Hyperion®

64-BIT ANALYTIC SERVICES

TABLE OF CONTENTS

1 PURPOSE
2 ABOUT THE CHIP ARCHITECTURE
2 OPERATING SYSTEMS SUPPORT
2 ADDRESSABILITY
3 SALIENT POINTS
3 PERFORMANCE RESULTS
9 CONCLUSION
10 APPENDIX

As 64-bit hardware becomes mainstreamed into I.T. infrastructures, the ability for software vendors to tackle problems previously out of reach becomes enticing. Software products that rely on larger memory addressability can exploit this hardware to extend their value to organizations and lower the cost of ownership as hardware farms can be consolidated.

PURPOSE

The main purpose of this paper is to focus on the 64-bit version of Hyperion System 9 BI+ Analytic Services[1] offering from Hyperion, and to highlight the salient features that make this a compelling release for customers to consider. This document is intended for readers who have general knowledge of Analytic Services and/or its predecessor, Hyperion Essbase.

Consider the following scenarios

- Allocating a new variance of $70,000 across 250,000 customers and 100 geographies. While this is a complex scenario it can be accomplished in a classic Essbase Analytic Services model.
- Aggregate and analyze spending patterns of 2,500,000 customers across 100 geographies. This is ultimately possible with an Enterprise Analytic Services model.
- Allocate the net variance and analyze the spending patterns of 25,000,000 customers across 1000 geographies.

Unlike the first two scenarios that have obvious solutions, the third one is much more complex. There is no precedence for an Analytic Services model of that scale. Historically speaking, classic Analytic Services models maximize for models with 3 to 4 million members. The recent technological advancements in the Analytic Services product in the form of Aggregate Storage, MDX support, Data Mining capabilities, duplicate name support etc. and its availability on the 64bit Itanium platform as a 64bit application have removed these historical limitations. With the release of Hyperion System 9 BI+ Analytic Services, it is possible to create and analyze much larger OLAP models.

1. Analytic Services was previously referred to as Hyperion Essbase, and now represents a significant advancement upon previous releases.

The first version of 64-bit Analytic Services was released as Hyperion Essbase 6.6.0 and 6.6.1 on the Itanium platform in June 2004. It was based on the 6.5.4 release of Essbase. Hyperion published the APB-1 benchmark using this version, results of which are available at the following link. http://www.hyperion.com/products/benchmarks/hp_itanium.cfm

Subsequently in June 2005, Analytic Services 7.1.4 was released on the 64-bit Itanium platform. Analytic Services 7.1.4 is based on the Analytic Services 7.1.3 release and combined the powers of the 64-bit platform and new Aggregate Storage technology.

The 64-bit functionality of Analytic Services has been carried forward into Hyperion System 9. Hyperion System 9 BI+ Analytic Services 9.0.0 is available on the 64-bit Itanium platform. Analytic Services contains both the Essbase Analytics Module and the Enterprise Analytics Module. Both modules leverage a common multi-dimensional analytic server component, the Analytic Services server.

ABOUT THE CHIP ARCHITECTURE

The Itanium chip was designed by Intel and HP together and the current version available is Itanium 2. The Itanium architecture is based on Explicitly Parallel Instruction Computing (EPIC) and supports highly parallel processing, large memory addressability and innovative compiler-based optimization. The parallel architecture includes a large number of execution registers that enable efficient, simultaneous processing for up to six instructions.

Comprehensive information about the chip architecture etc. can be found at:
http://www.intel.com/products/processor/itanium2/index.htm

OPERATING SYSTEMS SUPPORT

Hyperion System 9 BI+ Analytics Services 9.0.0 64-bit is currently available on Windows Server 2003 and HPUX 11.23 Operating Systems.

Information on Operating System Support and more can be found at:
http://www.intel.com/design/Itanium2/whitepaper/eco.pdf

ADDRESSABILITY

Scalability and performance of multi-dimensional analytic solutions is often directly related to the amount of memory available to these solutions. Perhaps the greatest advantage of 64-bit computing lies in its capability to address more than 4GB of memory per process. This opens up various avenues for increased scalability, compared to processes running in a 32-bit address space.

