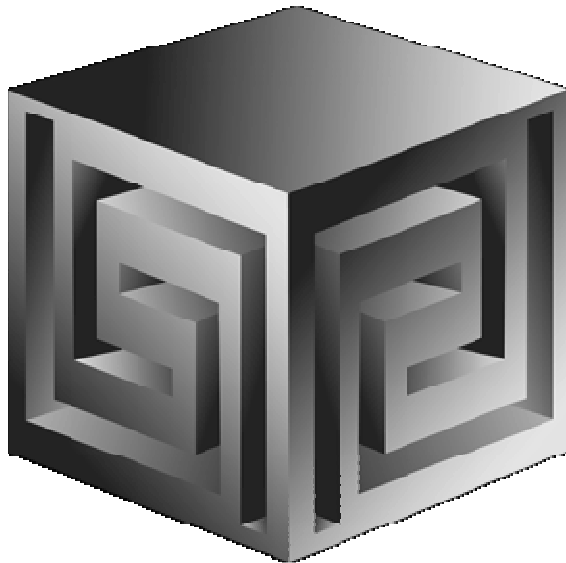


Implementing Oracle BI EE Using Oracle OLAP Cubes

**Dallas Oracle User Group
Oct 12, 2007**



**Cathye Pendley & Mark Thompson
cpendley or mthompson @vlamis.com**

Vlamis Software Solutions, Inc.

816-781-2880

<http://www.vlamis.com>

Copyright © 2007, Vlamis Software Solutions, Inc.



Vlami Software Solutions, Inc.

- **Founded in 1992 in Kansas City, Missouri**
- **Oracle Partner and reseller since 1995**
- **Specializes in ORACLE-based:**
 - ☐ **Data Warehousing**
 - ☐ **Business Intelligence**
 - ☐ **Data Transformation (ETL)**
 - ☐ **Web development and portals**
 - ☐ **Oracle OLAP-based applications**
- **Delivers**
 - ☐ **Design and integrate BI and DW solutions**
 - ☐ **Training and mentoring**
- **Expert presenters at major Oracle conferences**



Biography

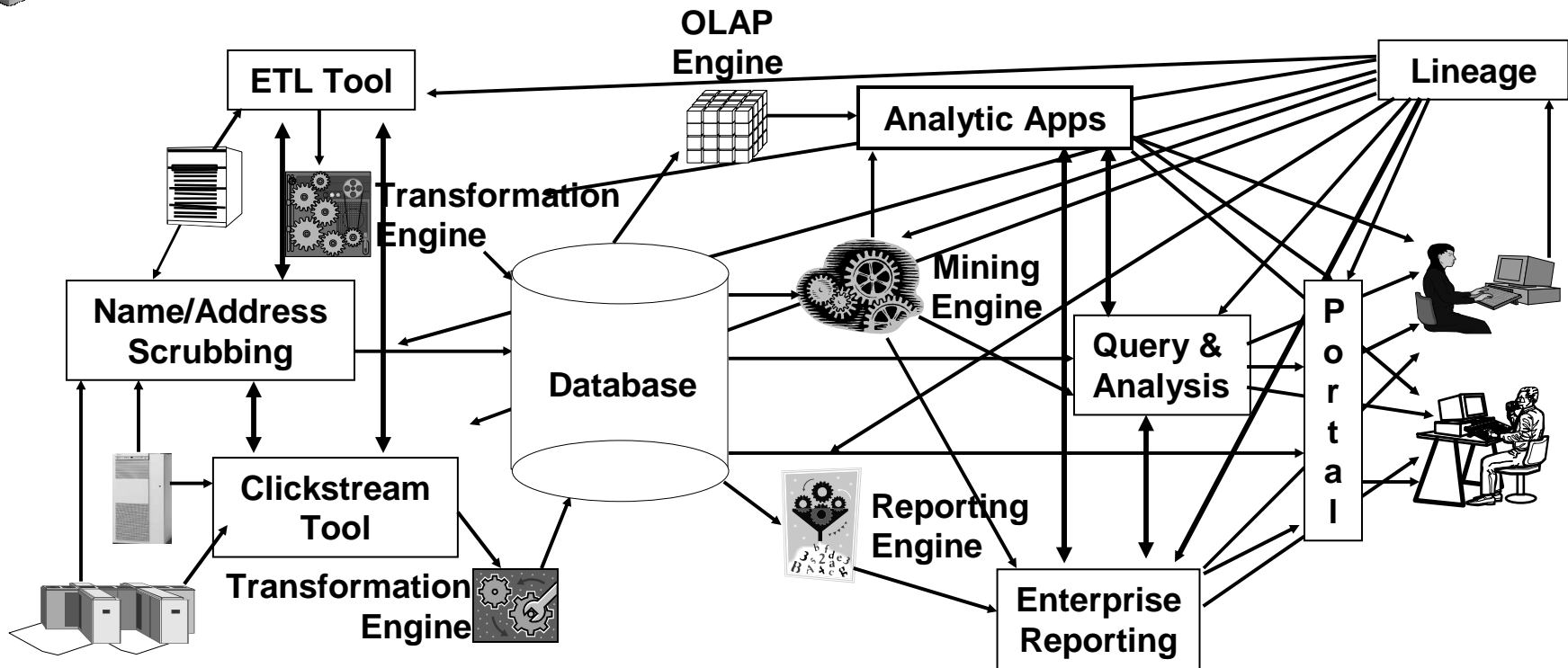
- **Mark Thompson, Senior Consultant**
 - ❑ 23 years working with Oracle OLAP and its predecessor (Express)
 - ❑ 5 years with Oracle Consulting and Braun Consulting
 - ❑ Joined Vlamis Software Solutions in 2006
 - ❑ Presenter at numerous Oracle user conferences
 - ❑ Author of several white papers on OLAP
 - ❑ OLAP tools and DML expert



