

Session #2129

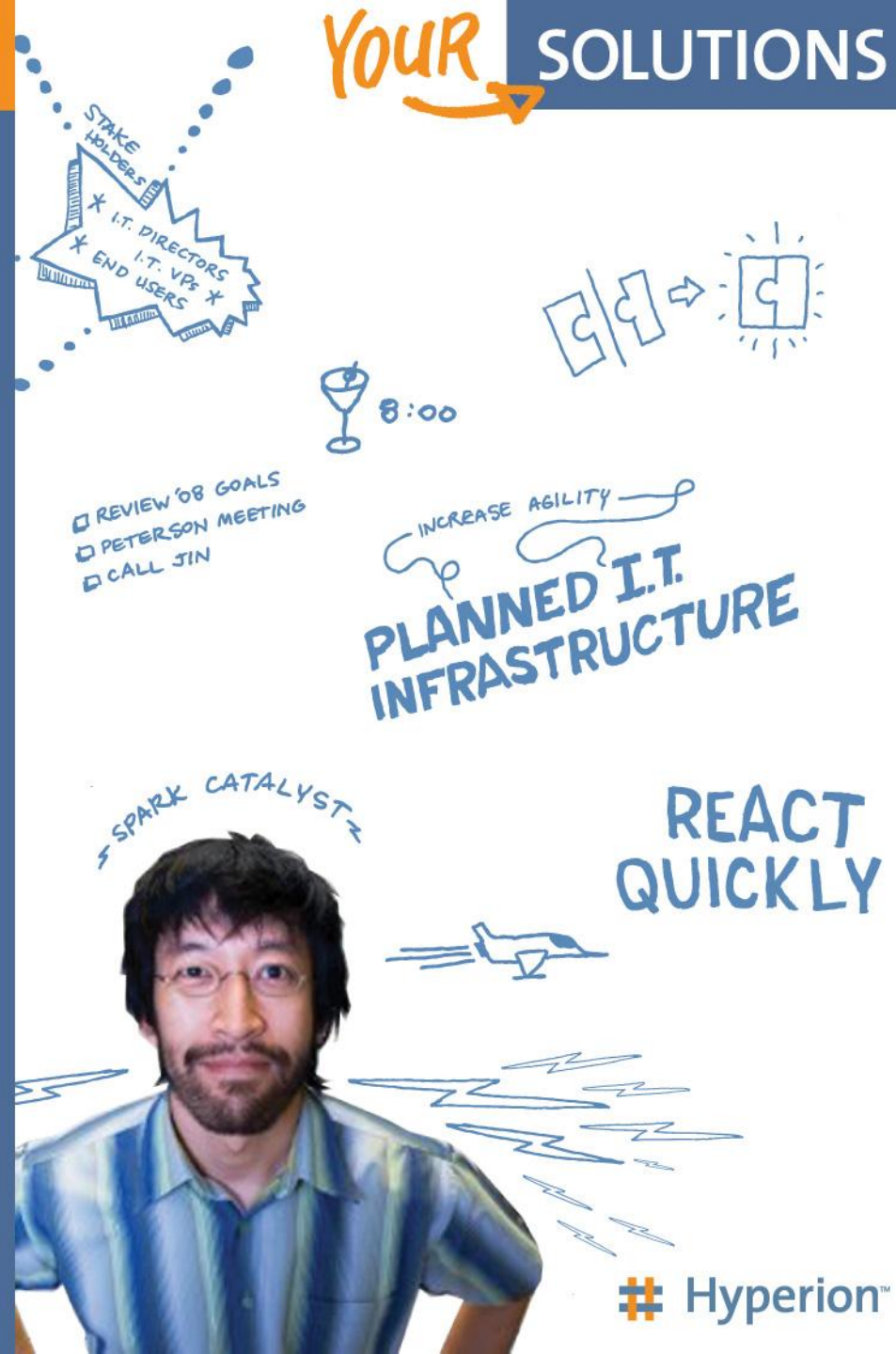
Optimization Techniques for Hyperion System 9 BI+ Essbase Analytics

Will Warren

Sr. Program Analyst,
Alliance Data

John Gibson

Senior Consultant,
interRel Consulting



Session Abstract

- Do you want to go beyond optimization presentations that talk only of theory and never show you techniques you can actually use? Do you want insight into design and optimization best practices when implementing Hyperion Essbase? This session will show you how Alliance Data improved system performance by an order of magnitude, as well as how it realized faster dimension builds, speedier data loads, and blazingly fast calculations by modifying caches, configuration settings, calculation commands, outline optimizations, and more! The presenters will share tips and tricks for design, optimization, implementation, and administration of Hyperion Essbase and finally, how to enhance the end-user experience and overall information delivery by adding dimensionality to your application. .



