

Paper 304-2009

The SAS® OLAP Server: Understanding and Solving Common Problems

Adam Budlong, SAS Institute Inc., Cary, NC

ABSTRACT

The SAS® OLAP Server is a complex and powerful tool that is used for storing and exploiting multidimensional data. Because using complex tools can present complex challenges, you need solutions that simplify your use of the SAS OLAP Server and increase operational reliability. This paper provides information intended to empower you with the knowledge and skills required to address these issues with confidence. Strategies and examples for resolving issues in both SAS® 9.1.3 and SAS® 9.2 are presented.

INTRODUCTION

Because the time you spend troubleshooting problems correlates to money spent, knowing how to diagnose a problem can save a company a significant amount of both. This paper equips SAS OLAP Server users with the knowledge to independently find solutions to problems that might arise during regular use of the software. For problems in which SAS Technical Support is involved, knowing where to look for important information and what information to provide to your Technical Support consultant can result in a much faster resolution to your problem.

Advanced SAS OLAP Server users and those familiar with the SAS Customer Support Web site might choose to skip the section "Self Help: Finding Answers through the SAS Customer Support Web Site", and go to the section "Assisted Help: Troubleshooting Problems through SAS OLAP Server Logging". That section is intended to give all SAS OLAP Server users a better understanding of the various log files, how they are used, and why it is helpful to have the logs available when you contact SAS Technical Support for assistance.

SELF HELP: FINDING ANSWERS THROUGH THE SAS CUSTOMER SUPPORT WEB SITE

SAS provides an array of resources to enable you to find answers and resolve problems with your SAS OLAP Server. Among the resources that are available to you, Samples and SAS Notes are an essential source of knowledge for any SAS user. You can find this information through the SAS Customer Support Web site under the **KNOWLEDGE BASE** heading (Display 1).¹

The screenshot shows the SAS Customer Support website interface. At the top left is the SAS logo with the tagline "THE POWER TO KNOW.". Below the logo is a search bar with the text "Search support.sas.com" and a "Search" button. The main navigation bar includes "support.sas.com", "KNOWLEDGE BASE", "SUPPORT", and "LEARNING CENTER". The "KNOWLEDGE BASE" section is expanded, showing a dropdown menu with the following items: "System Requirements", "Install Center", "Product Documentation", "Papers", "Samples & SAS Notes" (highlighted), "Focus Areas", "SUPPORT", "License Assistance", "Administrator Services", "Downloads & Hot Fixes", "LEARNING CENTER", "Bookstore", "Training", "Certification", "SAS Learning Edition", "Higher Education Resources", "SAS OnDemand For Academics", "COMMUNITY", "User Groups", "Events", and "E-Newsletters". On the left side of the page, there is a sidebar with a "KNOWLEDGE BASE" heading and a list of links: "System Requirements", "Install Center", "Product Documentation", "Papers", "Samples & SAS Notes", "Focus Areas", "SUPPORT", "Submit a Problem", "Check Problem Status", and "License Assistance". The main content area features a "Get certified at SAS Global" banner with a "REGISTER" button and a "Introducing the latest SAS blog: The Text Fro" section.

Display 1. Selecting the **samples & SAS Notes** category on the SAS Customer Support Web site

¹ support.sas.com

Samples and SAS Notes include suggestions, tips, and examples for using SAS software and solutions. SAS Notes also provide you with the latest information on a problem or error message that might be encountered while using the software.

Knowing how to effectively search the existing SAS Notes can save time when a problem or question arises. For example, using the keyword search is an efficient way to locate information about an error message. When you are searching for an error message, be sure to remove any data-specific words, such as cube names, user accounts, or paths. If too many results are displayed, you can use double quotation marks around segments of the error message. For example, suppose you receive an error message that is similar to the following:

