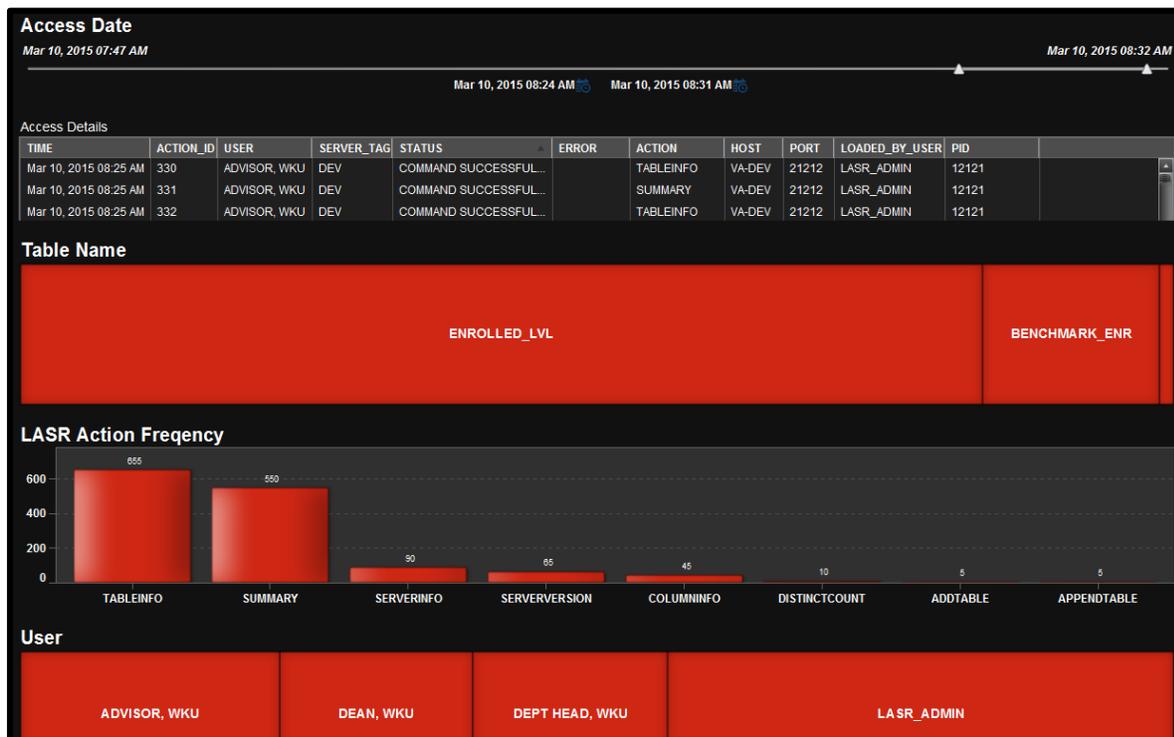


Figure 3. A Report Built in SAS Visual Analytics Designer, Displaying Log Data

CONCLUSION

While the input control used to enable logging for a SAS LASR Analytic Server in SAS Visual Analytics 6.4 is not actually a checkbox, SAS administrators can get information to better understand the use and performance of their SAS Visual Analytics system with just one extra mouse click. With a little extra work, it's possible to incorporate log data into a SAS Visual Analytics report.

RECOMMENDED READING

- SAS Institute Inc. 2014. *SAS® 9.4 Functions and CALL Routines: Reference, Third Edition*. Cary, NC: SAS Institute Inc.
Available at:
<http://support.sas.com/documentation/cdl/en/lefuctionsref/67398/PDF/default/lefuctionsref.pdf>
- SAS Institute Inc. 2014. *SAS® LASR™ Analytic Server 2.4: Reference Guide*. Cary, NC: SAS Institute Inc.
Available at: <http://support.sas.com/documentation/cdl/en/inmsref/67597/PDF/default/inmsref.pdf>
- SAS Institute Inc. 2014. *SAS® Visual Analytics 6.4: User's Guide*. Cary, NC: SAS Institute Inc.
Available at: <http://support.sas.com/documentation/cdl/en/vaug/67270/PDF/default/vaug.pdf>

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author at:

Chris Vincent
Institutional Research
Western Kentucky University
chris.vincent@wku.edu

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.