Agenda

- Introductions
- Optimization at Alliance
- Approach to Optimization & Tuning
 - Design Considerations
 - Performance Tuning
 - Administration
- Looking to System 9.3 New Features
- Conclusion

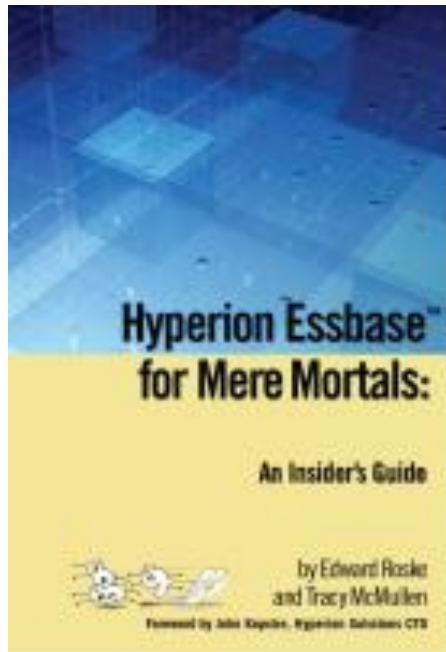


About interRel

- Headquarters in Texas, we consult nationwide
- Preferred Hyperion Partner, Oracle Partner
- Since our inception 10 years ago, focused only on Hyperion:
 - Implementations
 - Training
 - Publications
- 100% of our senior consultants are Hyperion Certified



Essbase for Mere Mortals



- The first book ever published on Essbase
- For end-users and administrators alike
- Foreword by John Kopcke, CTO of Hyperion Solutions
- Stop by booth 607, or order online at www.lulu.com



interRel Sessions

Monday, 23 rd April				Audience			Industry Focus	Customer Type
Time	Room	Session	Title	EU	DBA	IT		
1:00 PM	Swan Mockingbird 2	2103	Improving Financial Reporting at Michael's Stores: Tips and Tricks	✓	✓		Retail	Existing
2:30 PM	Swan Osprey Ballroom	1017	Eddie and the Consultants: An Updated Hyperion System 9 Musical	✓	✓	✓		All
4:00 PM	Swan Ballroom 6	1012	Why Move to Hyperion System 9? Busting the Migration Myths		✓	✓		All

Tuesday, 24 th April				Audience			Industry Focus	Customer Type
Time	Room	Session	Title	EU	DBA	IT		
8:30 AM	Dolphin S. Hemisphere V	2129	Optimization Techniques for Essbase: Tips and Tricks		✓			Existing
8:30 AM	Dolphin Oceanic 5	2130	Creating an Innovative Global Business Forecasting System at Alcon Labs	✓	✓		Healthcare, Manufacturing	New
8:30 AM	Swan Mockingbird 2	1065	Calc Scripts for Mere Mortals		✓			Existing
9:45 AM	Dolphin S. Hemisphere I	1067	Harnessing Hyperion Analyzer and System 9 Web Analysis: Tips and Tricks	✓	✓			Existing
11:00 AM	Dolphin S. Hemisphere V	1158	A Day in the Life of a Hyperion Essbase Administrator: Tips and Tricks		✓	✓		Existing
11:00 AM	Dolphin S. Hemisphere II	1188	Creating and Managing Financial Reports: Tips and Tricks		✓	✓		Existing
1:30 PM	Dolphin S. Hemisphere I	1168	Meeting Sarbanes-Oxley and Other Compliance Requirements with MDM	✓	✓			New
3:00 PM	Dolphin S. Hemisphere V	4004	Ask a Guru: Hyperion Essbase Tips & Tricks Roundtable		✓	✓		Existing
4:30 PM	Yacht Club Asbury Hall A	1011	Flexibility and Scalability Gains with Virtualization and Hyperion System 9			✓		Existing