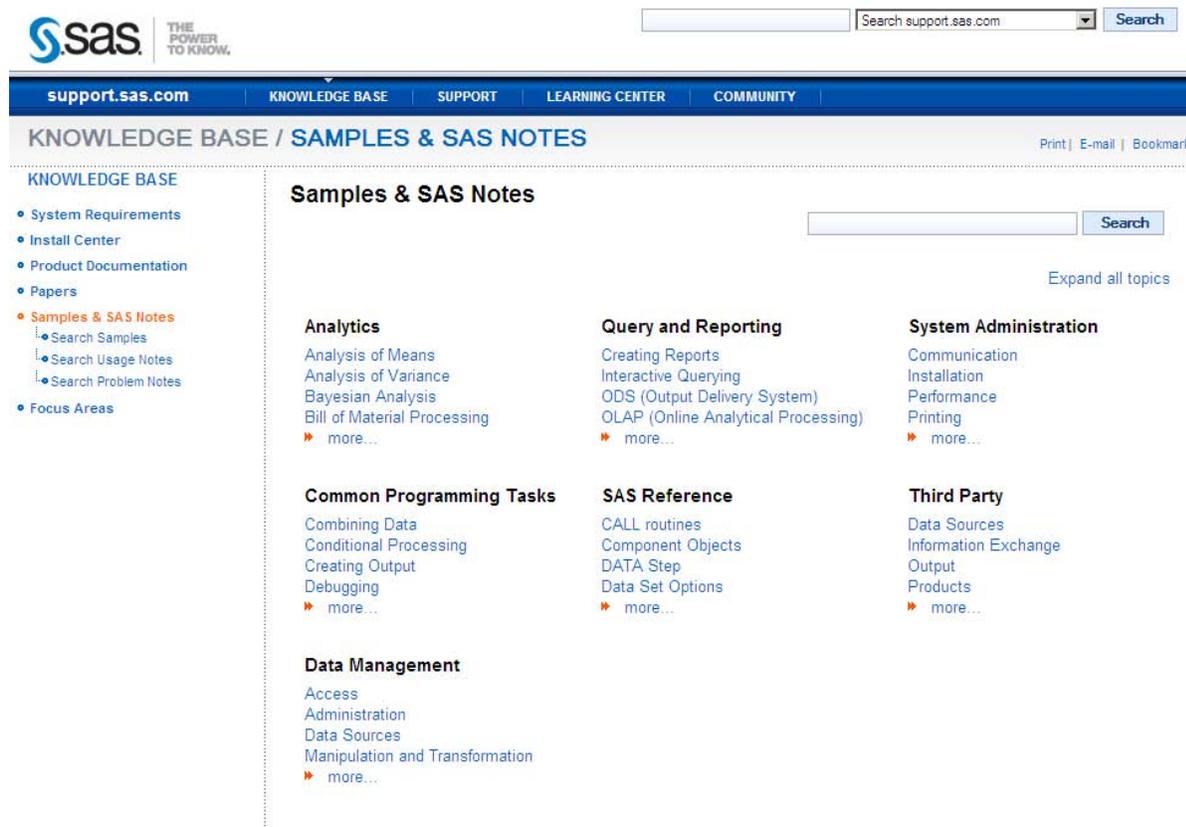
```
SAS Enterprise Guide cannot find the data file: OrionStar-Detail
for (Europe, All YQM)
```

The best combination of keywords for your search might be "SAS Enterprise Guide" "cannot find the data file", as shown in Display 2.



Display 2. Sample keyword search using double quotation marks

An alternative to locating Samples and SAS Notes with the Keyword search is to browse the Samples and SAS Notes topic areas that are shown in Display 3.



Display 3. Samples and SAS Notes topics on the SAS Customer Support site

You can find Samples and SAS Notes about Online Analytic Processing (OLAP) under the **Query and Reporting** heading. Selecting **OLAP (Online Analytic Processing)** under **Query and Reporting**

returns a results page that lists the Samples and SAS Notes for this topic. You can further refine your search by using the filter options that are available on the results page.

SAS Note 34425: "Commonly used SAS Notes for the SAS® 9.1 OLAP Server" (SAS Institute Inc. 2009a) contains a collection of notes that SAS Technical Support frequently provides to our customers who use the SAS 9.1.3 OLAP Server. A similar collection for the SAS 9.2 OLAP Server is documented in SAS Note 34426: "Commonly used SAS Notes for the SAS® 9.2 OLAP Server" (SAS Institute Inc. 2009b). The content of these collections is updated when new SAS Notes are available for the topics.

ASSISTED HELP: TROUBLESHOOTING PROBLEMS THROUGH SAS® OLAP SERVER LOGGING

SAS provides real-time support when you are unable to address a problem successfully on your own or you have a question that is not answered through self-help resources. When you contact SAS Technical Support to help you resolve problems with your SAS OLAP Server, providing information from the SAS OLAP Server logs can result in a quicker resolution to your problem.

SAS OLAP Server logging includes three components: SAS logs, Application Response Measurement (ARM) logs, and journal files. The SAS OLAP Server logs contain information such as library assignments and connection details when using the default options. Increasing the logging level as described below provides information on the communication with the SAS Metadata Server and the MDX queries that are being received. This higher level of logging also provides more detailed error messages than the default options. ARM logs contain information about query performance and what aggregations can improve cube performance. These logs can be used within SAS® OLAP Cube Studio and SAS® Data Integration Studio to determine which aggregations might be needed to improve the performance of your OLAP cube. Journal files are useful for debugging an OLAP query. The journal files are created when one or more of the debugging options is enabled in the Advanced Options dialog box of the SAS OLAP Server. A detailed discussion of these three SAS OLAP Server logging components follows.