Various components in the Analytic Services Server have their own memory requirements. Each such component has more memory at its disposal in the 64-bit address space. This translates to increased scalability in all aspects of Analytic Services. For example, if the Analytic Services metadata subsystem has 500 Mega Bytes of space available, then it would be restricted to a certain outline size. Building models which are larger than that is impossible within this limited address space. But with 64-bit addressability the same metadata subsystem would have much more space at its disposal allowing it to scale beyond current limits.

TABLE 1

	32-bit	64-bit	Performance Improvement
Machine Configuration	2x2.8 GHz CPUs, 1GB RAM	4x 1.6 GHz CPUs, 70GB RAM	
ASO Cache Size	256Mb	20Gb	
Dataload time	2782 seconds	1284 seconds	
Aggregation time	1600 seconds	1212 seconds	
Total time	4382 seconds	2496 seconds	1.8 times faster

SALIENT POINTS

The following list highlights the salient points of 64-bit Analytic Services.

- Supports larger models in both Essbase Analytics (Block Storage) and Enterprise Analytics (Aggregate Storage) modes. Analytic models with up to 80 million members have been successfully built in Aggregate Storage on 64-bit versions of the product. For comparison purposes, certain outlines containing up to 15 million members could be built using the 32-bit product. Outline Viewer screen shots of the 80 million member outline may be viewed as an appendix.
- Increased addressability enables various cache sizes to be set beyond 4 Gigabytes, hence increasing chances of completely 'in-memory' databases and improving performance for aggregations, data loads, calculations, concurrent user queries etc.
- Advanced compiler optimization takes advantage of the processor architecture, thus helping most models to perform better out of the box when compared to existing host platforms. The out of the box performance gains vary anywhere from 50% to 300% depending on the model and the type of operation.
- The 64-bit JVM no longer has restricted addressability; enabling clients like Analytic High Availability Services[2] to run significantly larger number of concurrent queries without having user logins timed-out by Analytic Services.
- Block storage calculation parallelism has been extended to 8-way on 64-bit. On the 32-bit platforms it is 4-way.
- The maximum limit on the Analytic Services configuration settings for AgentThreads and ServerThreads has been increased to 1024 for 64-bit platforms, as opposed to 512 for 32-bit platforms. This increases the concurrent user scalability. [Please consider suggested best practices when using these settings.]

2. Formerly Essbase Deployment Services

PERFORMANCE RESULTS

This section presents some empirical results which serve as validation for the observations made in the previous section.

AGGREGATE STORAGE PERFORMANCE

The above table (Table 1) shows results from tests done on an Enterprise Analytics (Aggregate Storage) database with approximately fifty thousand members in the largest dimension in the outline. The database size after aggregation is around 7.5 GB. This demonstrates the power of increased addressability and its effect on performance. The large cache size, which is highly exaggerated for this example, is used for the 64bit test to demonstrate addressability and its effect on performance.

TABLE 2

	32-bit	64-bit	Performance Improvement
Machine Configuration	2x2 GHz CPUs 2 GB RAM	4x1.3 GHz CPUs 16 GB RAM	
ASO Cache Size	500 MB	1 GB	
Dataload1 time	4632.58 seconds	3270 seconds	
Dataload2 time	4904.67 seconds	3451 seconds	
Dataload3 time	5077.99 seconds	3394.8 seconds	
Dataload4 time	5108.91 seconds	3571.42 seconds	
Dataload5 time	5310.72 seconds	3594.33 seconds	
Dataload6 time	5469.84 seconds	3578.69 seconds	
Dataload7 time	5540.09 seconds	3609.41 seconds	
Dataload8 time	5379.25 seconds	3791.93 seconds	
Dataload Buffer commit time	4951.67 seconds	2545.35 seconds	
Aggregation time	35784.5 seconds	16466.7 seconds	
Total time	**82160.22 seconds**	**47363.63 seconds**	**1.7 times faster**
Number of Aggregate Views Created	14	14	
Database Size	55.50 GB	55.50 GB	
Typical Query which filters top 50 entities based on a specific measure	36.125 seconds	27.39 seconds	1.3 times faster

The above table (Table 2) shows results from tests done on a larger Aggregate Storage database with approximately one and a half million members in the largest dimension in the outline. The database size after aggregation is around 55.50GB. This demonstrates the advantages of increased addressability and out-of-the-box performance gains.

