



## Document:

---

### Essbase ASO “A Quick Reference Guide”

---

## Description:

The document provides an overview on “Hyperion Essbase Aggregate Storage Option”. The document also outlines the major differences between BSO and ASO.

## History:

Version	Description Change	Author	Publish Date
0.1	Initial Draft	Gaurav Shrivastava	28-May-2011
01.	Review 1 <sup>st</sup>	Amit Sharma	14 <sup>th</sup> Jun 2011

# Table of Contents

1.	<a href="#">Introduction.....</a>	<a href="#">3</a>
2.	<a href="#">Key Difference between ASO and BSO.....</a>	<a href="#">3</a>
3.	<a href="#">Aggregate Storage Overview.....</a>	<a href="#">10</a>
	a. <a href="#">Key Aggregate Storage Characteristics.....</a>	<a href="#">11</a>
	b. <a href="#">Design Considerations.....</a>	<a href="#">11</a>
	c. <a href="#">Member Formulas.....</a>	<a href="#">12</a>
4.	<a href="#">Aggregate Storage Production Cycle .....</a>	<a href="#">12</a>
	a. <a href="#">Application and Database Trees .....</a>	<a href="#">13</a>
	b. <a href="#">Directory Structures .....</a>	<a href="#">13</a>
	c. <a href="#">Rules Files for Building Outlines .....</a>	<a href="#">14</a>
5.	<a href="#">Designing Aggregate Storage Outline Hierarchies.....</a>	<a href="#">17</a>
	a. <a href="#">Multiple Hierarchies .....</a>	<a href="#">18</a>
	b. <a href="#">Stored Hierarchies .....</a>	<a href="#">18</a>
	c. <a href="#">Dynamic Hierarchies .....</a>	<a href="#">19</a>
6.	<a href="#">Designing Alternate Hierarchies .....</a>	<a href="#">19</a>
	a. <a href="#">Attribute Dimension Design .....</a>	<a href="#">19</a>
	b. <a href="#">Shared Members Hierarchy Design.....</a>	<a href="#">20</a>
7.	<a href="#">Converting Block Storage to Aggregate Storage .....</a>	<a href="#">21</a>
	a. <a href="#">Selecting a Source Outline .....</a>	<a href="#">21</a>
	b. <a href="#">Verifying Outline Corrections.....</a>	<a href="#">23</a>
	c. <a href="#">Selecting a Destination.....</a>	<a href="#">24</a>

**Introduction:** Aggregate storage technique is used when application needs more dimensions and members in order to support higher degree of analysis without compromising the cube performance. Aggregate storage is mainly used for applications where reporting on business data is considered as primary requirements. Data load in aggregate storage is faster than block storage and the data consolidation at the higher level is done automatically. Aggregate storage required less space in disk and data retrieval is also faster because data is always available in aggregated form. Aggregate storage application is approximate is similar as block storage application but it has so many new features. Aggregate storage database used where application require large dimensionality.

**Customer analysis** - Data is analyzed from any dimension, and there are potentially millions of customers.

**Procurement analysis** - Many products are tracked across many vendors.

**Logistics analysis** - Near real-time updates of product shipments are provided.

Below are some benefits of ASO.

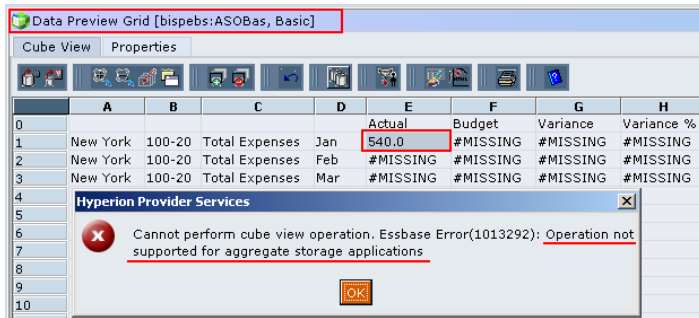
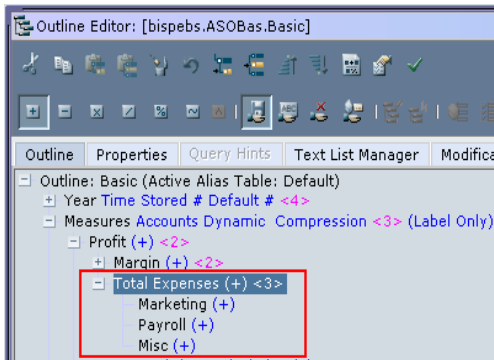
1. Faster load and calc times provide
2. Lower hardware costs
3. Lower maintenance costs
4. Higher availability

### Key Difference between Aggregate storage and block storage

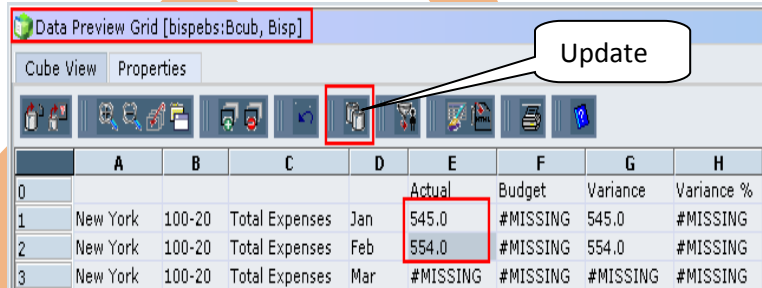
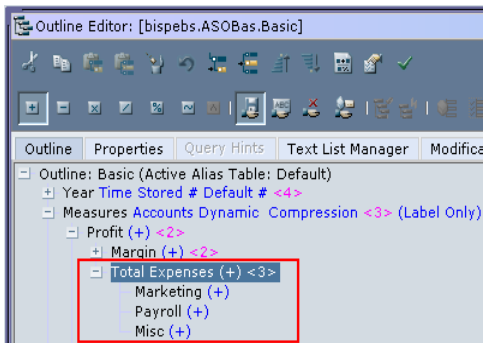
	Aggregate Storage	Block Storage
1	Data load can be possible at level 0 only	Data load is possible at any level
2	Write back functionality not supported	Write back functionality supported
3	No need to run consolidation operation	Need to run consolidation operation
4	Can set data load value	Can't set data load value
5	Allow to set system resource utilization	Not available
6	All calculation done through MDX	No calculation script
7	Complete cube has dynamic calc feature, all formulas and aggregation executes at runtime	Only dynamic storage members calculate formulas and aggregation at runtime
8	*.csc file creates for (aggregate storage)	*.csc file creates for (Block Storage)
9	Data access is faster	Comparative slower
10	Can have more number of dimension	Performance decrease as number of dimension increase
11	No sparse and dense dimension	Sparse and Dense dimension exist
12	Fast query processing	Comparative slower
13	Only level 0 data can be export	No restriction on data export
14	No currency database	Currency database exists

#### 1. Data load can be possible at level 0 only and write back functionality

In aggregate storage you can't load data at any level. In this example "Total Expenses" is level 1 member and if you load data in to it, Essbase will give you're an error.



Data load at any level is possible in Block Storage Application. Edit data field and click on update button for verification refresh data grid.



This example also shows that you can't write back in aggregate storage but it allow in block storage.

## 2. No need to run consolidation operation

When you load data in to aggregate storage, data will immediately available at all parent level of hierarchy. Load data in below combination of dimension, sales is level 0 member. We will load data in sales and verify that data will be immediately available for "Margin" level 1 member.

Data is not available for below combination.

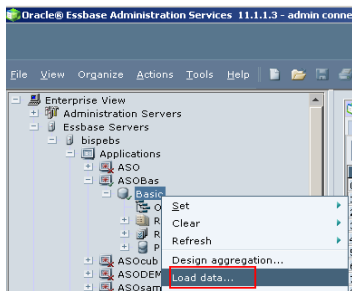
	A	B	C	D	E	F	G
0					Actual	Budget	Variance
1	Texas	400-10	Sales	Jan	#MISSING	#MISSING	#MISSING
2	Texas	400-10	Sales	Feb	#MISSING	#MISSING	#MISSING
3	Texas	400-10	Sales	Mar	#MISSING	#MISSING	#MISSING
4	Texas	400-10	COGS	Jan	#MISSING	#MISSING	#MISSING
5	Texas	400-10	COGS	Feb	#MISSING	#MISSING	#MISSING
6	Texas	400-10	COGS	Mar	#MISSING	#MISSING	#MISSING
7	Texas	400-20	Sales	Jan	#MISSING	#MISSING	#MISSING
8	Texas	400-20	Sales	Feb	#MISSING	#MISSING	#MISSING
9	Texas	400-20	Sales	Mar	#MISSING	#MISSING	#MISSING
10	Texas	400-20	COGS	Jan	#MISSING	#MISSING	#MISSING
11	Texas	400-20	COGS	Feb	#MISSING	#MISSING	#MISSING
12	Texas	400-20	COGS	Mar	#MISSING	#MISSING	#MISSING
13	Texas	400-30	Sales	Jan	#MISSING	#MISSING	#MISSING
14	Texas	400-30	Sales	Feb	#MISSING	#MISSING	#MISSING
15	Texas	400-30	Sales	Mar	#MISSING	#MISSING	#MISSING
16	Texas	400-30	COGS	Jan	#MISSING	#MISSING	#MISSING
17	Texas	400-30	COGS	Feb	#MISSING	#MISSING	#MISSING
18	Texas	400-30	COGS	Mar	#MISSING	#MISSING	#MISSING