interRel Sessions

Tuesday, 24 th April				Audience				
Time	Room	Session	Title	EU	DBA	IT	Industry Focus	Customer Type
8:30 AM	Dolphin S. Hemisphere V	2129	Optimization Techniques for Essbase: Tips and Tricks		✓			Existing
8:30 AM	Dolphin Oceanic 5	2130	Creating an Innovative Global Business Forecasting System at Alcon Labs	✓	✓		Healthcare, Manufacturing	New
8:30 AM	Swan Mockingbird 2	1065	Calc Scripts for Mere Mortals		✓			Existing
9:45 AM	Dolphin S. Hemisphere I	1067	Harnessing Hyperion Analyzer and System 9 Web Analysis: Tips and Tricks	✓	✓			Existing
11:00 AM	Dolphin S. Hemisphere V	1158	A Day in the Life of a Hyperion Essbase Administrator: Tips and Tricks		✓	✓		Existing
11:00 AM	Dolphin S. Hemisphere II	1188	Creating and Managing Financial Reports: Tips and Tricks		✓	✓		Existing
1:30 PM	Dolphin S. Hemisphere I	1168	Meeting Sarbanes-Oxley and Other Compliance Requirements with MDM	✓	✓			New
3:00 PM	Dolphin S. Hemisphere V	4004	Ask a Guru: Hyperion Essbase Tips & Tricks Roundtable		✓	✓		Existing
4:30 PM	Yacht Club Asbury Hall A	1011	Flexibility and Scalability Gains with Virtualization and Hyperion System 9			✓		Existing

Wednesday, 25 th April				Audience				
Time	Room	Session	Title	EU	DBA	IT	Industry Focus	Customer Type
8:30 AM	Dolphin S. Hemisphere V	1009	Hyperion System 9 User Provisioning: A Central Place for Managing Security		✓	✓		Existing
9:45 AM	Swan Osprey 1	2127	Making Fast Food Even Faster: Essbase at Taco Bueno	✓	✓		Retail, Services	Existing
11:00 AM	Swan Ballroom 6	1158B	A Day in the Life of a Hyperion Essbase Administrator: Tips and Tricks		✓	✓		Existing



Alliance Data

- Alliance Data is the one of the largest providers of transaction, credit and marketing services we serve the retail, petroleum, utility, financial services and hospitality markets.



Business Planning Goals

- To provide senior management with accurate & timely projected financial information at a Line of Business and total Company level in a SECURE environment.
- To enable management to take decisive action based upon fact rather than intuition.
- To reduce cycle times AND provide data integrity.
- To allow Corporate the ability to quickly spot anomalies based on trends and drill down to a lower level to research & resolve.



Business Planning Goals

- To empower end users to responsibly maintain accurate data (budget, forecast) with appropriate controls.
- To allow managers to do their OWN Ad-hoc reporting.
- To enable managers to quickly test models & assumptions using needed tools.
- To provide trending of historical information into future time periods for improved forecast accuracy.



Current State

- SMART system is built on Hyperion Essbase 6.5
- Actual – Pulled from PeopleSoft
- Budget / Forecast – Maintained in SMART
 - Most entered via spreadsheet lock & send
 - NAMs upload through uManage (an AlphaBlox app)
- Data Loads / Interfaces – Historical Actual data is loaded from PeopleSoft and historical Forecasts from a prior Essbase export text file.
- Foreign Exchange Conversions – The current Essbase system pulls all CAD conversion information from PeopleSoft. It does NOT handle spot/avg conversions in Essbase.
- Reporting – Most current Essbase reporting is via the excel add-in. Some users (primarily NAMs) still use AlphaBlox for forecast updates & reporting



Pain Points

- System performance
 - Batch window is too big (23:00 to 06:00)
 - Intra-Day Calc times are too long (~20 min)
 - System is unstable
- Audit – need visibility into Planning process.
- Maintenance – too many manual touch points.
- Architecture – how do we modernize?