SAS LOGS

By default, the SAS logs contain only minimal information such as library assignments and connection details as shown in the following output:

```
20090117:14.06.11.65: 00000308:          New client connection (33) accepted from
server port 5451 for user sasdemo@d15597. Encryption level is Credentials using
encryption algorithm SASPROPRIETARY. Peer IP address and port are
10.11.11.86:3317.
20090117:14.06.17.10: 00000488:          Client connection (33) closed.
```

The decreased logging levels reduce the amount of resources needed by the server. It is possible to increase the amount of logging for the server if problems are encountered.

The log directory for the SAS 9.1.3 release is typically found in the main directory of your SAS OLAP Server configuration. On Microsoft Windows machines, the directory path is as follows:

```
C:\SAS\<your-project-name>\Levl\<your-application-server-name>\OLAPServer\logs
```

To enable additional logging in this release, the configuration file must be updated. When additional details are required in the SAS log, add the setting IOMLEVEL=1. SAS Note 30454: "Setting additional debug options for the SAS® OLAP Server" (SAS Institute Inc. 2009c) provides detailed instructions on enabling this option. Other logging options are available for the SAS OLAP Server, but are not generally required. The IOMLEVEL=1 option instructs the SAS OLAP Server that the logging should contain information about the Integrated Object Model (IOM)² calls that are being made, the MDX queries that are being executed, and any errors that are encountered while processing (see the following sample output).

```
20090117:13.56.34.92: 00000382:          33:sasdemo@d15597: IOM ENTRY {comp:39d4130,
compRef:3b985d0}->ActionSetDeleteFunc()
20090117:13.56.34.92: 00000382:          33:sasdemo@d15597: IOM RETURN
0={comp:39d4130, compRef:3b985d0}->ActionSetDeleteFunc()
20090117:13.56.34.92: 00000382:          33:sasdemo@d15597: IOM RETURN -
1={compRef:2764110}->Session::ActionMetadata() : text=<?xml version="1.0"
```

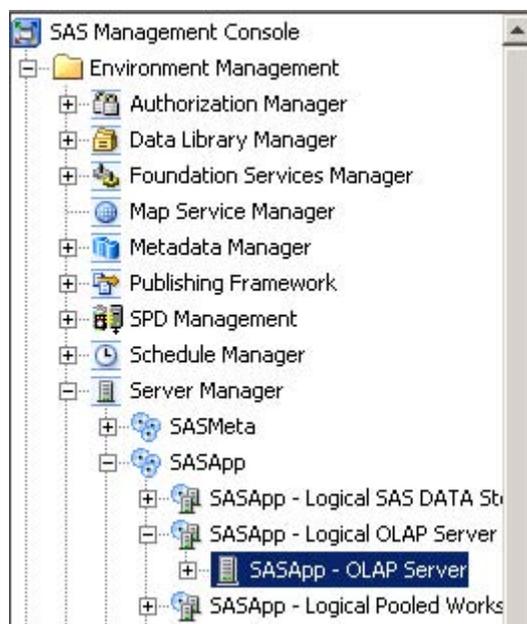
² The Integrated Object Model provides distributed object interfaces to SAS software features. IOM enables you to use industry-standard languages, programming tools, and communication protocols to develop client programs that access these services on IOM servers. The IOM Bridge communications protocol enables diverse clients to connect transparently to IOM servers on multiple platforms.

```
?><Exceptions><Exception><SASMessage severity="Error">The set is
empty</SASMessage></Exception></Exceptions>
20090117:13.56.34.92: 00000386:          33:sasdemo@d15597: IOM CALL
{compRef:2764110}->Session::MDXQuery():  queryStmt=SELECT
  { CrossJoin ( { [Time].[YQM].DefaultMember.Level.Members } , {
[Measures].[QuantitySUM] } ) } ON COLUMNS ,
  {
[Geography].[Global].DefaultMember.Level.Members } ON ROWS
FROM
```

The option should not remain enabled in a production environment because it can cause slower performance and will create very large log files.

With SAS 9.2, you can enable and disable comparable logging with the Log4SAS technology that is available in SAS® Management Console. Unlike SAS 9.1.3, the options set in the server's **Loggers** tab are recognized immediately by the server process. This means that the debugging log can be captured without restarting the server. To obtain similar information in your SAS OLAP Server log file, you must enable two loggers.