Data load text file

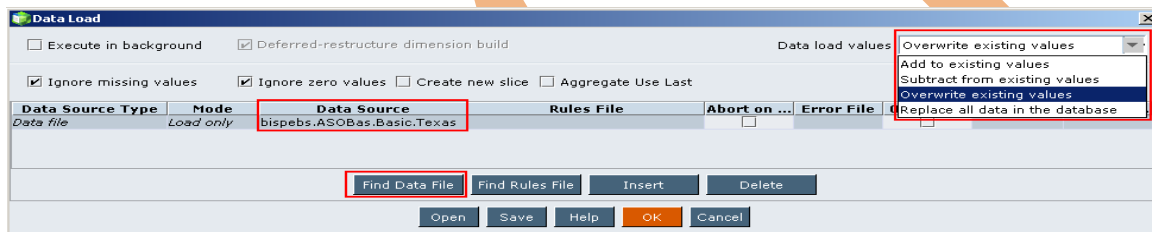
Texas.txt - Notepad

				Actual	Budget
Texas	"400-10"	Sales	Jan	547	875
Texas	"400-10"	Sales	Feb	858	457
Texas	"400-10"	Sales	Mar	654	659
Texas	"400-20"	Sales	Jan	547	875
Texas	"400-20"	Sales	Feb	858	457
Texas	"400-20"	Sales	Mar	654	659
Texas	"400-30"	Sales	Jan	858	457
Texas	"400-30"	Sales	Feb	654	659
Texas	"400-30"	Sales	Mar	547	875
Texas	"400-10"	COGS	Jan	247	875
Texas	"400-10"	COGS	Feb	458	457
Texas	"400-10"	COGS	Mar	254	659
Texas	"400-20"	COGS	Jan	247	300
Texas	"400-20"	COGS	Feb	458	400
Texas	"400-20"	COGS	Mar	554	500
Texas	"400-30"	COGS	Jan	458	400
Texas	"400-30"	COGS	Feb	554	600
Texas	"400-30"	COGS	Mar	400	600

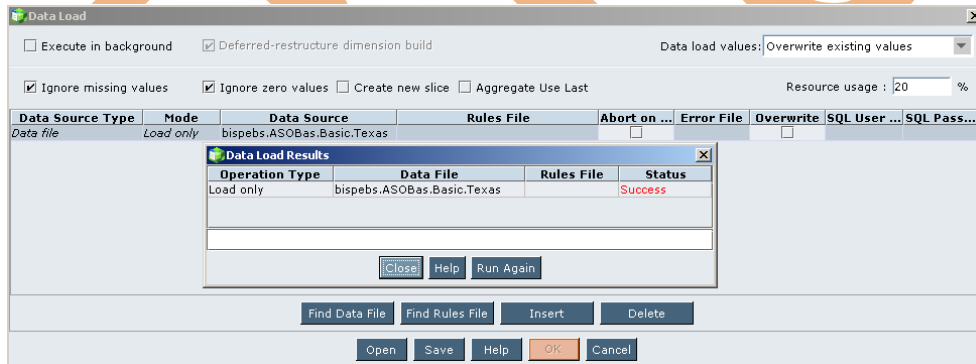
Right Click on data base → select load data



Select data file and data load value method then click ok.



Data is loaded successfully.



Without running any calculation script or consolidate operation data is available at level 0.

	A	B	C	D	E	F	G
0					Actual	Budget	Variance
1	Texas	400-10	Sales	Jan	547.0	875.0	-328.0
2	Texas	400-10	Sales	Feb	858.0	457.0	401.0
3	Texas	400-10	Sales	Mar	654.0	659.0	-5.0
4	Texas	400-10	COGS	Jan	247.0	875.0	-628.0
5	Texas	400-10	COGS	Feb	458.0	457.0	1.0
6	Texas	400-10	COGS	Mar	254.0	659.0	-405.0
7	Texas	400-20	Sales	Jan	547.0	875.0	-328.0
8	Texas	400-20	Sales	Feb	858.0	457.0	401.0
9	Texas	400-20	Sales	Mar	654.0	659.0	-5.0
10	Texas	400-20	COGS	Jan	247.0	300.0	-53.0
11	Texas	400-20	COGS	Feb	458.0	400.0	58.0
12	Texas	400-20	COGS	Mar	554.0	500.0	54.0
13	Texas	400-30	Sales	Jan	858.0	457.0	401.0
14	Texas	400-30	Sales	Feb	654.0	659.0	-5.0
15	Texas	400-30	Sales	Mar	547.0	875.0	-328.0
16	Texas	400-30	COGS	Jan	458.0	400.0	58.0
17	Texas	400-30	COGS	Feb	554.0	600.0	-46.0
18	Texas	400-30	COGS	Mar	400.0	600.0	-200.0

### Data at level 1

Data is consolidating automatically for parent level. Data is available for "Margin".

	A	B	C	D	E	F	G
0					Actual	Budget	Variance
1	Texas	400-10	Margin	Jan	300.0	0.0	300.0
2	Texas	400-10	Margin	Feb	400.0	0.0	400.0
3	Texas	400-10	Margin	Mar	400.0	0.0	400.0
4	Texas	400-20	Margin	Jan	300.0	575.0	-275.0
5	Texas	400-20	Margin	Feb	400.0	57.0	343.0
6	Texas	400-20	Margin	Mar	100.0	159.0	-59.0
7	Texas	400-30	Margin	Jan	400.0	57.0	343.0
8	Texas	400-30	Margin	Feb	100.0	59.0	41.0
9	Texas	400-30	Margin	Mar	147.0	275.0	-128.0

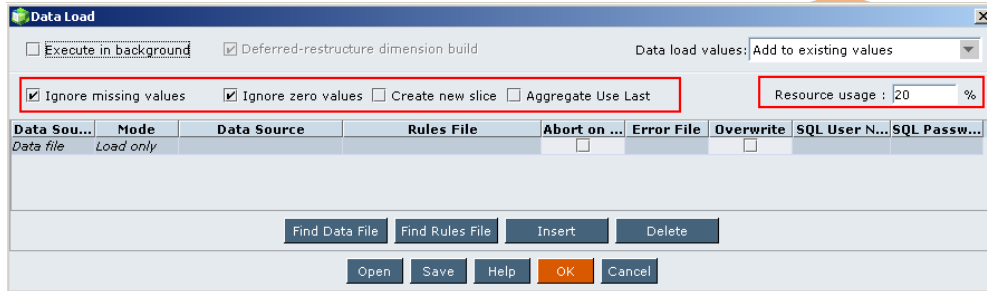
No Execute calculation option is available for aggregate storage application.

The screenshot shows a context menu for an ASO application. The menu items include: Set, Clear, Refresh, Design aggregation..., Load data..., Merge, Export..., Select compression dimension, Rename..., Delete, Edit, Create, Query tracking, User/group access, Expand all, Collapse all, Preview data, and Add to. The 'Execute calculation...' option is highlighted in red. A callout box labeled 'ASO application' points to the menu. Another callout box labeled 'BSO application' points to the 'Execute calculation...' option.

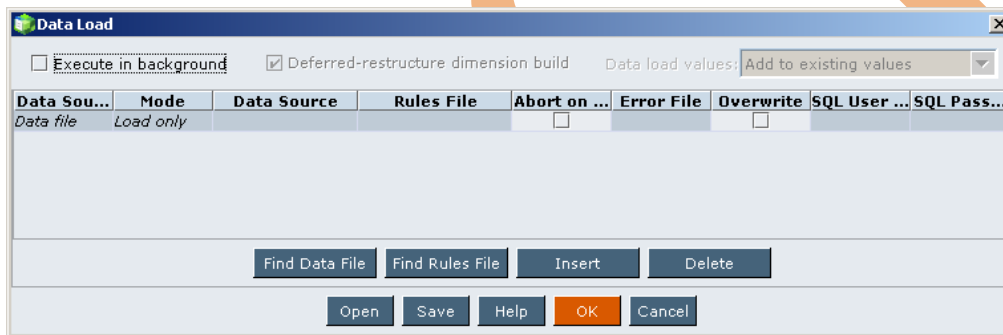
### 3. Set system resource utilization

While loading data aggregate storage allows you to set resource utilization. Resource utilization option supports to execute other tasks simultaneously. Some other options those are available for aggregate storage.