Proposed Solution

- Upgrade to Hyperion System 9 BI+ platform
- Automate daily PeopleSoft / SMART validation
- Add a Year dimension
- Break the ORG dimension into its component PeopleSoft dimensions
- Implement a dedicated staging area in a relational star schema and utilize Analytic Integration Services
- Re-architect the outline to take advantage of member formulas
- Implement Hyperion Planning for forecast & budget data and split reporting/budgeting needs



Benefits

- Increase System performance
 - Shrink nightly batch window by 50% to 3.5 hrs
 - Increase retrieval performance by 50%
 - Avg ~30sec to 15sec
 - Worst ~3-5min to ~2.5 min
 - Enable intra-day calcs to run & finish every 15 min
- Improve Accountability
 - Visibility into the Budget/Forecast planning process
 - Sarbanes/Oxley controls for authentication/permissions
- Simplify maintenance
- Modernize architecture



Decisions, Decisions

Select a Hyperion platform:

- System 9 Analytic Server
- Essbase 7.x

ORG dimension:

- Break into separate component dimensions
- Leave as is



Decisions, Decisions

Currency reporting in USD & local currency required?

- No
- **Yes** - Is B/S required in new cubes?
 - Yes: This is beyond the scope of current project plans. It can be very expensive.
 - **No**: The conversion is relatively straightforward. We still need to decide:
 - Break Currency into its own dimension (regular or attribute)
 - Reorganize the ORG dimension



Decisions, Decisions

To facilitate more efficient outline aggregations of allocations, more generic formulas are better across scenarios (Act, Bud, Fct). Can we pass form factors for Act accounts from Oracle to EssBase?

- Yes - Fewer required Calc Scripts
- No - More Calc Scripts



Decisions, Decisions

Use Integration Services?

- No
- **Yes** - Included as part of System 9 upgrade; Additional license required for 7x.



Requirements for System 9 upgrade

- MSAD or LDAP required
- Finalize hardware projections (buy new server, lease new server, etc)

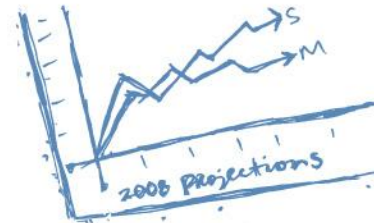


How Do We Get There?



Thought leader?!

- Key Stakeholders
- o Senior Mgmt
 - o Board of Directors
 - o Reg Agencies
 - o LOB Managers



Roadmap

Project Planning & Analysis ~ 2 weeks								
		Architecture: Installation / Upgrade and Configuration ~ 1 month						
	Detailed Design / Prototype ~1 month		Business Planning and Information Delivery Implementation ~2-4 months					
						Detailed Reporting (Journal Entry) ~1-2 months		
Short Term Enhancements ~ 1 month								

* Timelines are high level estimates and can be impacted by a number of factors including **scope**, resources, and budget.

Broken into projects managed separately...

Person-Weeks Estimate Separated by Project

	Low	High
1. System 9 upgrade	3	5
2. Analytic Server Optimization & Re-Architecture	18	24

-
- Plus an additional 2-3 weeks for Project Planning & Analysis for projects 2-3 (completed).
 - Includes part-time Project Management / QA for duration of projects 2.
 - Time is reduced when projects are run in conjunction to synchronization of work.
-

Approach to Optimization



Design Considerations



Design Considerations

- Minimize the Number of Dimensions
 - Avoid dimensions that do not offer descriptive data points
 - Reduce complexity and size of database
- Examine Dimension Combinations
- Avoid Repetition
 - Repeating indicates a need to split dimensions
 - Splitting dimensions reduces outline redundancy
- Avoid Interdimensional Irrelevance
- Split the database if necessary

Consider Using Attribute Dimensions

- Add dimensionality without increasing the size of the database
- View, aggregate and report
- Create crosstab reports
- Compare characteristics
- Group into ranges
- View multiple calculations
- Use in calculations and member formulas



When to Use Attributes

- Use crosstab reports
- Create reports with varying dimensions
- Hide a level of detail in reports
- Perform comparisons based on certain type of data
- Perform calculations based on characteristics
- Perform easy rollups on attributes
- Add dimensionality to the database without increasing sparsity of the database