Agenda

- **Brief background of BI EE and Oracle OLAP**
- **Why Oracle OLAP?**
- **What is Oracle OLAP?**
- **Oracle OLAP storage options**
- **Structure of Analytic Workspace**
- **Demonstration of BI EE on Oracle OLAP**
- **11g OLAP Preview – what changes?**

Business Intelligence Market Multi-Vendor, Un-integrated



- Protracted and complex implementation
- Escalating maintenance costs
- Software ***and Metadata*** Integration is key!



Oracle BI Suite Enterprise Edition

Unified Business Intelligence Infrastructure

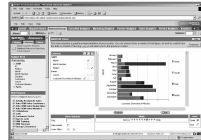
**Interactive
Dashboards**



**Reporting &
Publishing**



**Ad-hoc
Analysis**



**Proactive
Detection
and Alerts**



**Disconnected
Analytics**



**MS Office
Plug-in**



Simplified Business Model and Abstraction Layer

**Oracle
BI Server**

Intelligent Caching Services

Multidimensional Calculation and Integration Engine

Intelligent Request Generation and Optimized Data Access Services



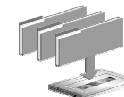
**OLTP & ODS
Systems**



**Data Warehouse
Data Mart**



**SAP, Oracle
PeopleSoft, Siebel,
Custom Apps**



**Files
Excel
XML**



**Business
Process**



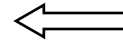
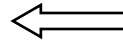
Changing Oracle BI Product Line

- **Back Ends**
 - ☐ Oracle relational (and Disco Administrator)
 - ☐ Oracle OLAP cubes
 - ☐ Heterogeneous sources (MS, SAP BW, etc.)