#### Dataload in aggregate storage

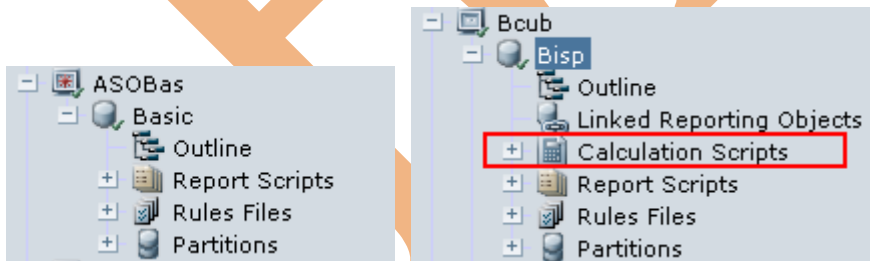


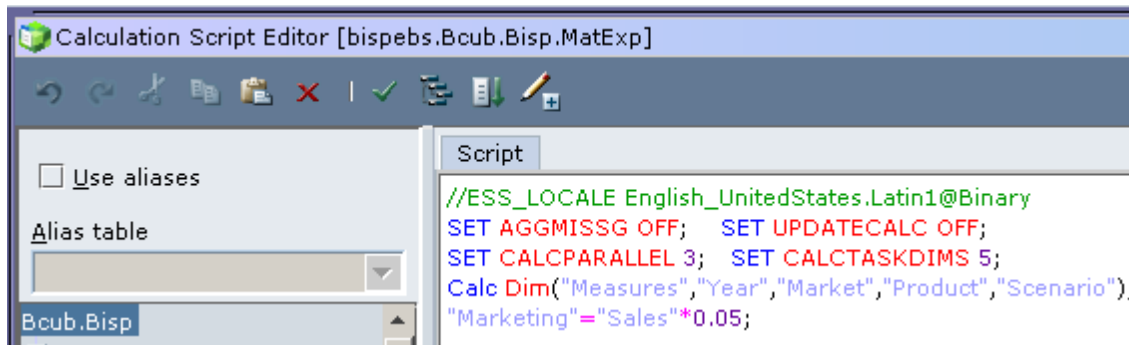
#### Dataload in block storage



### 4. Calculation done through MDX

Calculation script is not supported in aggregate storage applications. You have write calculation script for any calculation.





## 5. Data access is faster

Data extraction in aggregate storage is relatively faster than block storage database.

### BSO

```
[Sat Jun 04 16:57:57 2011]Local/Bcub/Bisp/admin/Info(1001103)
Regular Extractor Elapsed Time for [Data.rep] : [1.888] seconds
```

### ASO

```
[Sat Jun 04 16:59:41 2011]Local/ASOBas/Basic/admin/Info(1001103)
Regular Extractor Elapsed Time for [Data.rep] : [1.5] seconds
```

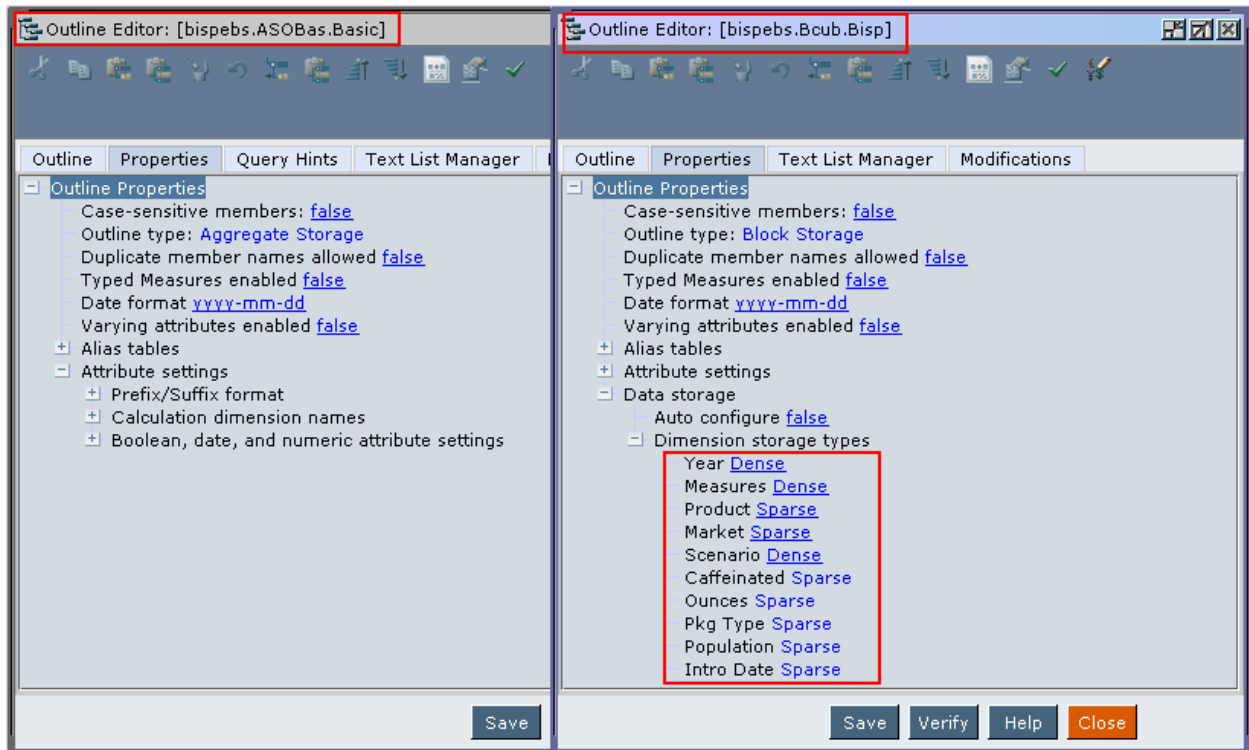
## 6. Aggregate storage dimension supports

Aggregate storage application supports more dimensions in comparison with block storage. The performance of block storage will be decrease as you increase number of dimensions in database. Aggregate storage database performance does not effects by number of dimension.

## 7. No sparse and dense dimension

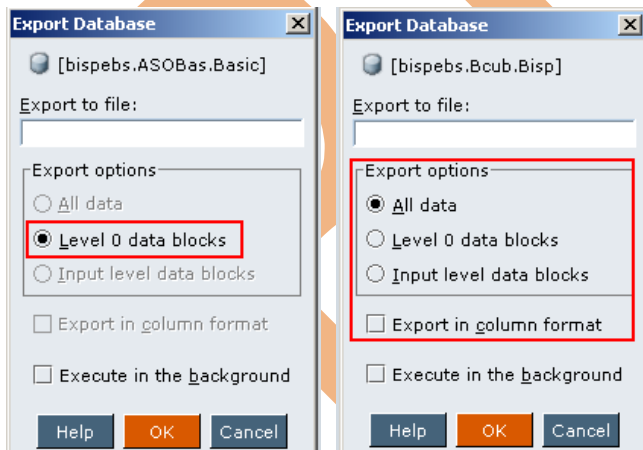
In aggregate storage application does not have dense and sparse dimension concepts.





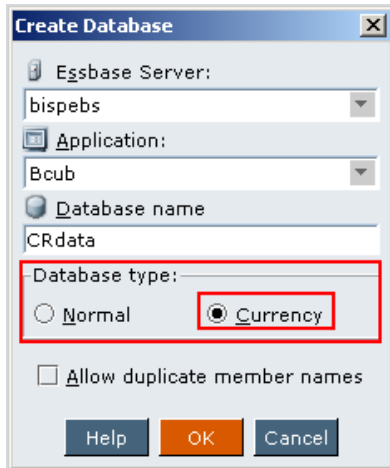
## 8. Restriction on data export

Aggregate storage database restrict to export data only for level 0 data block. Block storage allows you to use all data export options.

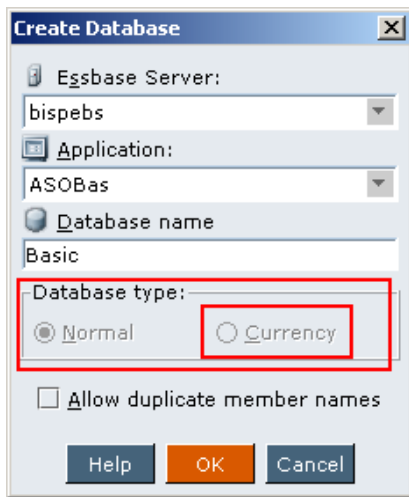


## 9. Creating currency database

You can create currency data base in block storage database.



You can't create currency data base in aggregate storage database. Because database type of currency or normal is not applicable to aggregate storage databases therefore it is not selectable.



## Aggregate Storage Overview

Aggregate storage is relatively newer the block storage application. It has additional features as compare to block storage. Aggregate storage database is aggregation-intensive cubes. It supports large numbers of dimensions and members. There is no concept of dense dimension in aggregate storage. It only supports extremely sparse data sets. Aggregate storage reduced calculation times and disk footprint and also reduced complexity in database development.