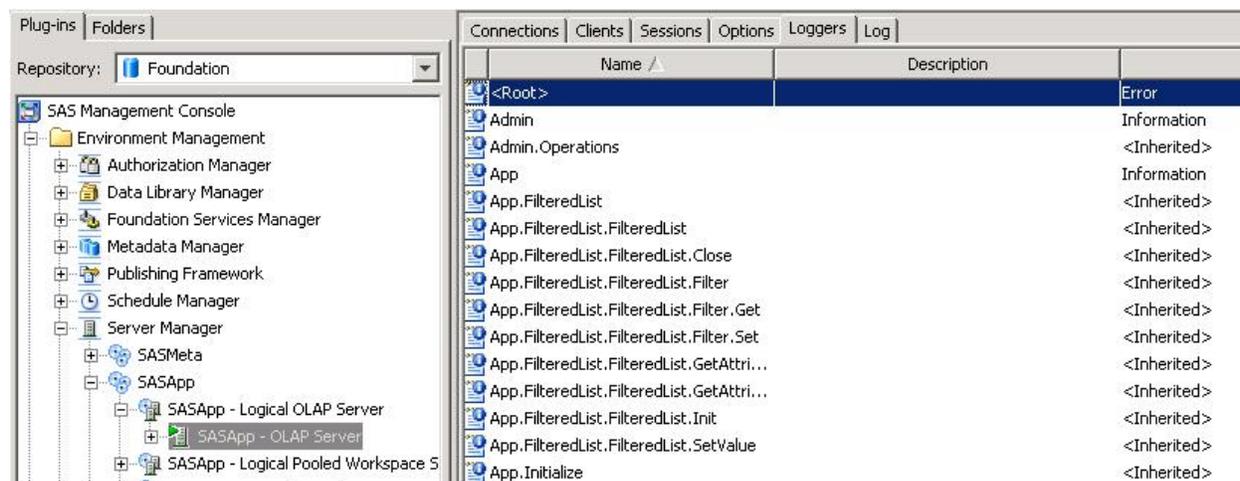
Enabling the loggers is accomplished using the SAS Server Manager Plug-In. The plug-in is available only to users who are granted ReadMetadata permission to the SAS Server Manager Plug-In. Locate the SAS OLAP Server definition in the SAS Server Manager and right-click that definition, as shown in Display 4.



Display 4. Locating the OLAP Server definition

From the context menu, select **Connect**. The active user will connect to the SAS OLAP Server if it is running. To modify the loggers, you must have WriteMetadata permission to the SAS OLAP Server object.

The **Loggers** tab is available after you connect. This tab displays all available loggers for the application server that you selected. Details on the loggers are planned to be available from the SAS Customer Support Web site following the release of SAS 9.2 Phase 2.



Display 5. Locating the Loggers Tab

To most closely replicate the IOMLEVEL=1 logging from SAS 9.1.3, you must set the App logger to a level of either **Debug** or **Trace**. The levels for the logger are outlined in Table 1.

Logger	Default Level	Debugging Level	Result
App	Information	Debug	Prints the IOM calls from the SAS OLAP Server and the responses
App	Information	Trace	Prints the IOM calls from the SAS OLAP Server and the responses, along with any parameters used
App.Session.Session.MDXQuery*	<Inherited>	Trace	Prints the MDX query string**

* SAS Management Console only displays loggers for objects that were already initialized. The App.Session.Session.MDXQuery object will not exist until at least one query is received by the SAS OLAP Server.

** The query string might be truncated. It is possible to increase the string limit by modifying the -OBJECTSERVERPARMS options JNLSTRMAX and JNLLINEMAX, as described in SAS Note 30454 (SAS Institute Inc. 2009c).

Table 1. Settings for the App Logger

The SAS OLAP Server folder also contains a sample logconfig.trace.xml file that can be used when additional logging is needed. Simply rename the existing logconfig.xml file found in the directory and replace it with the trace version. This requires that you restart your SAS OLAP Server process before the additional logging is performed.

To include the MDX Query as part of the trace XML file, add the following lines:

```
<logger name="App.Session.Session.MDXQuery" additivity="true">
  <level value="TRACE"/>
</logger>
```

ARM LOGS

Using ARM logs for the SAS OLAP Server can provide information about which queries are performing well, and what additional aggregations might be required to accelerate the cube performance.

After the options are set, as described below, the ARM log will contain information about who made the request, what cube was used, and what aggregations were found to answer the query. Timing for the query and the amount of data retrieved are documented as part of the record.

You can find an explanation of the values that are in an ARM record in these resources:

1. SAS® 9.1.3 Language Reference: Dictionary (SAS Institute Inc. 2007)
2. SAS® 9.2 Language Reference: Dictionary (SAS Institute Inc. 2008)

In both resources, the ARM_OLAP_SESSION subsystem outlines each record type and provides a description of each field.

In SAS 9.1.3, ARM logging required you to restart a server after enabling the ARM options. Instructions on enabling ARM logging are provided in SAS Note 33495, "Enabling the ARM log in a SAS® OLAP Server" (SAS Institute Inc. 2009d). ARM logs can be evaluated manually using the description of the ARM_OLAP_SESSION. The preferred approach is to use the Aggregation Tuning plug-ins found in both SAS OLAP Cube Studio and SAS Data Integration Studio.

With SAS 9.2, there are two options to enable ARM logging. The first is to use the loggers found in SAS Management Console. The settings are shown in Table 2.

Logger	Default Level	Debugging Level	Result
Perf.ARM.OLAP_SERVER	Warning	Information	Prints the ARM records required for tuning

Table 2. Logger settings in SAS Management Console

Using the loggers enables ARM logging immediately without requiring that you restart a server. The settings persist until the next time that the SAS OLAP Server is restarted.