ANALYTIC SERVICES AND ANALYTIC HIGH AVAILABILITY SERVICES SCALABILITY

The following table (Table 3) demonstrates the advantage of 64-bit addressability through Analytic High Availability Services. The tests consisted of querying the Sample/Basic database running Analytic services through the Analytic High Availability Services server. The client had a certain number of threads running a certain number of queries each (indicated by iterations in the table). The ad-hoc queries were grid queries which had operations like retrieve, zoom-in, zoom-out, pivot, keep-only, remove-only, etc. The table shows that in the 32-bit environment, no more than 500 client threads could be either created in the JVM or no more than 500 client threads could connect to Analytic Services without getting time-out errors. In the 64-bit environment, simultaneous querying against the Analytic Server was tested for up to 3000 client threads. This demonstrates significant advantages of increased addressability for both the JVM and the Analytic Server.

TABLE 3

Client Threads/ Iterations	Server Threads	Agent Threads	Time Taken 32-bit	Time Taken 64-bit	Comments
For ad-hoc queries					
100/10	Default(20)	Default(5)	04m33.704s	1m11.30s	No time out errors on 64-bit
500 /1	500	500	02m16.032s	1m24.98s	No time out errors on 64-bit
1000/1	Default	Default	—	35m29.84s	No time out errors on 64-bit
2000/10	Default	Default	—	1h14m59.24s	No time out errors on 64-bit
3000/10	Default	Default	—	2h3m10.09s	No time out errors on 64-bit
For MDX queries (relevant for Visual Explorer based analytics)					
100/10	Default	Default	01m50.343s	25.78s	No time out errors on 64-bit
500 /1	500	500	56.281s	23.35s	No time out errors on 64-bit
1000/1	Default	Default	—	1m27.08s	No time out errors on 64-bit
1000/10	Default	Default	—	10m22.42s	No time out errors on 64-bit
2000/10	Default	Default	—	57m40.90s	No time out errors on 64-bit
3000/10	Default	Default	—	2h38m29.20s	No time out errors on 64-bit

METADATA AND QUERY SCALABILITY

The following three tables show results of testing very large Aggregate Storage Option (ASO) databases.

The first table shows the time taken to build ASO databases with 40, 60, and 80 million members. Each outline is based on the *ASOsamp* database that is included with Enterprise Analytics. It has three large dimensions (Products, Stores, Geography) and several other smaller dimensions. The Measures dimension is similar to that in sample database.

The Products dimension has up to 10 levels, with a fan-out (number of children for each member) ranging from 5 to 8. The Stores dimension has up to 9 levels with a fan-out ranging from 7 to 11. The Geography dimension has up to 14 levels, with a fan-out of 4. In the Products and Stores dimensions, up to 10% of the members at each level are leaves, resulting in a ragged hierarchy. Member names in each of these dimensions use alphabetical characters, and the member name lengths vary as follows: Products: 23-27, Stores: 18-22, Geography: 13-17. The average member length is 18.

The dimensions are basic members without any UDAs, aliases or associated attribute dimensions.

The dimension build is done using text files and simple rules file. The text file contained parent-child pairs. All these files resided on the server itself.

Data load and aggregation information is also given. Each data load file contains randomly chosen clusters of members from the above three dimensions. Each cluster of data loaded is generated by picking a combination of leaf members from the non-Measures dimensions. Additional leaf members near (in the outline) the selected members in the Products, Stores, and Geography dimensions are picked to form a 15x30x10 cluster. 25000 such clusters are loaded into the 40M member outline. The number of such data load files is set based on a goal of reaching 3.2GB, 4.8GB, and 6.4GB as database sizes for three outlines. Data load files are each loaded into a buffer, and after all files have been loaded in this manner, a load from the buffer is performed.