## Key Aggregate Storage Characteristics

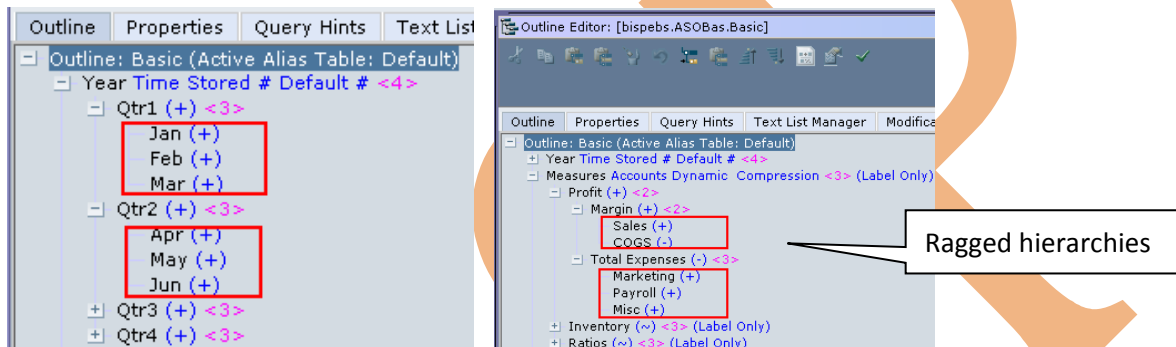
1. Data is loaded only at level 0
2. Member formulas are MDX queries
3. All formulas and aggregations are executed at runtime
4. Aggregation algorithm selects and stores most expensive queries

5. Outlines are paged
6. Block storage outlines can be converted to aggregate storage outlines
7. Hierarchy types follow formalized rules
8. Data is stored in table spaces
9. Creating Aggregate storage manually

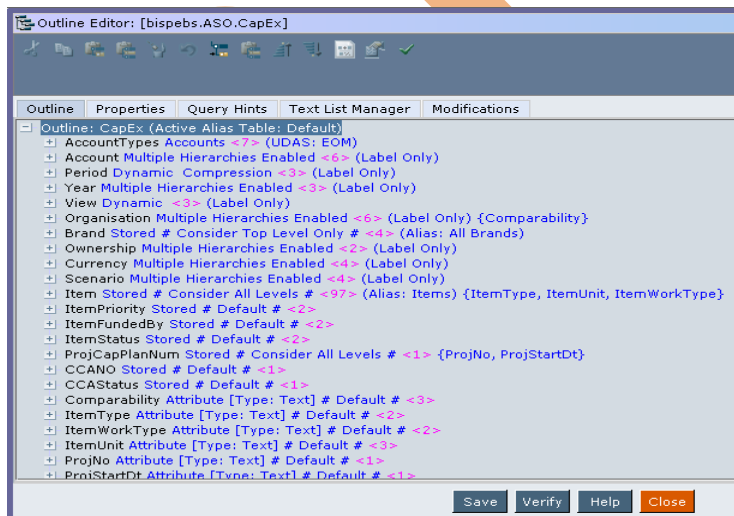
## Design Considerations

### Dimensions

**Ragged hierarchies supported-** Ragged hierarchy means it is not necessary that all members of hierarchy contain equal number of child.



**No limit to dimensions-** There is no limit on creating dimensions in aggregate storage database outline.



### Maximum level combinations

The maximum level of combinations between outline dimensions are  $2^{52}$ , which is very large. Large amount of data can be store in single database.

### Limitation on Database-

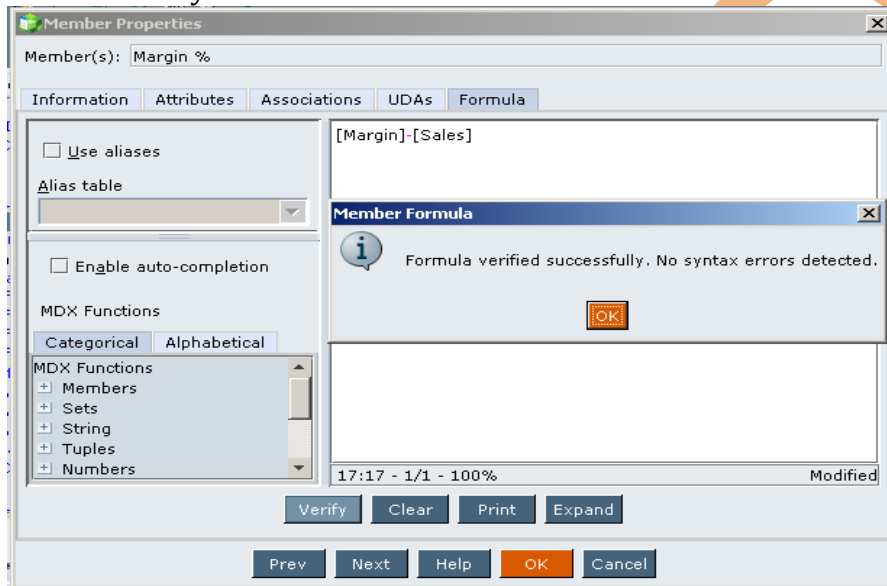
1. One database per application – Restriction for ASO application
2. MaxL commands Eecuted on application level – Because there is only one database in each application.

### 3. No currency conversion - Restriction for ASO application

## Member Formulas

When working with aggregate storage databases, you must write all member formulas in MDX. The Hyperion implementation of MDX is a customized version; it contains a series of commands that are specific to Essbase and is embedded in the MaxL shell.

Aggregate storage supports MDX, so write all member formulas in MDX. When converting an outline from block storage to aggregate storage, you may have difficulty converting block storage member formulas to MDX. You have to convert all member formulas in to MDX manually.



## Aggregate Storage Production Cycle

The production cycle for aggregate storage databases is similar as block storage database.

1. Create a database outline with database dimensions and hierarchies
2. Load data, using load rules to map to the database dimensions
3. Optional: Aggregate data by using stored or ad hoc aggregations
4. Analyze data in Excel through Smart View or Spreadsheet Add-in

Database aggregations decrease query times because many data values at upper-level intersections are calculated and stored, rather than being calculated dynamically on retrieval.

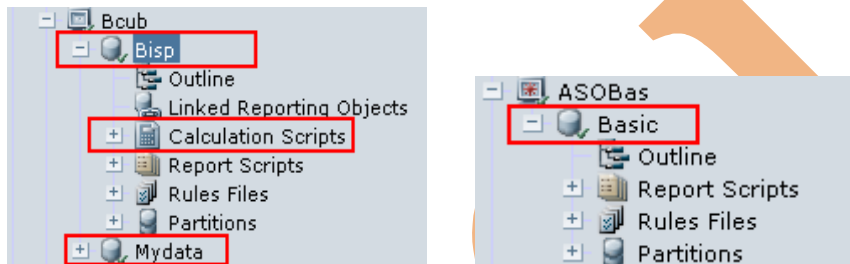
Instruction for creating aggregate storage database

1. Application and database name should be in eight characters
2. You can create only one aggregate storage database for each application

## Application and Database Trees

Block Storage application database tree has more than one database and calculation scripts.

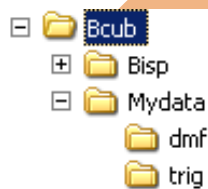
Aggregate storage application database tree has only one database and no calculation script exists.



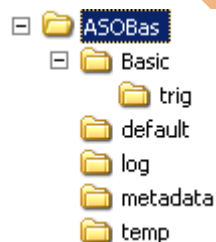
## Directory Structures

Directory contains same components in both aggregate and block storage database like outlines (OTL), load rules (RUL), and report scripts (REP). Aggregate storage databases may also contain aggregation script files (CSC).

This is sample directory structure for block storage database.



This is sample directory structure for aggregate storage database.



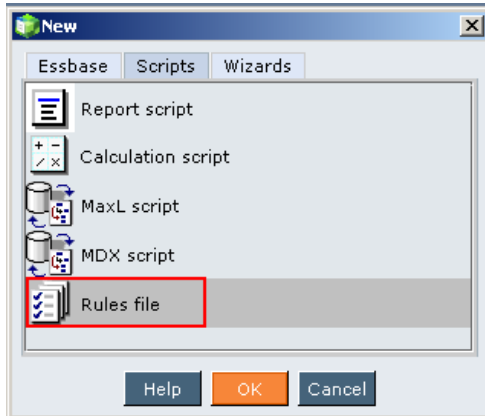
## Aggregate database objects

- a. Outlines (OTL)
- b. Load rules (RUL)
- c. Report scripts (REP)
- d. Aggregation scripts (CSC)

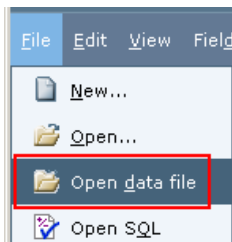
## Rules Files for Building Outlines

Creating rule file and building outline is same in aggregate storage as block storage.

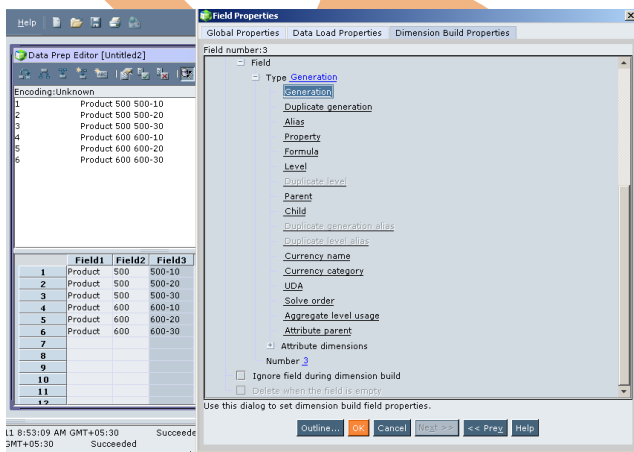
Go to file and create new rule file.



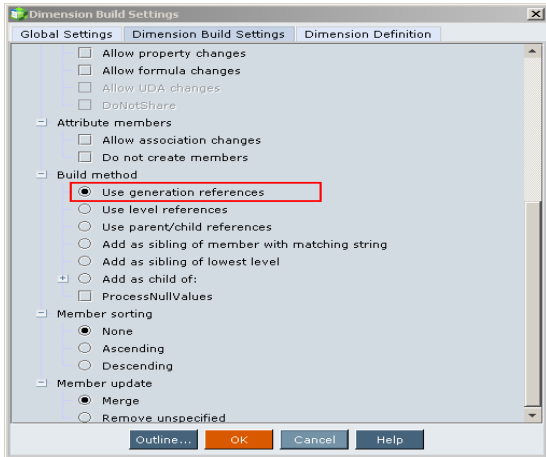
Go to file and open relative source file either text file or SQL file.