When Not to Use Attributes

- Define characteristics of dense dimensions
- Define characteristics that vary over time
- Calculate a value by placing a formula on a member
- Minimize retrieval time; attributes are Dynamic Calc



Dimension Ordering Guidelines

- Largest Dense Dimensions
- Smallest Dense Dimensions
- Smallest Aggregating Sparse Dimensions
- Largest Aggregating Sparse Dimensions
- Non-aggregating Sparse Dimensions



Dimension Ordering Guidelines

- Dense dimensions - define the data block and must reside at the top of the outline
- Aggregating Sparse dimensions - dimensions that will be calculated to create new parent values
 - Should reside directly below the last Dense dimension in the outline
 - Placing these dimensions as the first Sparse dimensions positions them to be the first dimensions included in the calculator cache
 - Gives them an ideal location within the database for optimized calculation performance.
- Non-Aggregating Sparse dimensions - dimensions that organizes the data into logical slices.
 - Example - Scenario, Year or Version
 - Typically small, flat dimensions used to separate data
 - Not crucial for these dimensions to be included in the calculator cache because their members are typically isolated in FIX statements
 - Data is often times more dispersed within the database



Dimension Ordering based on Member Counts– Example

Dimension	Type-Size
Accounts	D – 94
Time Periods	D – 21
Metrics (Hrs, AHR, \$)	D – 14
Scenarios	AS – 9
Job Code	AS – 1,524
Organization	AS – 2,304
Versions	NAS – 7
Years	NAS – 7



D=Dense, AS=Aggregating Sparse, NAS=Non-Aggregating Sparse

Dimension Ordering based on Dimension Density - Example

Dimension	Type-Size	Density After Calc	Density After Load	Data Points Created
Time Periods	D – 21	85%	85%	-
Metrics (Hrs, AHR, \$)	D – 14	22%	22%	-
<i>Accounts</i>	<i>D – 94</i>	3 %	2%	-
Scenarios	AS – 9	22%	11%	199
Job Code	AS – 1,524	.56%	.23%	853
Organization	AS – 2,304	.34%	.09%	783
Versions	NAS – 7	19%	19%	-
Years	NAS – 7	14%	14%	-



AllianceData

D=Dense, AS=Aggregating Sparse, NAS=Non-Aggregating Sparse

How to Determine Individual Dimension Density

1. Make the dimension the lone Dense dimension
 2. Load and calculate just that dimension
 3. Check the block density value in Administration Services >> Database >> Properties >> Statistics
- Ordering the dense dimensions from most dense to least dense maximizes the clustering of the data
 - A more condensed database will perform better than one where the data has a highly dispersed population of data



Optimized Dimension Order

		Original	Optimized
<p><i>Typical Hourglass</i></p> <p>→</p>		Accounts (D)	Time Periods (D)
		Time Periods (D)	Metrics (D)
		Metrics (D)	Accounts (D)
		Years	Job Code (AS)
		Versions	Organization (AS)
		Scenarios	Years (NAS)
		Job Code	Versions (NAS)
		Organization	Scenarios (NAS)
		Employee Status	Employee Status (Attr Dim)
		Fund Group	Fund Group (Attr Dim)
			<p><i>Modified Hourglass</i></p> <p>←</p>

D=Dense, AS=Aggregating Sparse, NAS=Non-Aggregating Sparse

Performance Tuning

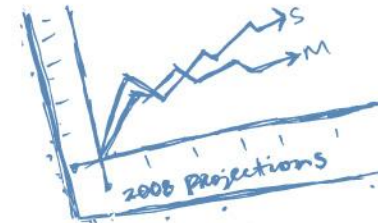
YOUR SOLUTIONS



Thought leader?!

- Key Stakeholders
- o Senior Mgmt
 - o Board of Directors
 - o Reg Agencies
 - o LOB Managers

Strike a BALANCE!!



Keep in Mind: Tuning

- **There isn't one right answer**
- Some of the tuning guidelines can contradict other tuning guidelines
- Tuning for calculations vs. tuning for retrievals
- The tuning information provided in this chapter is meant to help you in the development of your applications
- In some databases, these tuning tips will have significant impact
- In other databases, the tuning tips won't
- Test, test, test!!



