The OLAPCONTENTS Procedure
Shine the light onto your OLAP Cubes
Jerry Copperthwaite, SAS Institute, Cary, NC

Introduction

Prior to SAS® 9.4, seeing the layout of your OLAP cube, its dimensions, hierarchies, and levels, was not a simple process. Yes, the PROC OLAP step does return messages stating the cube was successfully created, but how can you determine the actual structure and layout of an existing cube? How can you determine if it had been created as specified? New to SAS 9.4, the OLAPCONTENTS procedure gives you a clear picture of your OLAP cube structure. Similar as PROC CONTENTS is for relational data, PROC OLAPCONTENTS will shine a light on what your OLAP cubes really look like. This paper explores how easy it is to use PROC OLAPCONTENTS, and the benefits this new SAS procedure provides to OLAP cube management.

Knowing the Contents of a Cube

Business Intelligence (BI) reporting is an important part of every business, and there are several reasons why you would need to know the contents of a SAS® OLAP Cube. Consider the following examples:

- Quickly find cubes with the information you need to write a report
- Compare the layout of two or more cubes
- Verify how report results were generated, based on the cube layout
- Diagnose a reporting problem based on the layout of information present in the cube

The OLAPCONTENTS procedure is a valuable tool to help catalog your company's wealth of OLAP information.   Since most SAS users are familiar with the CONTENTS procedure, the output from OLAPCONTENTS was purposefully formatted to resemble the CONTENTS procedure output.

Here is an example of OLAPCONTENTS output that provides a summary of a cube named SIMBA. As you can see, the cube has 5 dimensions, 6 hierarchies and 17 levels. There are also 20 measures defined in the cube as well as 24 properties:

The SAS System

The OLAPCONTENTS Procedure

Cube Name	SIMBA		Dimensions	5
Cube Description	SIMBA: XMLA Council Conformance Suite Data - Multiple Hier		Hierarchies	6
Cube Fact Table	OLAPSIO.SIMBA		Levels	17
Cube Drillthrough Table	OLAPDNT.SIMBA		Properties	24
			Calculated Members	0
			Measures	20
			Named Sets	0

Before OLAPCONTENTS, this kind of information was obtained either from a saved SASLog output from the OLAP procedure, or from manually tabulating the output from SAS® OLAP Cube Studio. Both of these methods are tedious and error prone, and provide output that is not in a user friendly format.

Using the OLAPCONTENTS Procedure

The OLAPCONTENTS procedure gives users the ability to examine information about the cubes defined on an OLAP server. The output from this procedure contains information about the tables used to load the cube, the layout of dimensions hierarchies, levels, properties, and the cube measures.

The syntax for using the OLAPCONTENTS Procedure is straightforward.
- CONNECT to a running OLAP server
- DISPLAY the cube(s) information
- DISCONNECT from the OLAP server.

Here is the syntax:

> **PROC OLAPCONTENTS** [<*connection-options*>];
> **CONNECT** <*connection options*>;
> **DISPLAY CUBE** *cube-name* [OUT=*data-set*] <*display-options*>;
> **DISCONNECT;**
> **RUN**;

CONNECT Statement

A connection to a running SAS® OLAP Server is established by providing connection information on the PROC OLAPCONTENTS statement or by running the CONNECT command. The HOST= option specifies the host name or network IP address of the computer hosting the OLAP Server. When the OLAP Server and the SAS session are running on the same computer, the value "localhost" may be used. The PORT= option specifies the TCP port the SAS OLAP Server is using. The USERID= and PASSWORD= options specify the valid userid and password information, respectively, used to connect to the OLAP Server. It

is worth mentioning that the OLAP Server must be running and referencing the cube(s) which you wish to have information displayed.

DISPLAY CUBE Statement
This statement allows you to display information for the dimensions, hierarchies, and levels for a cube. The only requirement of this statement is that you must specify a cube name.

DISPLAY CUBE cubename

By default, all available information for a cube is displayed. However, since the default cube report can be lengthy, you may want to show less information about each individual cube. You can refine the report and focus on specific areas of the cube information. Utilize the various DISPLAY options to filter out parts of the information you wish to exclude. For instance, if you had no interest in the calculated members in a cube, you could use the NOCALC (or NOCALUCLATED) option.