Aggregation is done with a goal of limiting the database growth to a factor of 1.25.

TABLE 4A

Dimbuild	**40M**	**60M**	**80M**
Products/Stores/Geography #mbrs	7M/13M/20M	10M/20M/30M	13M/27M/40M
#mbrs (M)	40	60	80
Products time (sec)	611	855	1138
Stores time (sec)	1235	1969	2719
Geography time (sec)	2188	3398	4786
Restructure time (sec)	426	675	938
Outline construction time (sec)	4460	6897	9581
Peak memory (KB)	8305664	12296080	16301040
Resting memory (KB)	6072904	8828648	11651360
Outline size (bytes)	4276502528	6193209344	8168325120
Database start time (sec)	420	729	932
ASO cache size (MB)	40	50	57
Data load			
# Load files	10	15	20
Load into buffer time (sec)	3780	5910	8160
Load from buffer time (sec)	160	218	347
Load time total (sec)	3940	6128	8507
Database file size (bytes)	3,288,334,336	4,940,890,112	6,593,445,888
Input data size (cells)	451,867,500	677,812,500	903,757,500
Aggregation to 1.25 stopping			
# views built	21	18	20
Aggregation time (sec)	2228	3652	2728
Database file size (bytes)	4,018,183,232	5,737,807,872	7,566,524,416
Aggregate data size (cells)	78,418,826	84,721,885	106,403,241

The second table (Table 4B) shows the results of running level-0 MDX queries on the 40M member database. A total of 25000 random queries were generated and files with 10 queries/file were prepared. A single Win32 client was used to submit the MDX queries using a custom API client program. The client contains various levels of retry loops if network problems were encountered.

The number of users issuing these MDX queries was varied from 100 to 1500. Each client process contained 100 threads, each of which logs in, and reads a designated query file containing 10 queries. The login time and the time to execute the MDX queries were measured. Minimum, maximum, and average values of these measures were computed.

The number of queries/sec was computed from the elapsed time (which is close to the maximum client elapsed time) and the number of successfully executed queries. The AgentThreads and ServerThreads configuration settings were set to 1024. There was no think time between queries. All times in seconds.

The level-0 query format reflects a scenario where a user has picked some small subgroups of related Products, Stores, and Geography entities, and is interested in certain measure values for them. For the other dimensions, a slice of the cube has also been picked. The members in a subgroup are related in the sense that they are likely to be siblings or first cousins.

TABLE 4B

#users	Avg Login time (seconds)	Avg Query time (seconds)	Avg Login+Query time (seconds)	#queries	Queries/Sec
100	2.83	39.6	42.4	1000	19.1
200	16.5	65	81.4	2000	19.3
300	30.6	68.6	99.2	2530	18.8
400	28.1	110.7	138.8	3380	19.5
500	39.8	161.7	201.5	5000	19.7
600	45.4	196.8	242.1	6000	19.7
700	167.4	28.7	196.1	7000	19.2
800	56.4	270	328.4	8000	19.9
900	84.8	262.3	347.1	8980	19.8
1000	242.1	34.4	276.5	9980	19.3
1100	62.9	389.3	452.1	10950	20
1200	83.9	403	486.9	12000	19.8
1300	102.2	406.1	508.3	12745	19.9
1400	119.1	373.6	492.7	12690	19.9
1500	89.7	416.3	506.1	12569	19.8

Level-0 MDX queries have the following format:

```
SELECT
MemberRange(Lag([<pm>], <l>), Lead([<pm>], 9-<l>)) ON AXIS(0),
MemberRange(Lag([<sm>], <l>), Lead([<sm>], 9-<l>)) ON AXIS(1),
MemberRange(Lag([<gm>], <l>), Lead([<gm>], 9-<l>)) ON AXIS(2),
Members([Measures]) ON AXIS(3)
FROM
<app>.<db>
WHERE
<slicer>
```

<pm> is a leaf member picked from the Products dimension, <sm> is a leaf member from the Stores dimension, and <gm> is a leaf member from the Geography dimension. <l> is an integer between 0 and 9, and therefore 9-<l> is in the same range. The expression on each of the first three axes results in a set of 10 leaf members each. <slicer> is a single tuple containing members picked from the remaining dimensions. The members picked in the query come from a pool of members into which data has been loaded.