Improve Essbase Performance

- Periodically reset a database
 - Over time page files grow
 - Maxl – alter database appname.dbname reset
- Explicit Restructure (welcome back)
 - alter database *DBS-NAME* force restructure
- Delayed Free Space Recovery
 - alter database *DBS-NAME* recover freespace



Compression

- Can use multiple compressions under 7x
- Each block will use one type of compression
 - None
 - zLib
 - Good for sparse data
 - Will only use zLib
 - Index Value Pair
 - Can't assign directly
 - Good for large blocks with sparse data
 - Bitmap
 - Good for non-repeating data
 - Will use Bitmap or IVP
 - RLE = Run Length Encoding
 - Good for data with zeros
 - Good for data that repeats (such as budgeting)
 - Will use RLE, Bitmap, or IVP



Tuning Compression

- Utilize parallel calculation by ordering your dimensions correctly (hourglass on a stick)
- Consider re-organizing dimensions and setting compression to RLE to reduce database size
- Consider using RLE, because it will allow each block to be RLE, Bitmap, or Index-Value Pair as needed



Caches

■ Index Cache

- Last index page into RAM, next out of RAM as cache is filled
- Default is 1024
- Generally, set to hold index in RAM
- Cache can be too big if index is huge

■ Data Cache

- Last block into RAM, next out of RAM as caches are filled
- Default is 3072
- Cache can be too big
- Uncompresses block in RAM (using more data cache)



Factors Affecting Cache Sizing

- Database size
- Block size
- Index size
- Available memory
- Data distribution
- Sparse / dense configuration
- Needs of database (e.g. complexity of calculations)



Priority for Memory Allocation

1. Index Cache
2. Data File Cache
3. Data Cache



Guideline for Index Cache

- Default
 - Buffered I/O: 1024 KB (1048576 bytes)
 - Direct I/O: 10240 KB (10485760 bytes)
- Guideline:
 - Combined size of all `essn.ind` files, if possible; otherwise, as large as possible
 - Do not set this cache size higher than the total index size, as no performance improvement results



Guideline for Data File Cache

- Only set if using Direct I/O
- Default
 - Direct I/O: 32768 KB (33554432 bytes)
- Guideline
 - Combined size of all *essn.pag* files, if possible; otherwise as large as possible



Guideline for Data Cache

- Default
 - 3072 KB (3145728 bytes)
- Guideline
 - 0.125 * Combined size of all `essn.pag` files, if possible; otherwise as large as possible
 - Increase value if any of these conditions exist:
 - Many concurrent users are accessing different data blocks
 - Calculation scripts contain functions on sparse ranges, and the functions require all members of a range to be in memory (for example, when using `@RANK` and `@RANGE`)
 - For data load, the number of threads specified by the `DLTHREADSWRITE` setting is very high and the expanded block size is large



Cache Hit Ratios

- Hit Ratios evaluate how well caches are being utilized
- Indicates the percentage of time that a requested piece of information is available in the cache
- Higher the better
- Right click on the Database and select Properties. Navigate to the Statistics tab in Administration Services to view hit ratios
- Index Cache Hit Ratio setting indicates the success rate in locating index information in the index cache without having to retrieve another index page from disk
 - Goal = 1
- Data File Cache Hit Ratio setting indicates the success rate in locating data file pages in the data file cache without having to retrieve the data file from disk
- Data Cache Hit Ratio setting indicates the success rate in locating data blocks in the data cache without having to retrieve the block from the data file cache
 - Goal = .3 or higher



Calculator Cache

- Analytic Services uses the calculator cache bitmap if the database has at least two sparse dimensions, and either of these conditions are also met:
 - You calculate at least one, full sparse dimension
 - You specify the SET CACHE ALL command in a calculation script
- The best size for the calculator cache depends on the number and density of the sparse dimensions in your outline
- Default calculator cache size is set in the `essbase.cfg`
- You can set the size of the calculator cache within a calculation script (setting is used only for the duration of the calculation script)