Here are the options for the DISPLAY CUBE statement:

> **nodimension | nodim**
> **nohierarchy | nohier**
> **nolevel | nolev**
> **noproperty| noprop**
> **nocalculated| nocalc**
> **nomeasure | nomeas**
> **noset**
> **summaryonly**

As an example, here is the syntax to DISPLAY only the measures information for the SIMBA cube. Issue the DISPLAY command after connecting to an OLAP Server:

DISPLAY CUBE SIMBA NODIM NOHIER NOLEV NOSET NOPROP NOCALC;

The SAS System

The OLAPCONTENTS Procedure

Cube Name	SIMBA	Dimensions	5
Cube Description	SIMBA: XMLA Council Conformance Suite Data - Multiple Hier	Hierarchies	6
Cube Fact Table	OLAPSIO.SIMBA	Levels	17
Cube Drillthrough Table	OLAPDNT.SIMBA	Properties	24
		Calculated Members	0
		Measures	20
		Named Sets	0

List of MEASURES				
Measure Name	**Measure Statistic**	**Measure Caption**	**Measure Description**	**Measure Expression**
COGS	SUM	Sum of COGS	Cost of Goods Sold	
Cost of Goods Sold	CALCULATION	Cost of Goods Sold	Cost of Goods Sold	Aggregate([measures].[COGS]), GROUP='[Year To Date]'
MARGIN	CALCULATION	MARGIN	MARGIN	[measures].[sales] - [measures].[cogs],format_string='dollar16.2'
MARGIN_PERCENT	CALCULATION	MARGIN_PERCENT	MARGIN_PERCENT	([measures].[margin] / [measures].[sales]) * 100, format_string='8.2'
MARKETING_EXPENSE	SUM	Sum of MARKETINGEXPENSE	Marketing Expense	
MISC_EXPENSE	SUM	Sum of MISCEXPENSE	Misc Expense	
nu_MONTH		Number of Unique Values for month in TIME	Number of Unique Values for month in TIME	
nu_populationgroup		Number of Unique Values for populationgroup in POPULATION	Number of Unique Values for populationgroup in POPULATION	
nu_QUARTER		Number of Unique Values for quarter in TIME	Number of Unique Values for quarter in TIME	
nu_region		Number of Unique Values for region in GEOGRAPHICAL	Number of Unique Values for region in GEOGRAPHICAL	
nu_stategeo		Number of Unique Values for state in geographical	Number of Unique Values for state in geographical	

|				
nu_statepop		Number of Unique Values for state in population	Number of Unique Values for state in population	
nu_year		Number of Unique Values for year in TIME	Number of Unique Values for year in TIME	
PAYROLL_EXPENSE	SUM	Sum of PAYROLLEXPENSE	Payroll Expense	
PROFIT	CALCULATION	PROFIT	PROFIT	[measures].[margin] - [measures].[total_expenses], format_string='dollar20.2'
PROFIT_PERCENT	CALCULATION	PROFIT_PERCENT	PROFIT_PERCENT	([measures].[profit] / [measures].[sales]) * 100, format_string='8.2'
SALES	SUM	Sum of SALES	Sum of SALES	
STOCK_ADDITIONS	SUM	Sum of STOCKADDITIONS	Stock Additions	
Total Sales	CALCULATION	Total Sales	Total Sales	Aggregate([measures].[SALES]), GROUP='[Year To Date]'
TOTAL_EXPENSES	CALCULATION	TOTAL_EXPENSES	TOTAL_EXPENSES	[measures].[marketing_expense] + [measures].[payroll_expense] + [measures].[misc_expense],format_string='dollar20.2'

DISCONNECT Statement

Disconnecting from the current OLAP Server is done with the DISCONNECT command. Once you have disconnected from the server, you can use the CONNECT statement to connect to a different OLAP Server.

When the OLAPCONTENTS procedure is terminated, using the QUIT or the RUN command, an automatic server disconnect is executed. It is not necessary to explicitly disconnect from the server prior to terminating the procedure.