The third table (Table 4C) is similar to Table 4B except that the random queries generated refer to upper-level members. All times in seconds.

The upper-level queries try to reflect the core of the demands that would be placed on the server when users navigate through data at various levels of aggregation.

Upper-level MDX queries have the following format:

```
SELECT
Except(Ancestors([<pm>], Generations(Dimension([<pm>]), 1)),
{[<pm>]}) ON AXIS(0),
Except(Ancestors([<sm>], Generations(Dimension([<sm>]), 1)),
{[<sm>]}) ON AXIS(1),
Except(Ancestors([<gm>], Generations(Dimension([<gm>]), 1)),
{[<gm>]}) ON AXIS(2),
Members([Measures]) ON AXIS(3)
FROM
<app>.<db>
WHERE
<slicer>
```

Ancestors of the chosen Products, Stores, and Geography leaf members are included in each query. The leaf members themselves are excluded.

TABLE 4C

#users	Avg Login time (seconds)	Avg Query time (seconds)	Avg Login+Query time (seconds)	#queries	Queries/Sec
100	0.96	183.4	184.4	1000	4.26
200	8.16	311.8	320	2000	4.44
300	15.9	493.4	509.3	3000	4.41
400	9.27	707.1	716.4	3990	4.41
500	8.75	925.4	934.2	4966	4.4
600	25.1	1147.8	1172.9	5960	4.39
700	35.2	1278.2	1313.4	6700	4.37
800	25.9	1456.8	1482.7	7467	4.38
900	29.8	1679.4	1709.2	8630	4.39
1000	28.4	1913.4	1941.7	9663	4.37
1100	25.7	1787.6	1813.3	9613	4.36
1200	48.6	2042.6	2091.2	11126	4.36
1300	51.2	2185.2	2236.5	11114	4.36
1400	49.8	1726.4	1776.2	10634	4.36
1500	61.9	1678.3	1740.2	10690	4.34

Both the query results show linear increase in users serviced with no degradation in throughput.

BLOCK STORAGE PERFORMANCE

In addition, various other tests were done featuring Essbase Analytics (Block Storage) cubes. Similar performance gains were observed out-of-the-box.

Table 5 table shows out-of-the-box performance comparisons on a medium-sized Block Storage database using Buffered IO.

Table 6 shows out-of-the-box performance comparisons on a medium-sized Block Storage database using Direct IO.

For Block storage applications we see performance gain when compared to 32-bit. The amount of gain varies with the particular model and the particular aggregation/calc being performed. The 64-bit environment also has the flexibility of using Direct IO with large enough cache size to accommodate almost the entire database in memory.

TABLE 5

LOAD & CALC	SIZE	32 BIT	64 BIT	Performance Improvement
	43MB	30.78	28.62	
LOAD TIMES (SEC)	102MB	77.61	71.24	
	241MB	180.08	131.32	
	3.2GB	1847.74	1176.04	1.6 times faster
CALC TIMES (SEC)	6.0GB	4262.6	2503.79	1.7 times faster
	15GB	7303.34	6791.89	1.1 times faster