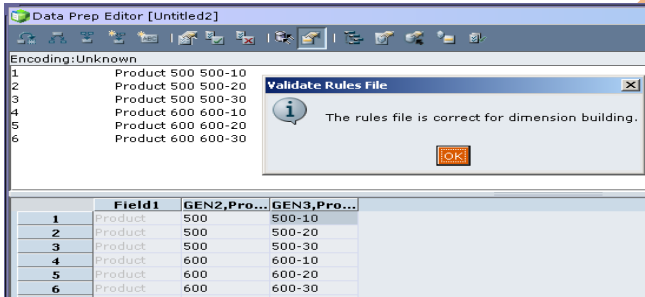
Set "Dimension Build Properties" for source file then click ok.



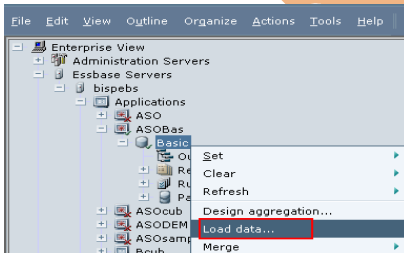
Set Dimension build settings



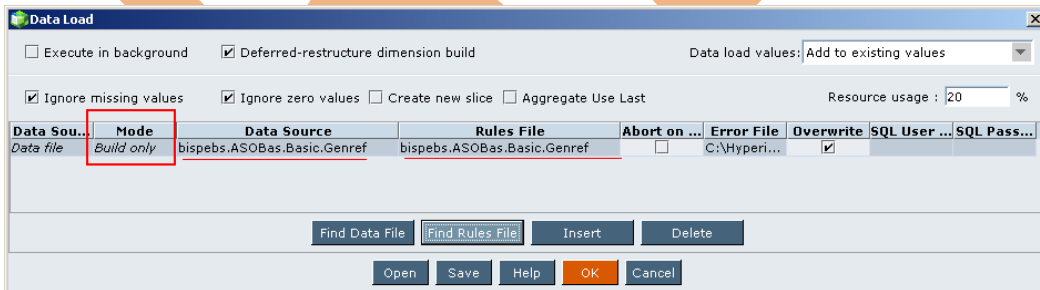
## Validate



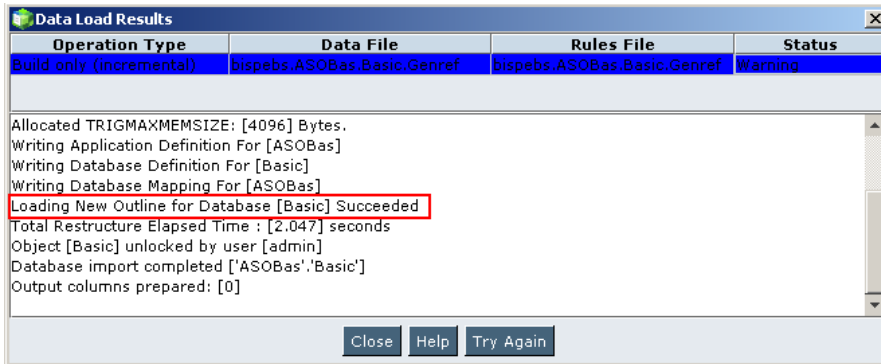
Save rule file and load data.



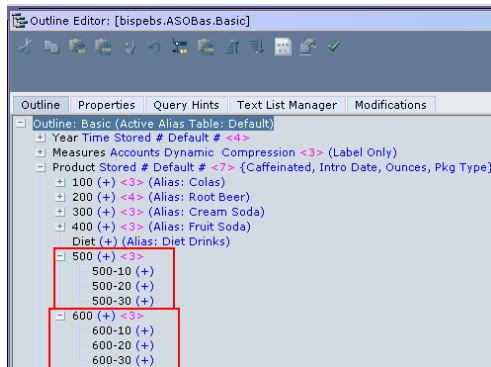
Select data load mode as "Build only" then data source and rule file click ok.



New outline dimension is loaded successfully.

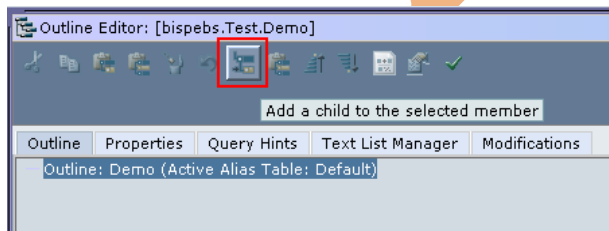


Verify in existing outline.



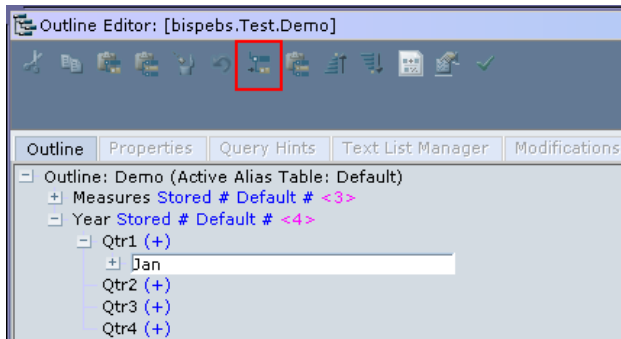
## Designing Aggregate Storage Outline Hierarchies

You can design outline manually by using toolbar. You can create new dimensions add siblings, add child and set properties through toolbar.



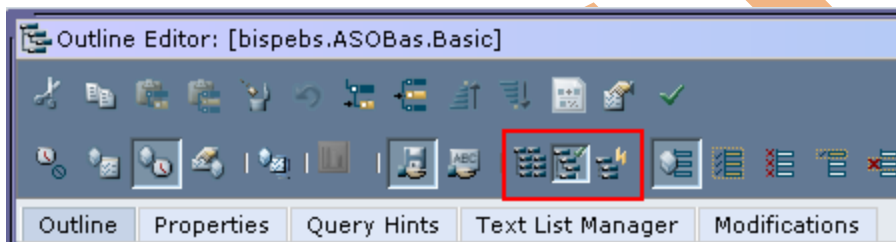
Adding Child in dimension member





There are three types of hierarchies in aggregate storage.

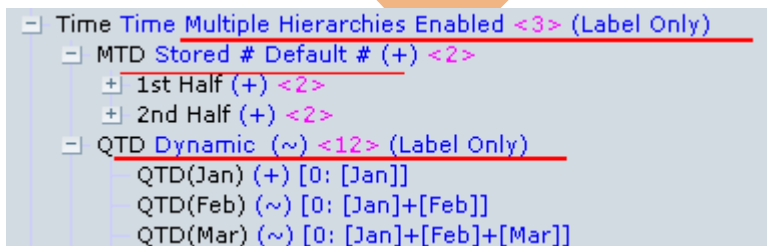
1. Multiple hierarchy
2. Stored hierarchy
3. Dynamic hierarchy



Aggregation hierarchies are structures usually comprising two or more levels of detail that must aggregate from the bottom up to provide a top-level total.

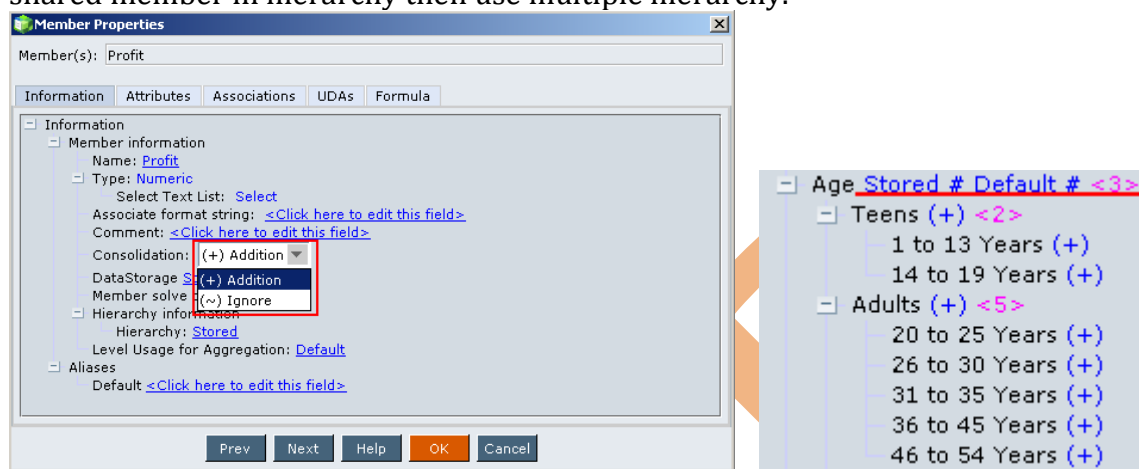
## Multiple Hierarchy

When you tag a dimension as “Multiple hierarchies enabled” the dimension member is automatically tagged as Label Only. To use multiple hierarchies in a dimension, you must enable multiple hierarchies for that dimension.