SAS® OLAP Cube Studio

Before OLAPCONTENTS, the only way that you could visually verify the structure of a cube was by using SAS OLAP Cube Studio. A major drawback to this approach is that the wealth of information about the cube is not easily viewable. Here is an example of what is displayed for the SIMBA cube measures when using SAS OLAP Cube Studio:
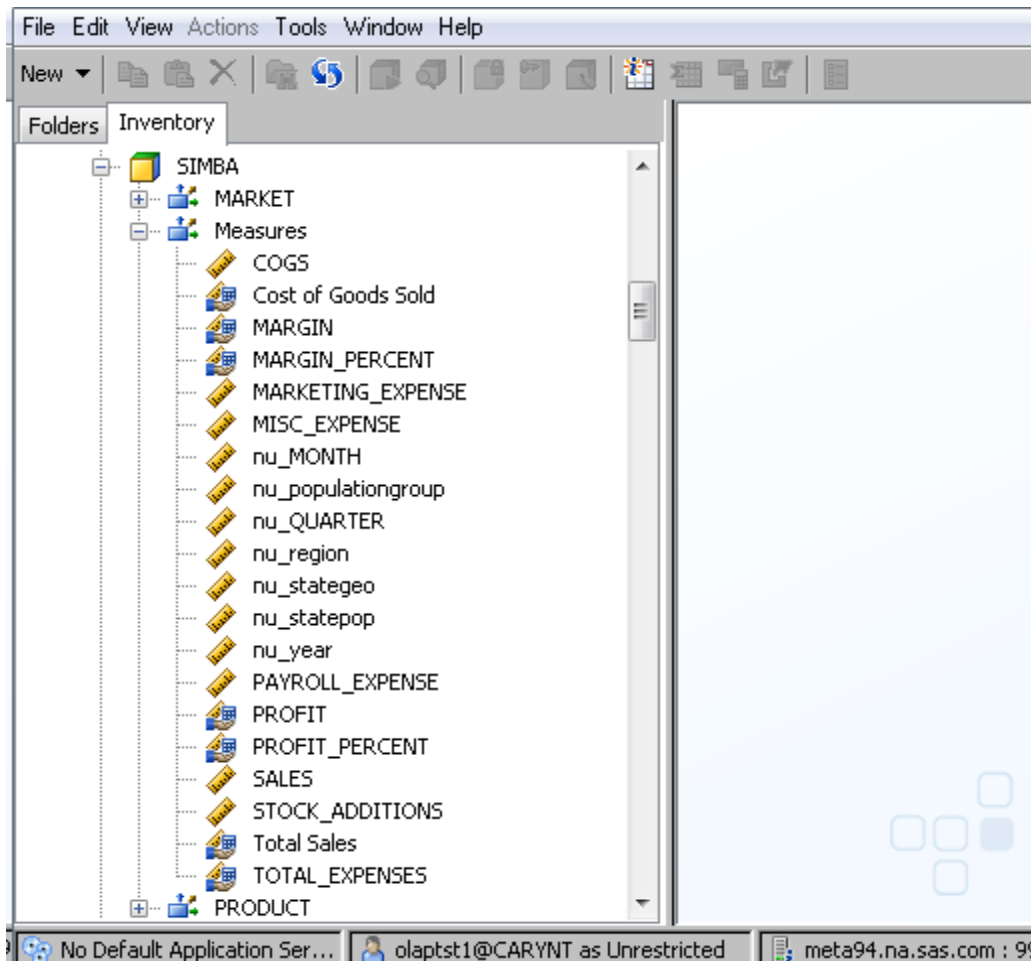
Advanced Features of OLAPCONTENTS

Just as a powerful spotlight can give you a clear and accurate picture of your surroundings at night, OLAPCONTENTS gives you a clear and accurate picture of an OLAP cube. The following advanced features of OLAPCONTENTS are especially helpful in providing this picture:

- The ability to display cube dimension level cardinality
- The ability to capture output of the DISPLAY command in a SAS® Data Set
- Use of the Output Delivery System to format output

Displaying Dimension Level Cardinality
The number of unique members at any cube dimension level is referred to as the cardinality of that dimension level. Having a way to access this number quickly and easily for any dimension level is of great value to users wishing to query a cube. For instance, you can use this information to make more efficient queries against a cube by avoiding crossing two dimension levels with high cardinality (when possible).

Here is an example of OLAPCONTENTS output that includes only the cardinality information for the SIMBA cube. This report was generated by issuing the following DISPLAY command:

DISPLAY CUBE SIMBA NODIM NOHIER NOMEAS NOSET NOPROP NOCALC;

The SAS System

The OLAPCONTENTS Procedure

Cube Name	SIMBA	Dimensions	5
Cube Description	SIMBA: XMLA Council Conformance Suite Data - Multiple Hier	Hierarchies	6
Cube Fact Table	OLAPSIO.SIMBA	Levels	17
Cube Drillthrough Table	OLAPDNT.SIMBA	Properties	24
		Calculated Members	0
		Measures	20
		Named Sets	0

LEVELS for GEOGRAPHICAL Hierarchy

Level	Level Name	Member Cardinality	Level Caption	Level Description
1	All	1	All GEOGRAPHICAL	All GEOGRAPHICAL
2	REGION	4	REGION	REGION
3	STATE	20	STATE	STATE

LEVELS for POPULATION Hierarchy

Level	Level Name	Member Cardinality	Level Caption	Level Description
1	All	1	All POPULATION	All POPULATION
2	POPULATIONGROUP	3	POPULATIONGROUP	POPULATIONGROUP
3	STATE	20	STATE	STATE