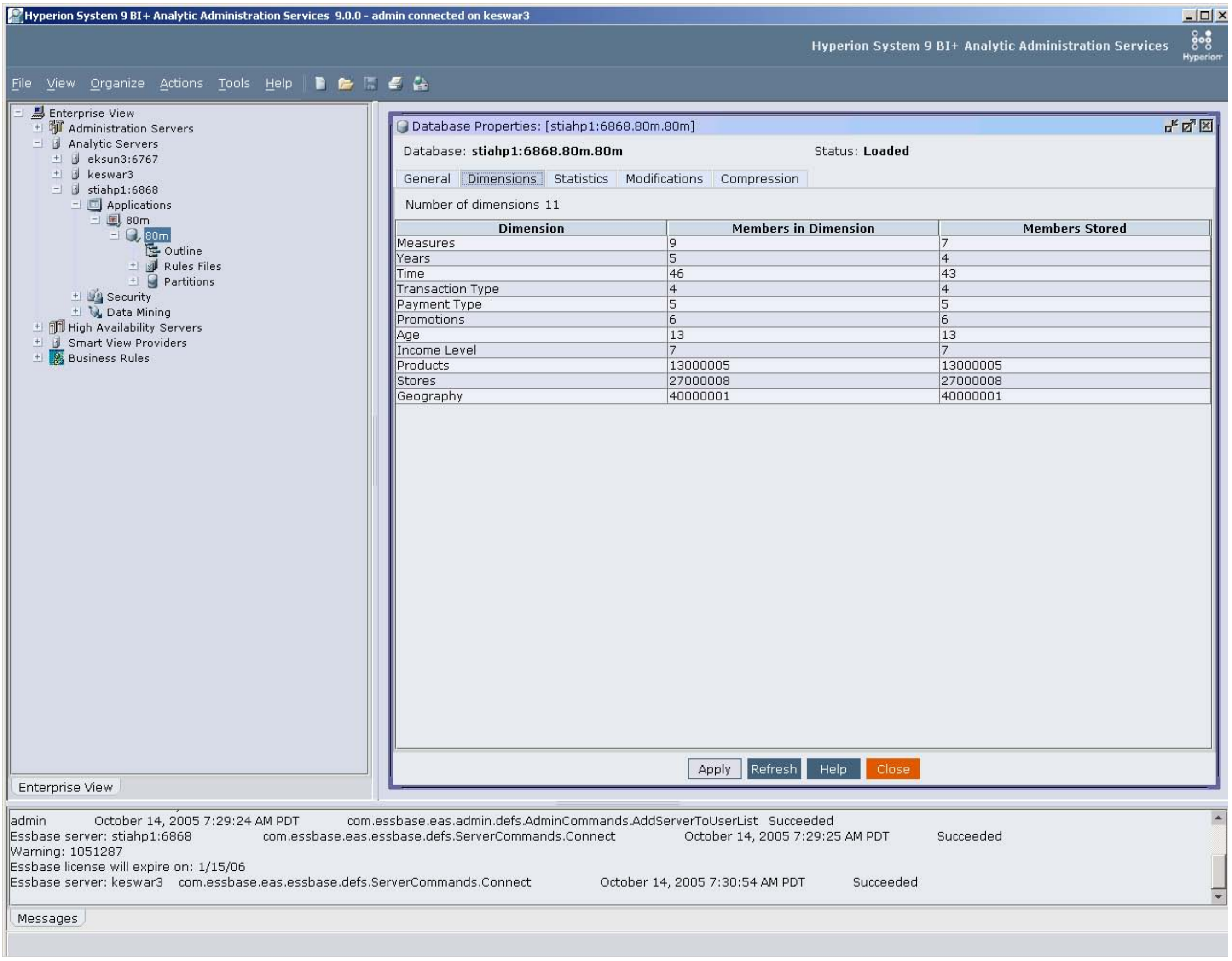
TABLE 6

LOAD & CALC	SIZE	32 BIT	64 BIT	Performance Improvement
	43MB	31.01	20.84	
LOAD TIMES (SEC)	102MB	72.5	68.49	
	241MB	152.66	132.97	1.2 times faster
	3.2GB	6310.79	1013.14	
CALC TIMES (SEC)	6.0GB	10605.3	2076.8	5 times faster
	15GB	NA	4889.52	

CONCLUSION

The Hyperion System 9 BI+ Analytic Services release 9.0.0 64-bit is available on HP-UX on Itanium and Windows on Itanium. The greatest advantage of 64-bit is the increased addressability which in turn increases the scalability of Analytic Services. The chip architecture is exploited by the compilers to produce optimal code which performs better out-of-the-box when compared to equivalent 32-bit platforms. The test results in the sections above have demonstrated 100% performance gain in aggregation and calc times; ability to build very large models in Aggregate storage; linear increase in concurrent queries with no degradation in throughput; support for more number of concurrent queries in the Analytic Services High Availability environment etc. Hyperion is excited about exploring the 64-bit platform further with new releases of Analytic Services. Also, with the next generation Itanium chips on the roadmap, 64-bit Analytic Services on Itanium promises to take scalability to the next level, and allow customers to gain better insight into the complex forces driving their business.