## Stored Hierarchy

Stored hierarchy has only addition as consolidation operator. You can use the stored hierarchy type where aggregation is the only mathematical requirement. If you have some shared member in hierarchy then use multiple hierarchy.



Advantages:

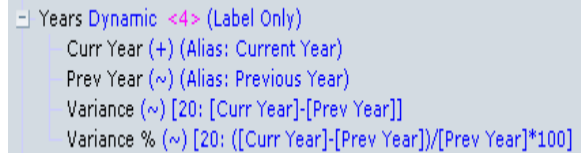
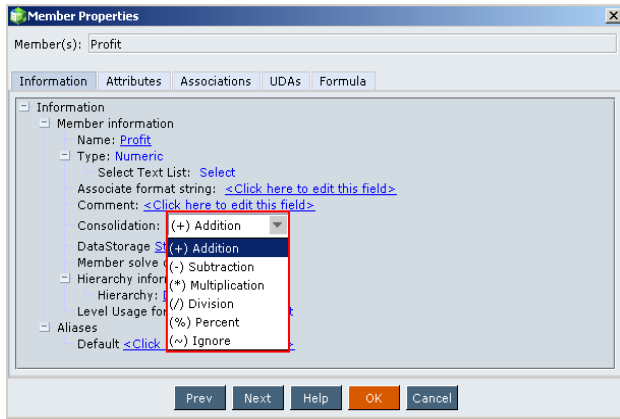
1. Potential to store aggregated data
2. Enhanced query performance

Considerations:

1. Limited use of unary operators
2. Limited use of Label Only
3. Support for only one instance
4. Dynamic Hierarchy

## Dynamic hierarchy

The Dynamic hierarchy allows you to do complex calculations and member formulas. Dynamic hierarchies are calculated, the data retrieval time may be longer than for data retrieved from stored hierarchies.



### Advantages:

1. Any consolidation operator
2. Member formulas
3. No Label Only restrictions
4. Unlimited shared members

### Considerations:

1. Members calculated during retrieval (never preaggregated)
2. Potentially reduced query performance

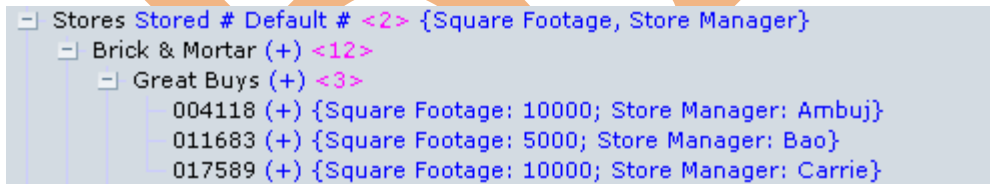
## Designing Alternate Hierarchies

### Attribute dimension hierarchy

Attribute dimension hierarchy is an alternate hierarchy used for classify additional information of dimension.

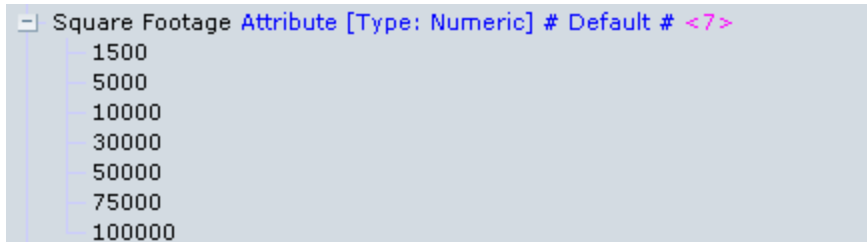
### Advantages:

1. Attribute dimension can be assign for any base dimension
2. Are treated like stored alternate hierarchies



### Considerations:

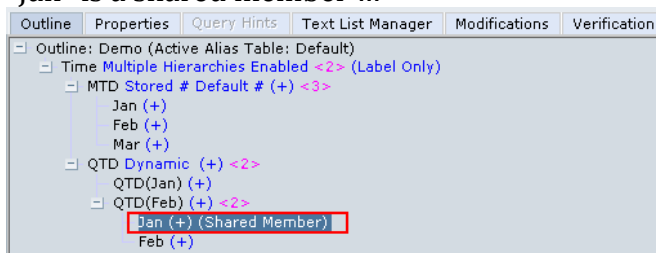
1. Can perform only addition calculations
2. Are calculated dynamically during retrieval



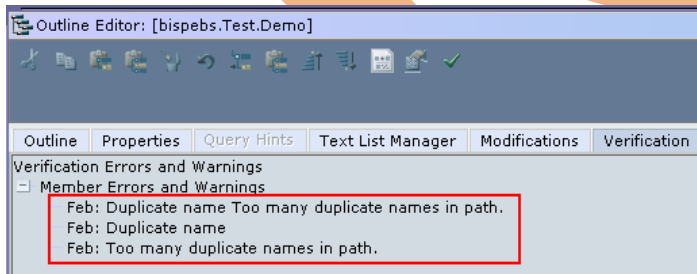
## Shared members hierarchy

Shared member hierarchy is also an alternate hierarchy all shared member refers to stored members of outline. In aggregate storage application only multiple hierarchies can have shared members.

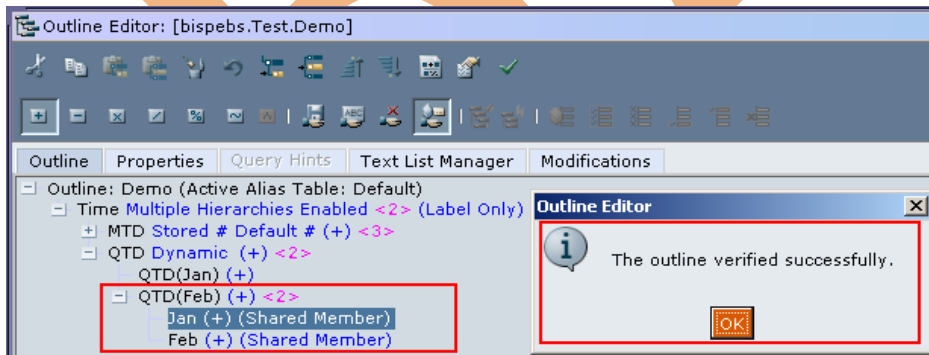
“Jan” is a shared member ...



But “Feb” is not a shared member, So Essbase will through the below error messages.

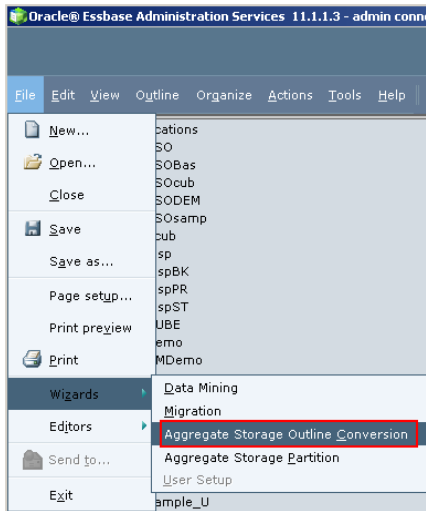


Make “Feb” as shared member and then save it.



## Converting Block Storage to Aggregate Storage

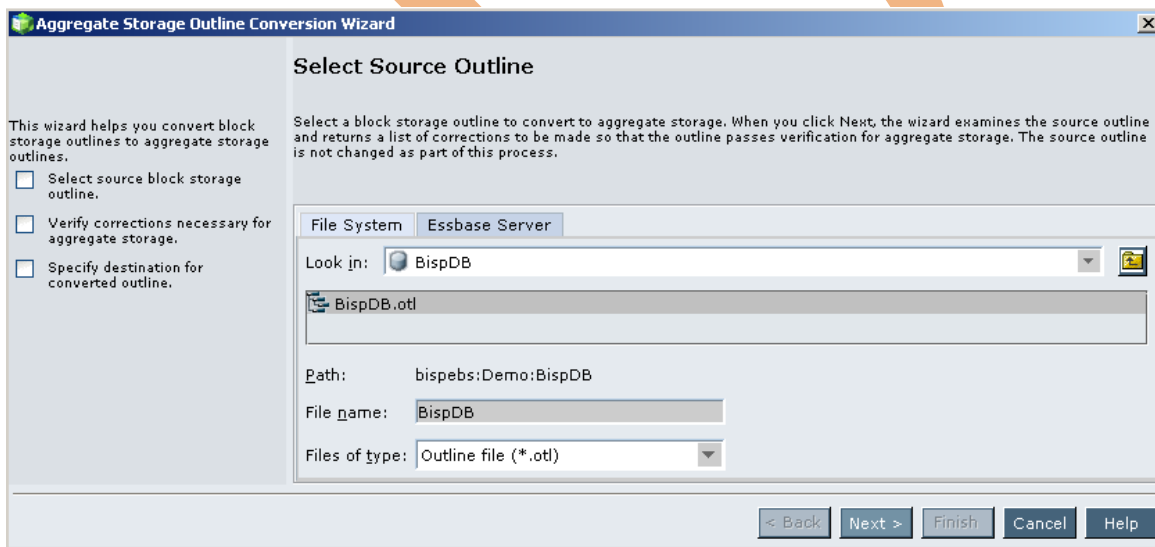
There is simple way to converting block storage application to aggregate storage application through conversion wizard. There are many difference between block storage and aggregate storage, so when you convert block storage application to aggregate storage application, wizard will reject not applicable options.