The second option creates a persistent change to the ARM logging, but it does require you to restart the SAS OLAP Server to initialize those changes. To change the default setting of your ARM log for the server, locate the following lines in your logconfig.xml file, which is found in the configuration directory of the SAS OLAP Server. An example of the file location for a Microsoft Windows environment is `C:\SAS\<your-project-name>\Lev1\<your-application-server-name>\OLAPServer\logconfig.xml`. In this file you will find these XML lines:

```
<logger name="Perf.ARM.OLAP_SERVER" additivity="true">
  <level value="WARN"/>
  <appender-ref ref="ARM"/>
</logger>
```

By changing the level value, as shown in the following example, you can create a permanent change to the ARM logging for your server:

```
<logger name="Perf.ARM.OLAP_SERVER" additivity="true">
  <level value="INFO"/>
  <appender-ref ref="ARM"/>
</logger>
```

The SAS 9.2 ARM logs no longer contain an option to print the MDX string, but you can include this option in the SAS log by using the App.Session.Session.MDXQuery logger described earlier.

JOURNALS

Journal files are useful tools for debugging an OLAP query. These files are created when one or more of the debugging options is enabled in the Advanced Options dialog box of the SAS OLAP Server. You should be careful when you enable the debugging options because certain combinations can cause problems during execution. Except for those options outlined in the following section, you should not enable additional debugging options without explicit instructions from SAS Technical Support.

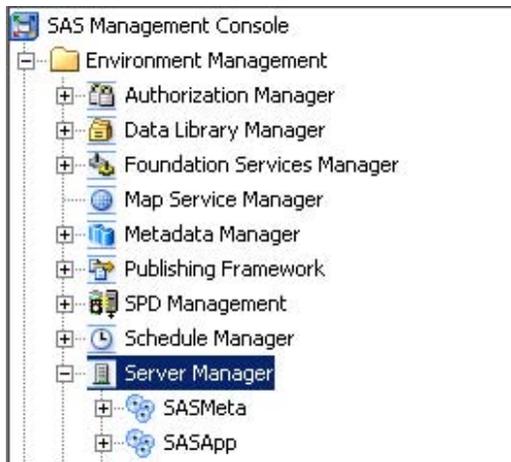
You access the Advanced Options dialog box in both SAS 9.1.3 and SAS 9.2 using the same series of steps in SAS Management Console.