APPENDIX



Hyperion System 9 BI+ Analytic Administration Services 9.0.0 - admin connected on keswar3
Hyperion System 9 BI+ Analytic Administration Services
File View Organize Actions Tools Help
Enterprise View
Administration Servers
Analytic Servers
eksun3:6767
keswar3
stiahp1:6868
Applications
80m
80m
Outline
Rules Files
Partitions
Security
Data Mining
High Availability Servers
Smart View Providers
Business Rules
Enterprise View
Database Properties: [stiahp1:6868.80m.80m]
General Dimensions Statistics Modifications Compression
Apply Refresh Help Close
Messages

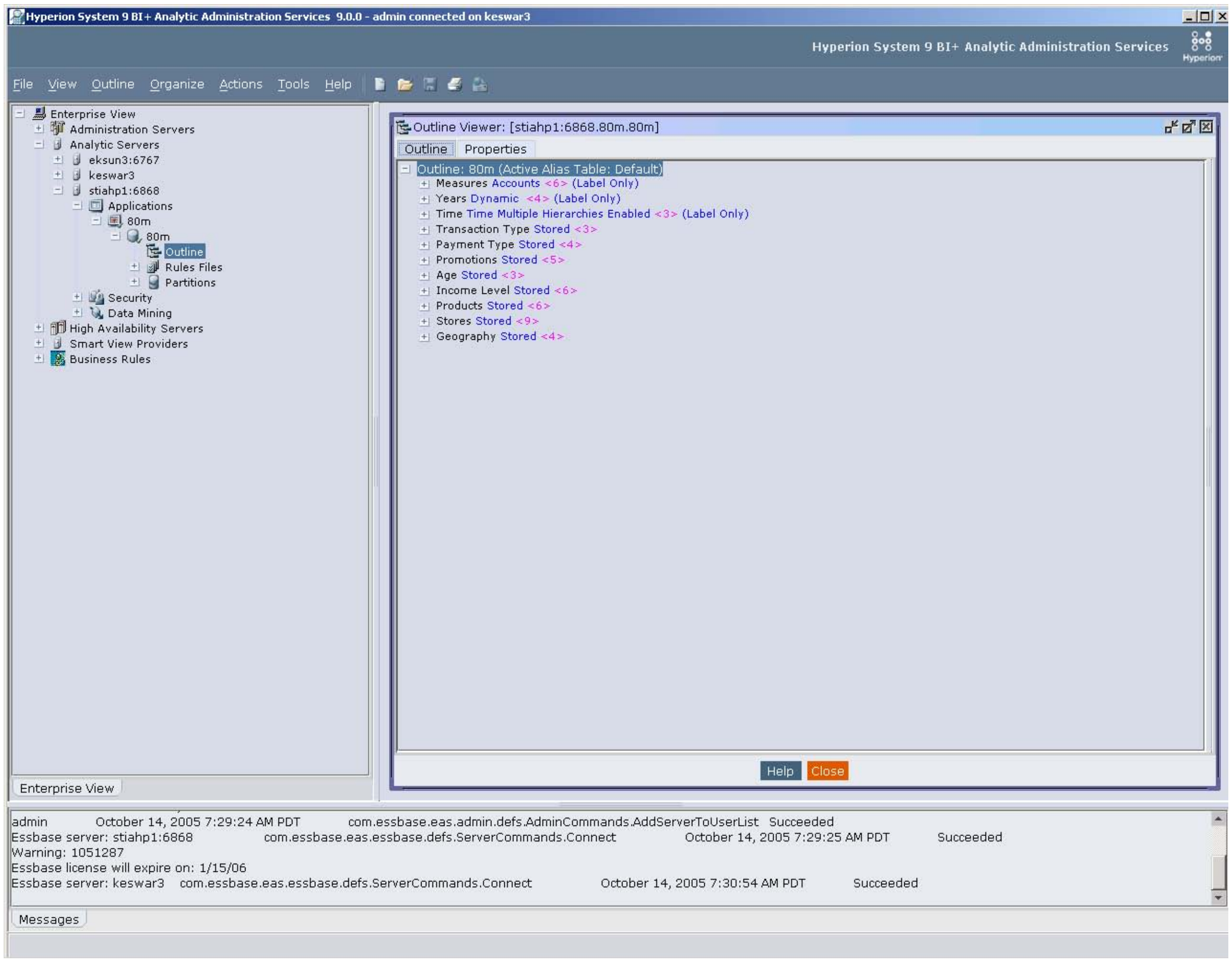
Database: **stiahp1:6868.80m.80m** Status: **Loaded**