## Conversion steps for Block Storage to Aggregate Storage

1. Select a source outline
2. Verify and correct block storage-only features (either manually or automatically)
3. Select a destination for the converted outline

### Step #1 → Select Source Block storage Outline

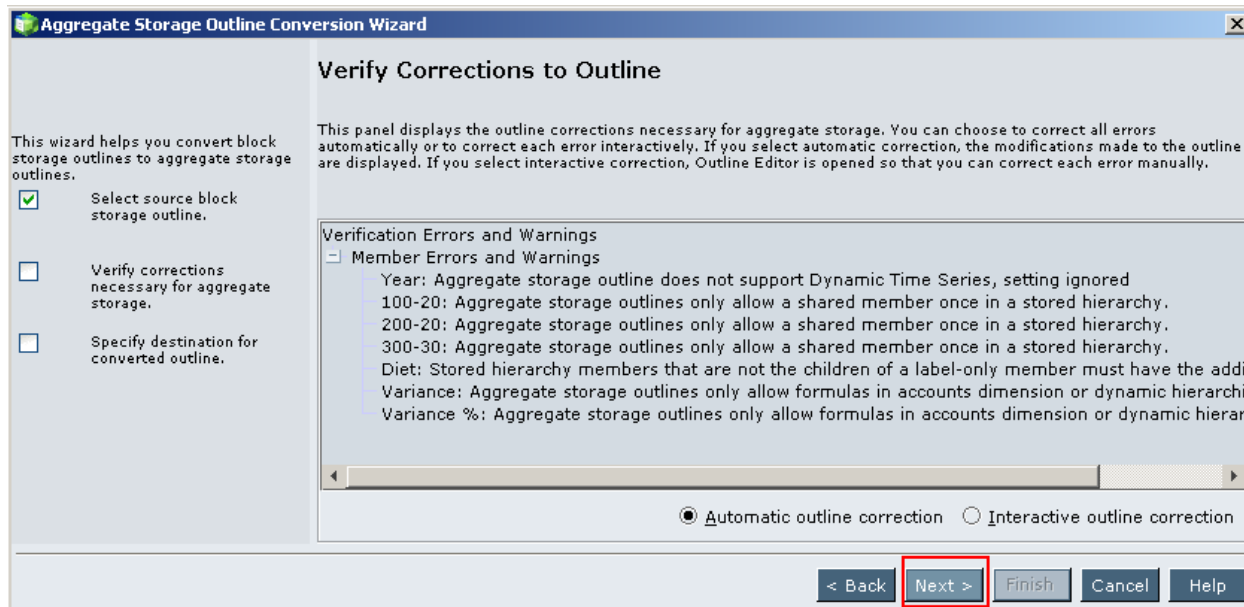


### Step #2 → Verify and correct block storage-only features

This wizard will give you the list of features which are only supported by block storage application.

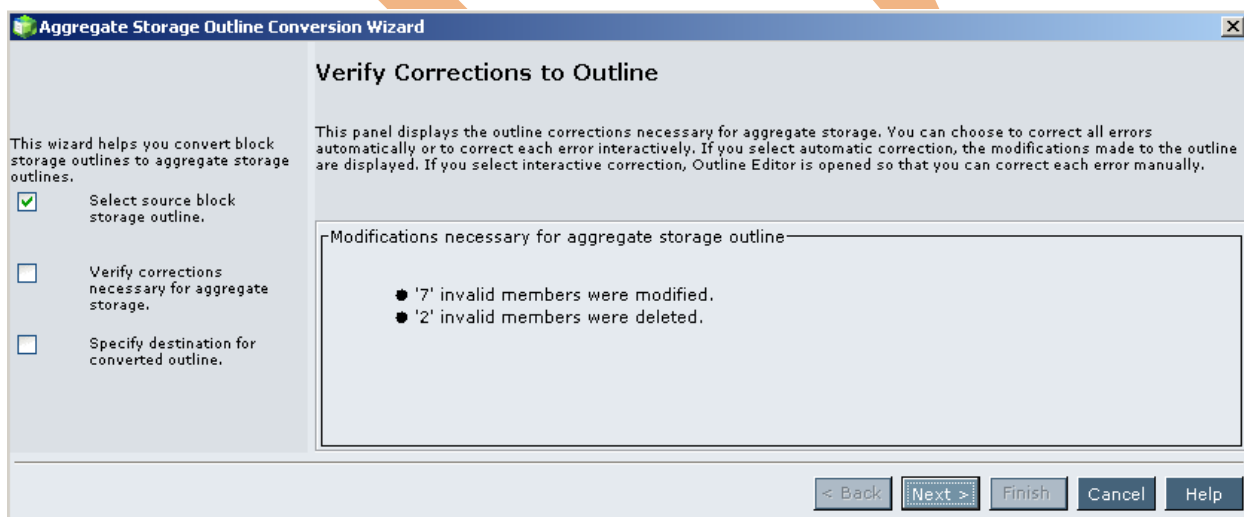
Warning comes in conversion of block storage to aggregate storage, because some properties does not support in aggregate storage. This warning information says that

shown features are not supported in aggregate storage like dynamic time series, shared member and member formula.



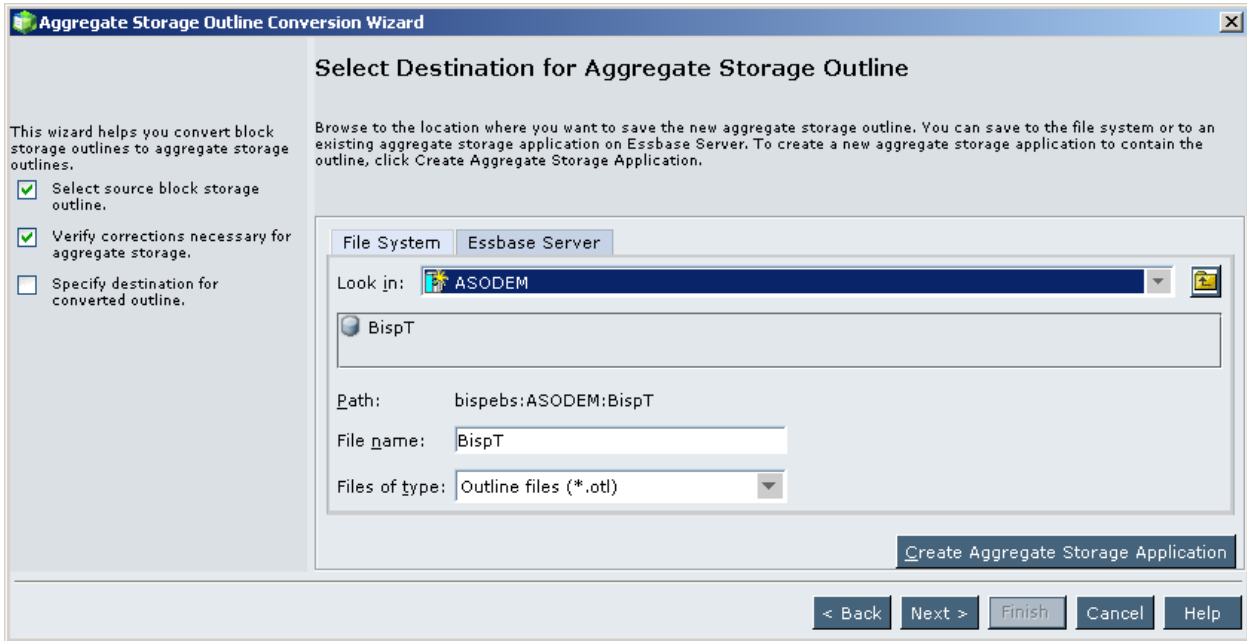
### Modification information from BSO to ASO

Conversion wizard will automatically modify some member properties and delete invalid members.