1. Expand the Server Manager.

SAS 9.1.3

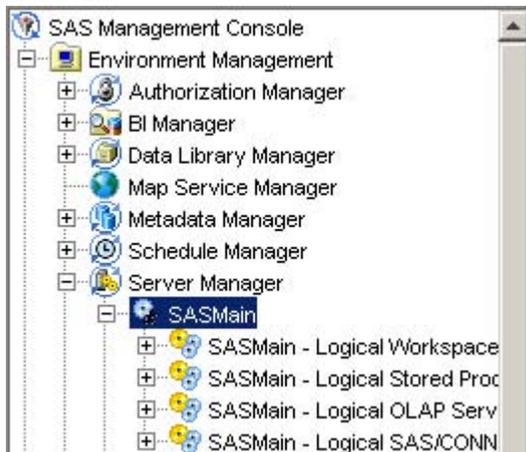


SAS 9.2

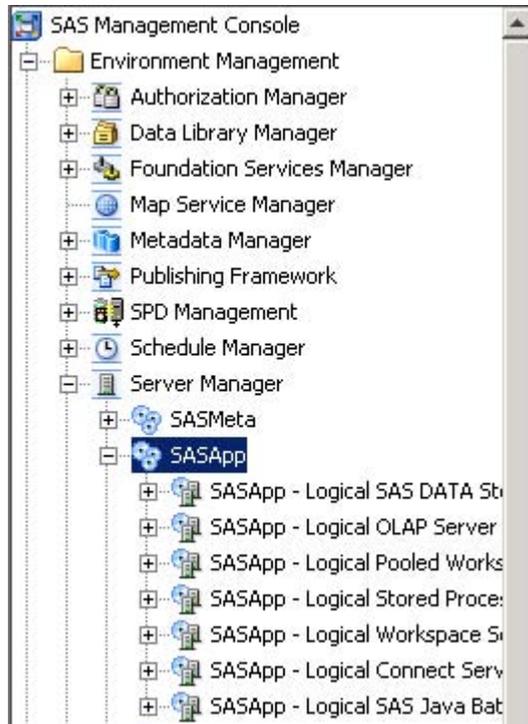


2. Expand your application server (for example, SASMain or SASApp).

SAS 9.1.3

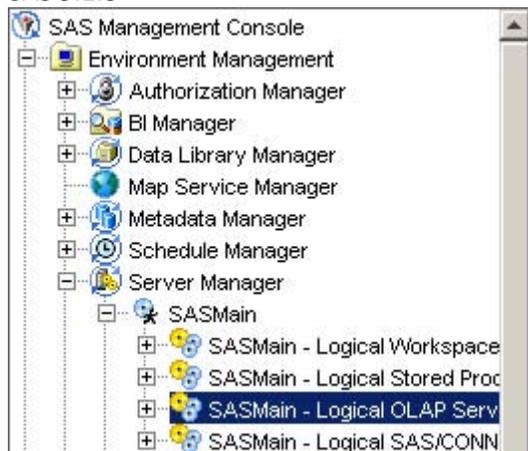


SAS 9.2

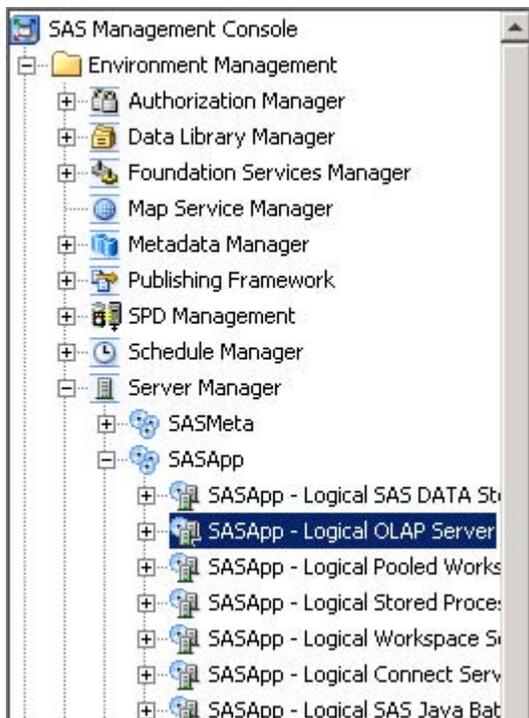


3. Select the logical OLAP Server definition.

SAS 9.1.3

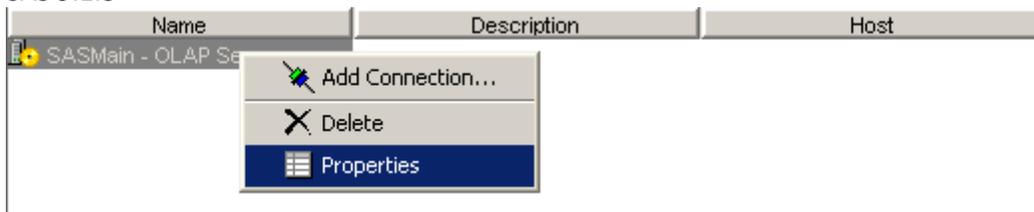


SAS 9.2

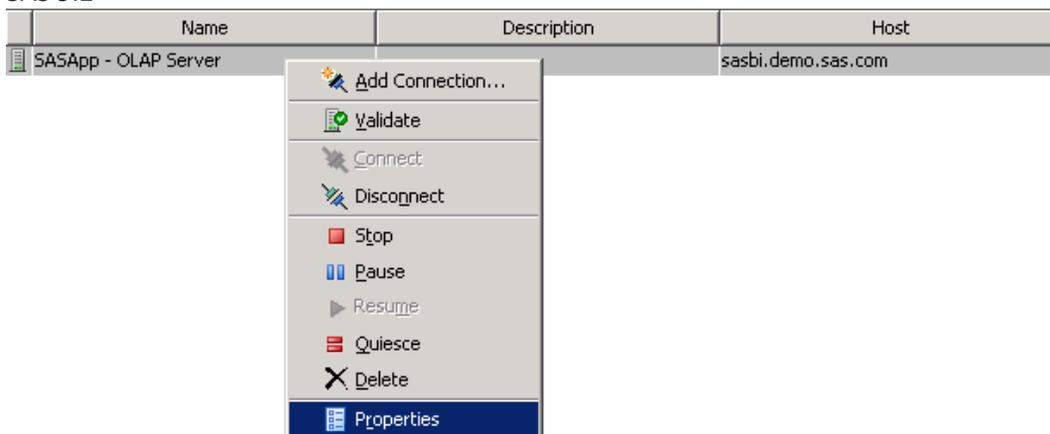


4. In the right pane of SAS Management Console, right-click the OLAP server definition. Select **Properties**.

SAS 9.1.3

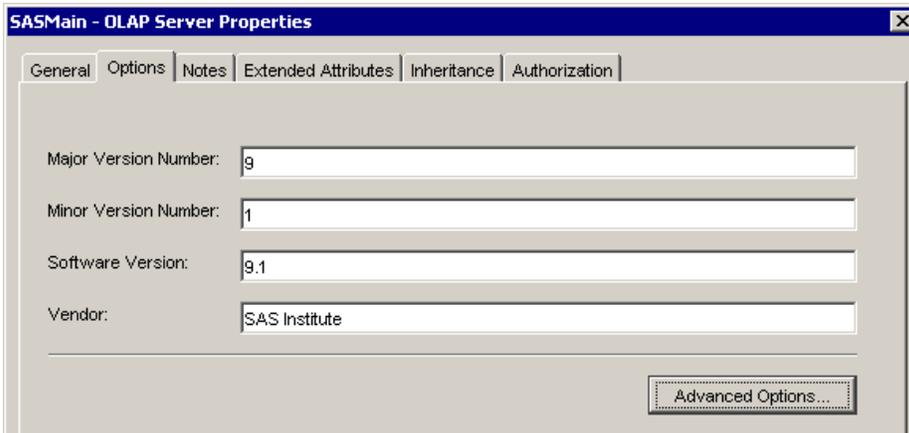