Number of dimensions 11

Dimension	Members in Dimension	Members Stored
Measures	9	7
Years	5	4
Time	46	43
Transaction Type	4	4
Payment Type	5	5
Promotions	6	6
Age	13	13
Income Level	7	7
Products	13000005	13000005
Stores	27000008	27000008
Geography	40000001	40000001

```
admin            October 14, 2005 7:29:24 AM PDT          com.essbase.eas.admin.defs.AdminCommands.AddServerToUserList   Succeeded
Essbase server: stiahp1:6868            com.essbase.eas.essbase.defs.ServerCommands.Connect             October 14, 2005 7:29:25 AM PDT          Succeeded
Warning: 1051287
Essbase license will expire on: 1/15/06
Essbase server: keswar3    com.essbase.eas.essbase.defs.ServerCommands.Connect             October 14, 2005 7:30:54 AM PDT          Succeeded
```

APPENDIX


Hyperion System 9 BI+ Analytic Administration Services 9.0.0 - admin connected on keswar3
Hyperion System 9 BI+ Analytic Administration Services
File View Outline Organize Actions Tools Help
Enterprise View
Administration Servers
Analytic Servers
eksun3:6767
keswar3
stiahp1:6868
Applications
80m
80m
Outline
Rules Files
Partitions
Security
Data Mining
High Availability Servers
Smart View Providers
Business Rules
Outline Viewer: [stiahp1:6868.80m.80m]
Outline
Properties
Help
Close
Enterprise View
Messages

```
Outline: 80m (Active Alias Table: Default)
  Measures Accounts <6> (Label Only)
  Years Dynamic  <4> (Label Only)
  Time Time Multiple Hierarchies Enabled <3> (Label Only)
  Transaction Type Stored <3>
  Payment Type Stored <4>
  Promotions Stored <5>
  Age Stored <3>
  Income Level Stored <6>
  Products Stored <6>
  Stores Stored <9>
  Geography Stored <4>
```

```
admin          October 14, 2005 7:29:24 AM PDT          com.essbase.eas.admin.defs.AdminCommands.AddServerToUserList  Succeeded
Essbase server: stiahp1:6868          com.essbase.eas.essbase.defs.ServerCommands.Connect          October 14, 2005 7:29:25 AM PDT          Succeeded
Warning: 1051287
Essbase license will expire on: 1/15/06
Essbase server: keswar3    com.essbase.eas.essbase.defs.ServerCommands.Connect          October 14, 2005 7:30:54 AM PDT          Succeeded
```

PRODUCT INFORMATION TEL: 1.800.286.8000 (U.S. ONLY)
CONSULTING SERVICES E-MAIL: NORTHAMERICAN_CONSULTING@HYPERION.COM TEL: 1.203.703.3000
EDUCATION SERVICES E-MAIL: EDUCATION@HYPERION.COM TEL: 1.203.703.3535
WORLDWIDE SUPPORT E-MAIL: WORLDWIDE_SUPPORT@HYPERION.COM

PLEASE CONTACT US AT WWW.HYPERION.COM/CONTACTUS FOR MORE INFORMATION.

Hyperion®

HYPERION SOLUTIONS CORPORATION WORLDWIDE HEADQUARTERS
5450 GREAT AMERICA PARKWAY, SANTA CLARA, CA 95054
TEL: 1.408.744.9500 FAX: 1.408.588.8500