LEVELS for PRODUCT Hierarchy

Level	Level Name	Member Cardinality	Level Caption	Level Description
1	All	1	All PRODUCT	All PRODUCT
2	FAMILY	4	FAMILY	FAMILY
3	SKU	13	SKU	SKU

LEVELS for SCENARIO Hierarchy

Level	Level Name	Member Cardinality	Level Caption	Level Description
1	All	1	All SCENARIO	All SCENARIO
2	SCENARIO1	2	SCENARIO1	SCENARIO1

LEVELS for SUPPLIER Hierarchy

Level	Level Name	Member Cardinality	Level Caption	Level Description
1	All	1	All SUPPLIER	All SUPPLIER
2	SUPPLIERNAME	3	SUPPLIERNAME	SUPPLIERNAME

LEVELS for TIME Hierarchy

Level	Level Name	Member Cardinality	Level Caption	Level Description
1	All	1	All TIME	All TIME
2	YEAR	1	YEAR	YEAR
3	QUARTER	4	QUARTER	QUARTER
4	MONTH	12	MONTH	MONTH

Writing DISPLAY Output to a SAS Data Set
As with many other SAS procedures, OLAPCONTENTS allows users to write output to a SAS Data Set. This allows you to manipulate and examine and manipulate the output, as data, using other SAS procedures. Here is an example where OLAPCONTENTS output is written to a SAS Data Set; followed by using the PRINT procedure is used to provide a specific view of the results.

The SAS System

DimensionName	HierarchyName	LevelName
MARKET	GEOGRAPHICAL	All
		REGION
		STATE
	POPULATION	All
		POPULATIONGROUP
		STATE
PRODUCT	PRODUCT	All
		FAMILY
		SKU
SCENARIO	SCENARIO	All
		SCENARIO1
SUPPLIER	SUPPLIER	All
		SUPPLIERNAME
TIME	TIME	All
		YEAR
		QUARTER
		MONTH

This view of the cube information was generated by issuing the following SAS statements:

```
PROC OLAPCONTENTS;
CONNECT HOST="HOSTNAME" USERID="USERNAME" PW="PASSWORD" PORT=5451
PROTOCOL=BRIDGE;
DISPLAY CUBE SIMBA OUT=SIMBADATA NOMEAS NOPROP;
RUN;

PROC PRINT DATA=SIMBADATA NOOBS;
VAR DIMENSIONNAME HIERARCHYNAME LEVELNAME;
RUN;
```

Customizable with the Output Delivery System

The SAS® Output Delivery System (ODS) is a flexible component of SAS which allows users to customize, format, and direct SAS output to a variety of output destinations. Since the OLAPCONTENTS procedure was developed using ODS, users can take full advantage of the capabilities it provides to generate report quality output.

Here is an example where the output of OLAPCONTENTS has been saved to an HTML file and additional text has been added to the output. This is what appears on the web page created by the HTML file:

The SAS System

MY CUBE REPORT
A SUMMARY OF THE SIMBA CUBE

Cube Name	SIMBA	Dimensions	5
Cube Description	SIMBA: XMLA Council Conformance Suite Data - Multiple Hier	Hierarchies	6
Cube Fact Table	OLAPSIO.SIMBA	Levels	17
Cube Drillthrough Table	OLAPDNT.SIMBA	Properties	24
		Calculated Members	0
		Measures	20
		Named Sets	0

The HTML file that created this web page was generated by issuing the following SAS statements:

```
ODS HTML FILE="CUBEREPORT.HTML";
ODS TEXT="MY CUBE REPORT";
ODS TEXT="A SUMMARY OF THE SIMBA CUBE";

PROC OLAPCONTENTS;
CONNECT HOST="HOSTNAME" USERID="USERNAME" PW="PASSWORD" PORT=5451
PROTOCOL=BRIDGE;
DISPLAY CUBE SIMBA SUMMARYONLY;
RUN;

ODS HTML CLOSE;
```

Coming Attractions for OLAPCONTENTS

While the OLAPCONTENTS procedure gives users an extremely valuable tool for discovering the layout of OLAP cubes, the following future additions are also being considered.

1) Display of cube aggregations
2) Display the directory path to the physical cube files

Conclusion

The OLAPCONTENTS procedure is a powerful tool for viewing the layout of OLAP cubes. Having an easy way to explore the contents of cubes keeps the management of OLAP information for report writing user friendly. OLAPCONTENTS is a great way to shine a light on OLAP information that is ready for further analysis.