SAS 9.2

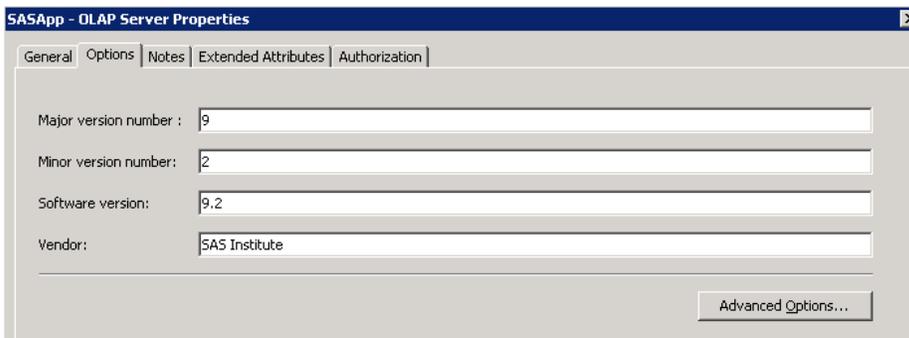


5. Select the **Options** tab to display the properties dialog box, as shown here:

SAS 9.1.3

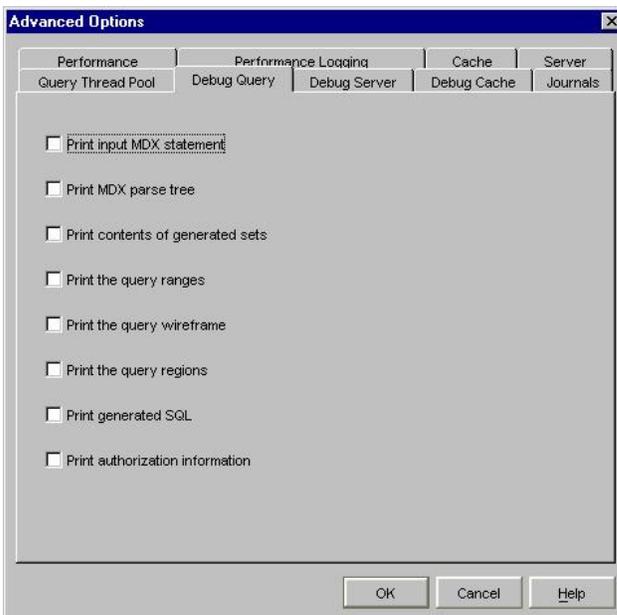


SAS 9.2

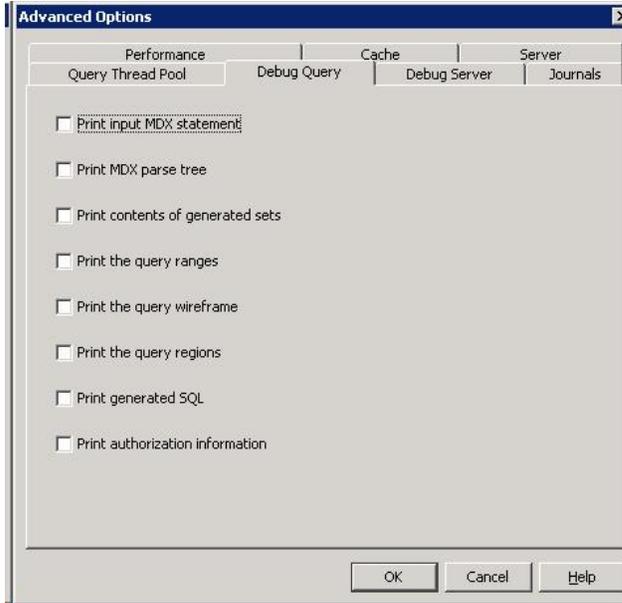


6. Click **Advanced Options** to display the Advanced Options dialog box, as shown here:

SAS 9.1.3



SAS 9.2



The Advanced Options dialog box contains several tabs, which include the debugging tools. The most commonly used options are found on the **Debug Query** tab. Table 3 lists these options and a brief description of what the option causes the server to print in the journal files.