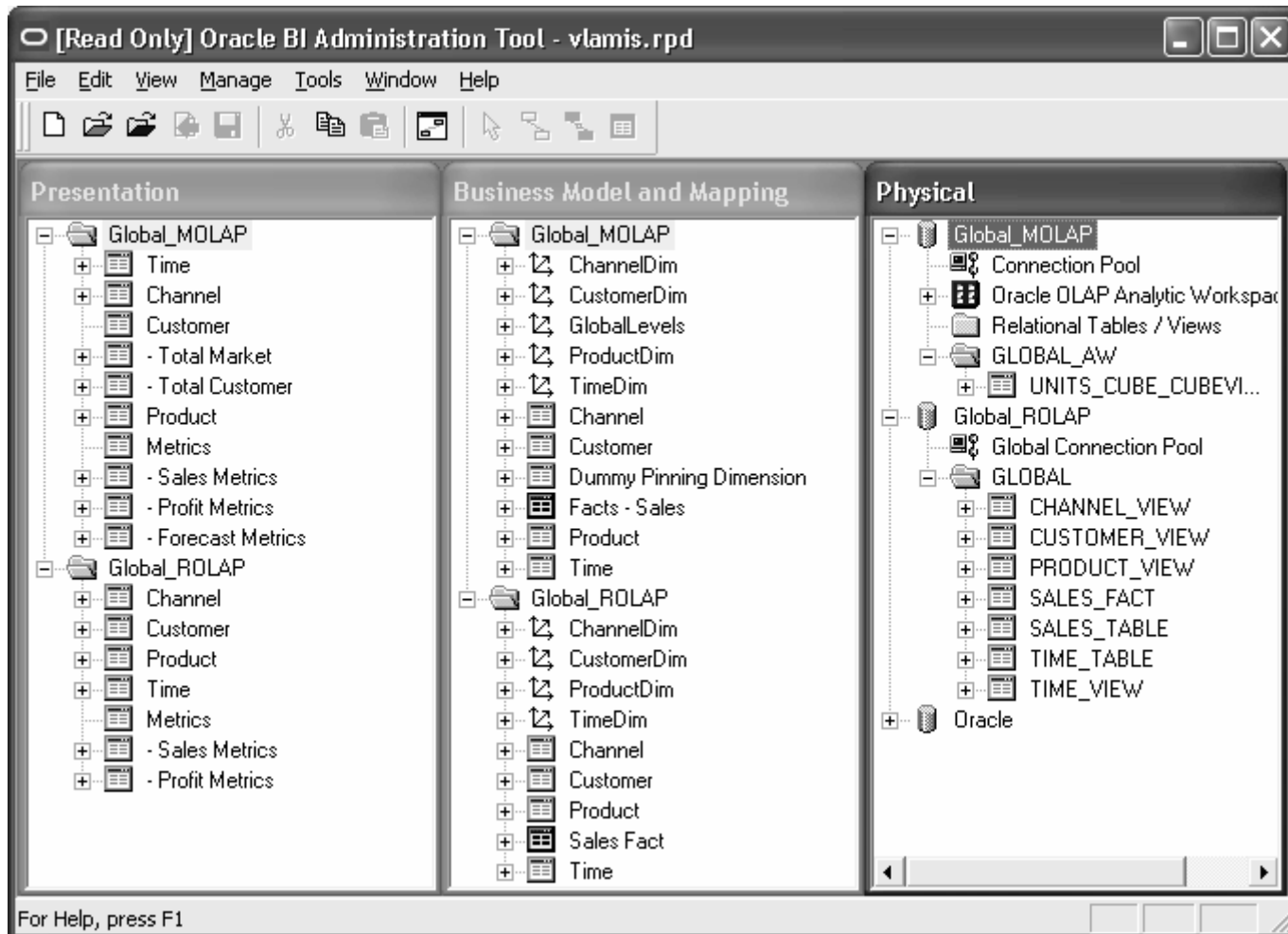
- **Front Ends**
 - ☐ Oracle BI EE (Siebel)
 - ☐ Oracle BI SE (Discoverer, BI Beans)
 - ☐ Oracle BI SE One (stripped down Siebel)



BI EE Metadata Editor



Data Flows





Definition of OLAP

OLAP stands for On Line Analytical Processing.

That has two immediate consequences: the *on line* part requires the answers of queries to be fast, the *analytical* part is a hint that the queries itself are complex.

i.e. Complex Questions with FAST ANSWERS!