Calculator Cache Bitmap

- Bitmap dimensions
 - Sparse dimensions from the database outline that Essbase fits into the bitmap until the bitmap is full
 - Each member combination of the sparse dimensions placed in the bitmap occupies 1 bit of memory
 - Must be enough space in the bitmap for every member combination of a sparse dimension for it to be placed in the bitmap
- Anchoring dimensions
 - Remaining one or more sparse dimensions in the database outline that do not fit into the bitmap.
- Essbase starts with the first sparse dimension in the database outline and fits as many sparse dimensions as possible into the bitmap.
- Calculator cache controls the size of the bitmap; therefore controlling the number of dimensions that can fit into the bitmap
- Essbase cannot use the bitmap to determine whether or not blocks exist for Anchoring dimensions



Guideline for Calculator Cache

- Factors
 - Available memory
 - Nature and configuration of the database
- Calculator cache = Bitmap size in bytes * Number of bitmaps
- Bitmap size in bytes = Max ((member combinations on the bitmap dimensions/8), 4)
- Number of bitmaps = Maximum number of dependent parents in the anchoring dimension + 2 constant bitmaps
- Minimum bitmap size is 4 bytes
- See appendix for example



Fragmentation

- Unused disk space
- Watch out for
 - Read/write databases where users constantly update data
 - Execute calcs around the clock
 - Frequent updates and recalc's of dense members
 - Poorly designed data loads
 - Large number of Dynamic Calc and Store members
 - Isolation level of uncommitted access with commit block = zero



Remove Fragmentation

- Perform an export of the database, delete all data in the database with `CLEARDATA`, and reload the export file
- Force a dense restructure of the database



Commit Blocks

- Using Uncommitted Access
 - When Commit Level is reached, blocks write to hard drive
- Default is 3000 blocks; Increase to avoid I/O of frequent commits
- Setting Commit Blocks to Zero
 - Writes at completion of the entire transaction
 - Will dramatically improve calculation time
 - Will fragment your PAG file during a calculation
 - Resource intensive



Statistics to Monitor

- Compression ratio - ratio of the compressed block size (including overhead) to the uncompressed block size
- Data block size - determined by the amount of data in a particular combination of dense dimensions.
 - Data block size is $8n$ bytes, where n is the number of cells that exist for that combination of dense dimensions.
 - Guideline - 8 to 100 KB



Optimized Data Load Order

Outline Order	Data File Order and Sort
Time Periods (D)	Scenarios (NAS)
FTE Metrics (D)	Versions (NAS)
Accounts (D)	Years (NAS)
Job Code (AS)	Organization (AS)
Organization (AS)	Job Code (AS)
Years (NAS)	Accounts (D)
Versions (NAS)	FTE Metrics (D)
Scenarios (NAS)	Time Periods (D)
Employee Status (Attr Dim)	
Fund Group (Attr Dim)	



Data Load Tips

- Follow data file dimension load order described in previous slide
- Use dense dimension for data column headers
- Avoid unnecessary data fields in source data
- Load from the server vs. the client
- Pre-aggregate records before loading



Faster Calculations

1. Outline consolidation
2. Member formulas
3. Calc scripts



Restructuring

- You Essbase outline will constantly change
 - New accounts, new entities, new products
- Changes to the outline forces Essbase to restructure the database
- Can be a time consuming process depending on the type of restructure and database size



Full Restructure

- Implicit – run when an otl is updated (manually or via dimension build)
- Move, delete, or add a dense member
- Restructures the data blocks
- Regenerates the index
- Requires a recalculation of the database
- Time consuming



Sparse Restructure

- Implicit – run when an otl is updated (manually or via dimension build)
- Move, delete, or add a sparse member
- Does NOT restructure the data blocks
- Regenerates the index
- Usually much faster than Full Restructure



Outline Restructure

- Implicit – run when an otl is updated (manually or via dimension build)
- Change that effects the outline only
 - Add or change alias, formula, etc.
- Does NOT restructure the data blocks
- Does not restructure the index
- Very fast



Explicit Restructure

- Administration manually initiates a database restructure



Reducing Restructure Time

- If you change a dimension frequently, make it sparse.
- Use incremental restructuring to control when Essbase performs a required database restructuring.
- Select options when you save a modified outline that reduce the amount of restructuring required