Option Name	Tab	Result
Print input MDX statement	Debug Query	Prints the MDX query along with the ARM transaction ID to the journal file.
Print generated SQL	Debug Query	Prints the SQL query generated for external aggregation tables and drill-through tables.
Print authorization information	Debug Query	Prints the authorization state for the query including metadata object permissions and member-level permission conditions.

Table 3. Debug Query tab options and descriptions

Journal files are always written to the starting directory of the SAS OLAP Server process. In most cases, this is the application server directory above your SAS OLAP Server configuration. The files have the naming convention of SSNJ##### where ##### is a sequence of numbers that indicate the order in which the file was created (starting with 0).

One benefit to including the MDX query string in the journal is that it can be matched easily to a record in the ARM file. Each session that sends a query to the SAS OLAP Server generates, in its journal, a string that contains the MDX query along with an ARM transaction number, as shown in the following sample output:

```
20090107:10.38.32.32: NOTE: MDX statement [ARM Trans 2]: SELECT
CROSSJOIN({[TIME].[ALL TIME].[1993].CHILDREN}, {[MEASURES].[ACTUALSUM],
[MEASURES].[PREDICTSUM]}) ON COLUMNS, {[GEOGRAPHY].[ALL GEOGRAPHY].CHILDREN} ON ROWS
FROM [SALESCUBE]
```

The ARM log contains a series of records that correspond to this transaction ID (which are highlighted in the following sample output) and any records for information that is related to this transaction.

```

C,1546961912.320000,2,2,2,1,1,0.328125,1.171875
C,1546961912.491000,2,3,3,2,2,0.343750,1.218750
U,1546961912.804000,2,1,1,0.343750,1.218750,2,SalesCube           1Geography           3
U,1546961912.804000,2,1,1,0.343750,1.218750,2,SalesCube           2Products           2
U,1546961912.804000,2,1,1,0.343750,1.218750,2,SalesCube           3Time           3
U,1546961912.804000,2,3,3,0.343750,1.218750,2,SalesCube           1 3 1 3 2 2 3 3
U,1546961912.804000,2,3,3,0.343750,1.218750,2,SalesCube           2 2 3 2 1 1
P,1546961912.804000,2,3,3,0.343750,1.218750,0,2,1,144,3,0,SalesCube
P,1546961912.804000,2,2,2,0.343750,1.218750,0,24,SalesCube

```

Using the transaction ID can help to quickly associate an MDX query with the set of information in the ARM log.

CONCLUSION

Knowing how to locate answers to the more commonly encountered problems with your SAS OLAP Server enables you to independently and efficiently solve problems that arise during regular SAS OLAP Server usage. In cases where you need further, knowing which logging files provide pertinent information can help your SAS Technical Support team bring a timely and satisfactory resolution to your SAS OLAP Server problems.

RECOMMENDED READING

SAS Institute Inc. 2009a. SAS Note 34425: "Commonly used SAS Notes for the SAS® 9.1 OLAP Server". Available at support.sas.com/kb/34/425.html.

SAS Institute Inc. 2009b. SAS Note 34426: "Commonly used SAS Notes for the SAS® 9.2 OLAP Server". Available at support.sas.com/kb/34/426.html.

SAS Institute Inc. 2009c. SAS Note 30454: "Setting additional debug options for the SAS® OLAP Server" Available at support.sas.com/kb/30/454.html.

SAS Institute Inc. 2009d. SAS OLAP Server 9.1.3. Product Documentation Web site. Available at support.sas.com/documentation/onlinedoc/olap/index.html.

SAS Institute Inc. 2009e. SAS Technical Papers for SAS OLAP Server and SAS OLAP Cube Studio Web site. Available at support.sas.com/resources/papers/tnote/tnote_olap.html.

SAS Institute Inc. 2009f. SAS Training Web site (various courses for the SAS OLAP Server). Available at support.sas.com/ctx/trnfilter/search?searchString=olap.

SAS Institute Inc. 2008. "ARMSUBSYS= System Option." *SAS® 9.2 Language Reference: Dictionary*, Cary, NC: SAS Institute Inc. Available at support.sas.com/documentation/cdl/en/lrdict/59540/HTML/default/a002165157.htm#a002214658.

SAS Institute Inc. 2007. "ARMSUBSYS= System Option." *SAS® 9.1.3 Language Reference: Dictionary*, Cary, NC: SAS Institute Inc. Available at support.sas.com/onlinedoc/913/docMainpage.jsp?_topic=lrdict.hlp/a002295669.htm.

CONTACT INFORMATION

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

Adam Budlong
 SAS Institute Inc.
 SAS Campus Drive
 Cary, NC 27513
 Phone: 919-677-8008
 Fax: 919-531-9449
 E-mail: support@sas.com
 Web: support.sas.com

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.