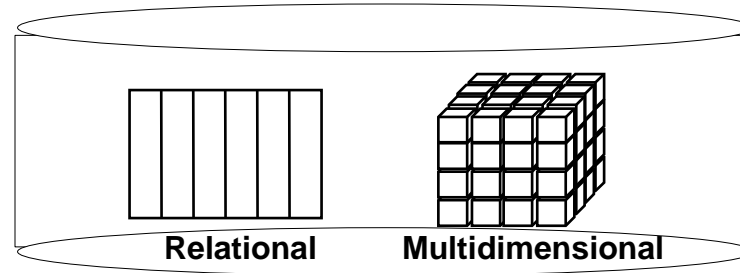
What Does Oracle OLAP Add to a DW?

- **Multidimensional user view of data**
- **Users create own reports**
- **Users create own measures**
- **Easy drill-down, rotate**
- **Iterative discovery process (not just reports)**
- **Ad-hoc analysis**
- **Easy selection of data with business terms**
- **OLAP DML with what-if, forecasting**

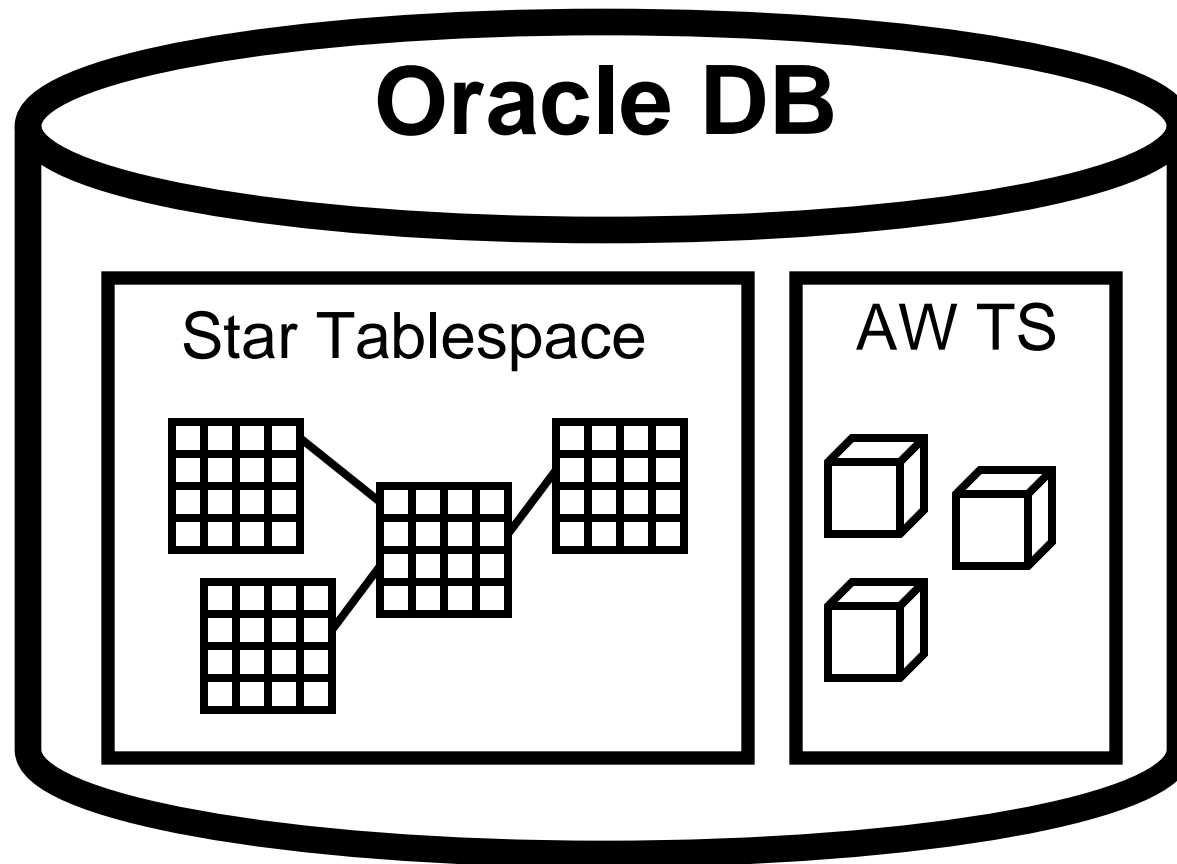


ROLAP vs. MOLAP

- What is ROLAP? (Relational)
- What is MOLAP? (Multidimensional)
- It's all in how the data is stored



Analytic Workspaces Are Stored in Tablespaces in OLAP





What is an Analytic Workspace?