Looking to Hyperion System 9.3



New 9.3 Features

- Calc command to remove of # Missing blocks
- Non-consolidating members ^
 - Profit
 - Sales
 - Unit
 - Price ^
 - Tells Analytic Services not to aggregate this member across ANY dimension
 - Similar to ~
 - ~
 - Do not rollup to parent
 - Will still roll up for across other dimensions



- Do not Aggregate

	A	B	C	D	E	F	G
1		New York	Massachusetts	Florida	Connecticut	New Hampshire	East
2	Week 1 2006	10	#Missing	#Missing	#Missing	#Missing	10
3	Week 2 2006	#Missing	20	#Missing	#Missing	#Missing	20
4	Week 3 2006	#Missing	#Missing	30	#Missing	#Missing	30
5	Week 4 2006	#Missing	#Missing	#Missing	40	#Missing	40
6	Week 5 2006	#Missing	#Missing	#Missing	#Missing	50	50
7	January 2006	10	20	30	40	50	150
8							



- Do not consolidate

	A	B	C	D	E	F	G
1		New York	Massachusetts	Florida	Connecticut	New Hampshire	East
2	Week 1 2006	10	#Missing	#Missing	#Missing	#Missing	#Missing
3	Week 2 2006	#Missing	20	#Missing	#Missing	#Missing	#Missing
4	Week 3 2006	#Missing	#Missing	30	#Missing	#Missing	#Missing
5	Week 4 2006	#Missing	#Missing	#Missing	40	#Missing	#Missing
6	Week 5 2006	#Missing	#Missing	#Missing	#Missing	50	#Missing
7	January 2006	#Missing	#Missing	#Missing	#Missing	#Missing	#Missing
8							+



Unchanged Cells in Calculation

- EXCLUDE / ENDEXCLUDE
- Calculate everything except ... a subsection
- Opposite of a FIX / ENDFIX

EXCLUDE (South, West)

Calc Dim (Accounts, Market):

ENDEXCLUDE

Extracting Essbase Data – Pre System 9.3

- Report script
- HAL
- Jexport
- Data exports
- Visual Basic
- Excel Add-in
- Database Export



New in System 9.3 - Subset Data Export

- Export slices of data
- Leverages the calc engine as a native function
 - Faster than report scripts and JEXPORT
 - Calc engine is faster than report engine
- Embed in a calc script
- Use within a Fix statement to define the slice to export
- Can push to mutiple formats
 - CSV, tab, relational database table



Binary Export/Import

- Different
- Goal is to move / copy out data blocks themselves in compressed encrypted format
- Fast backups
- Calc script has option to export in binary format
- Calc script has option to import binary format
- Exported file can only be imported into database with the same dimensionality
- Binary format
- DATAEXPORT “BINFILE” “[file_name]”;
- DATAIMPORTBIN “[file_name]”;
- Ignores fixes on dense members (copies blocks – intersection of sparse dimensions)



Essbase Analytics 9.3

- Calc Enhancements
 - Subset data export
 - Binary export
 - Remove #missing
 - Non-consolidating members
 - Unchanged cells
- General Enhancements
 - 64 bit platform support
 - MaxL password encryption
 - Run As support
 - Reference cubes
 - Analytic Provider Services



Expanded 64-bit Support

- AIX
- Solaris
- Windows (Itanium chip set)
- Opteron (Windows based)
- Xeon (Windows based)

- Analytic Server only



Reference Cubes

- Improve performance of XREF calculations
- Creates a small in memory cube
 - Shares the same memory space
- If database A needs information from another database, A can pull information from reference cube instead of across the server
- Considerations
 - 8000 cells
 - Dimensions only, no hierarchies
 - No dimension types
 - Reference cube types can coexist
 - Example: Small rate and driver cube spinning in memory copies



Conclusion

- Introductions
- Optimization at Alliance
- Approach to Optimization & Tuning
 - Design Considerations
 - Performance Tuning
 - Administration
- Looking to System 9.3 New Features
- Tuning Enterprise Analytics / ASO
- Conclusion



Session #2129

Optimization Techniques for Hyperion System 9 BI+ Essbase Analytics

Will Warren
Sr. Program Analyst,
Alliance Data

John Gibson
Senior Consultant,
interRel Consulting

Please Complete Your Session Evaluation





Hyperion™

The future in sight