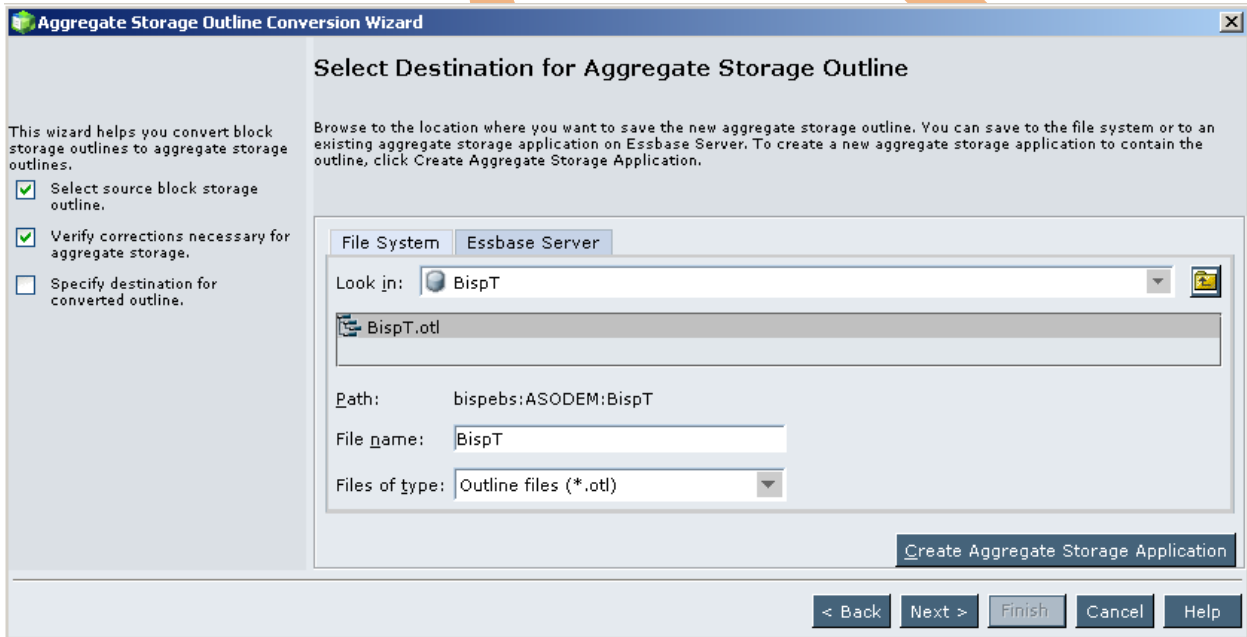


### Step #3 → Select Target Aggregate Storage Application

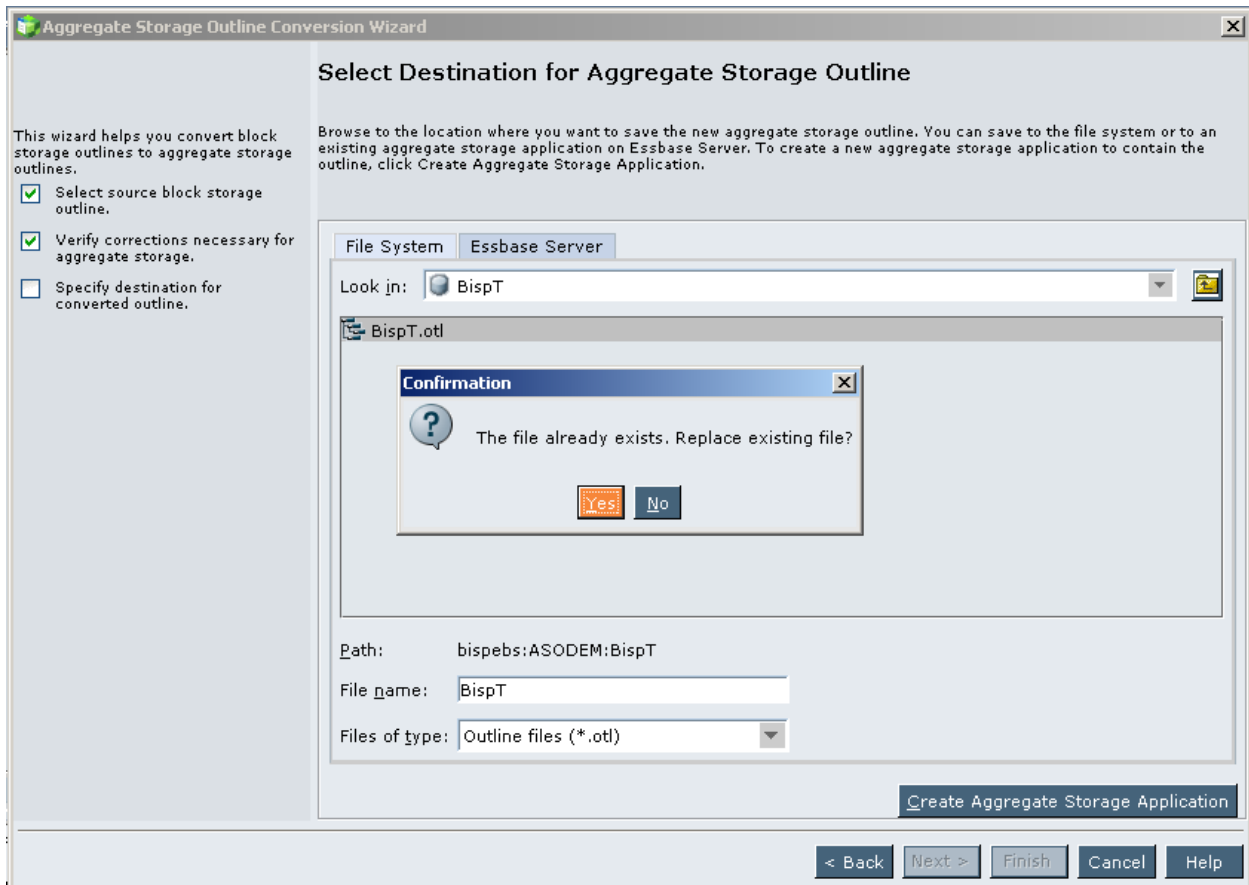
You can select target application and database outline then replace the existing outline from the new one. You also can create new aggregate storage application and convert block storage to aggregate storage.



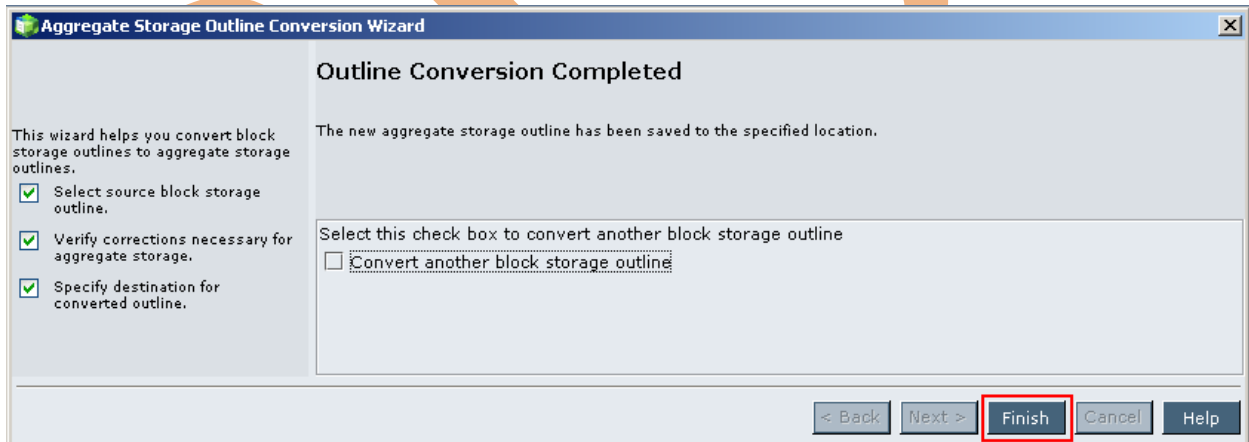
## Select Outline



Select and replace the existing outline



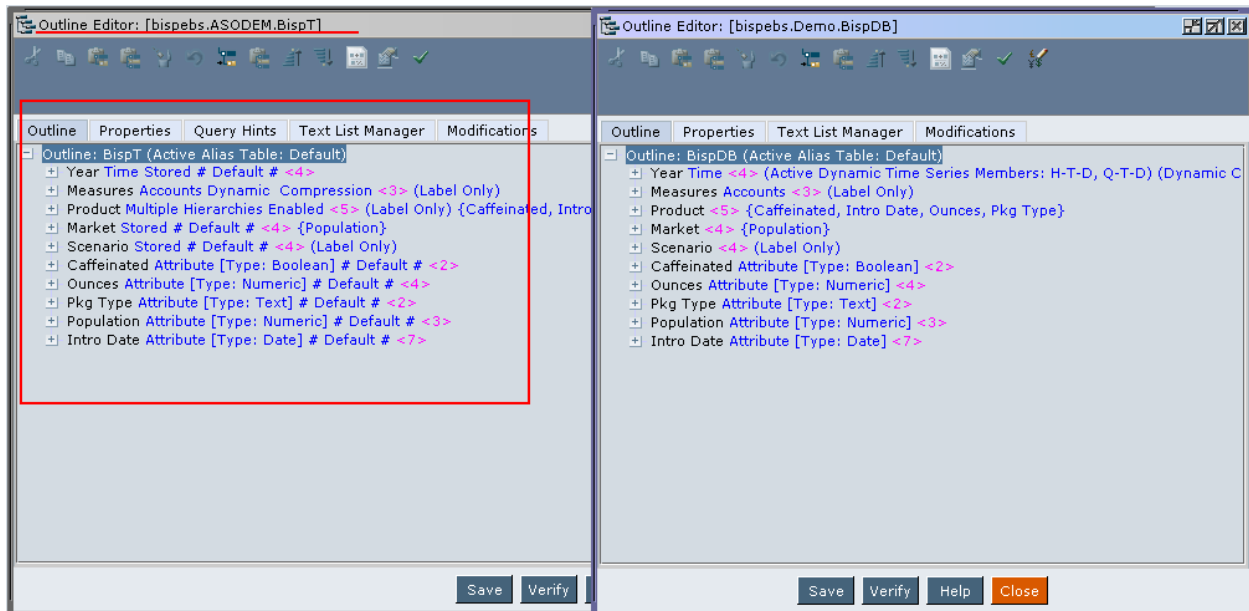
Click on finish.....



## Converted Block Storage Application

Block storage application successfully converted into aggregate storage application.





The unsupported features replaced by supported features.

- 1) Year dimension is converted from dynamic to storage
- 2) Measures dimension hierarchy converted as dynamic
- 3) Product dimension storage hierarchy converted as Multiple Hierarchy
- 4) All member formulas are rejected