Oracle Enterprise Manager Console

File Navigator Object Tools Configuration Help

ORACLE Enterprise Manager

GLOBAL

GLOBAL_AW

Tables

AW\$GLOBAL

Indexes

Materialized View

Partitions

Triggers

DATE_TAB

Indexes

Views

Synonyms

Sequences

Clusters

Source Types

User Types

HR

General Constraints Storage Options LOB Storage Statistics

Name: AW\$GLOBAL

Schema: GLOBAL_AW

Tablespace: GLOBAL_AW

Table: ☒ Standard ☐ Organized Using Index (IOT)

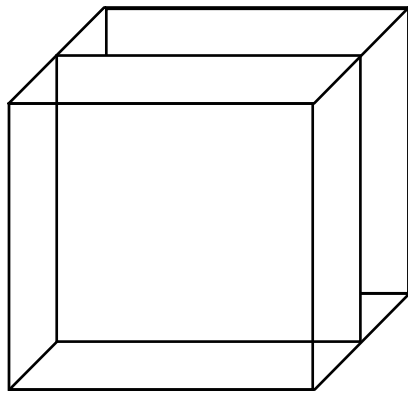
Columns

Name	Datatype	Size	Scale	Nulls?
PS#	NUMBER	10	0	✓
GEN#	NUMBER	10	0	✓
EXTNUM	NUMBER	8	0	✓
AWLOB	BLOB			✓
OBJNAME	VARCHAR2	60		✓
PARTNAME	VARCHAR2	60		✓

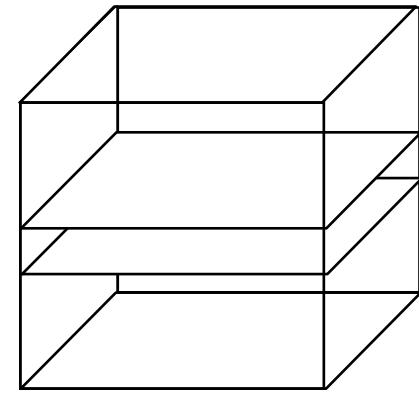
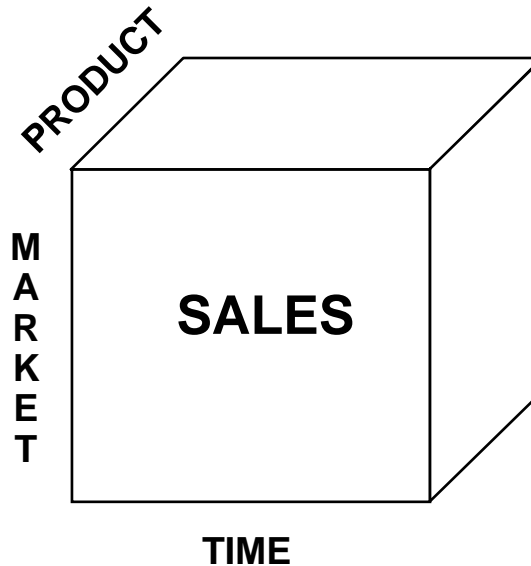


OLAP AW Stores Data in Cubes

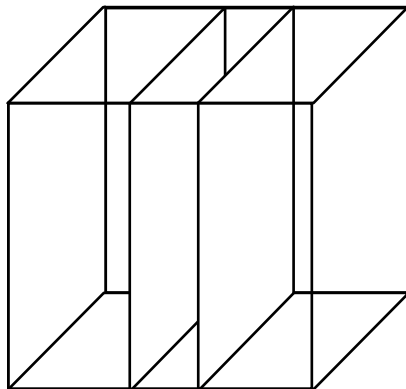
Fast Flexible Access to Summarized Data



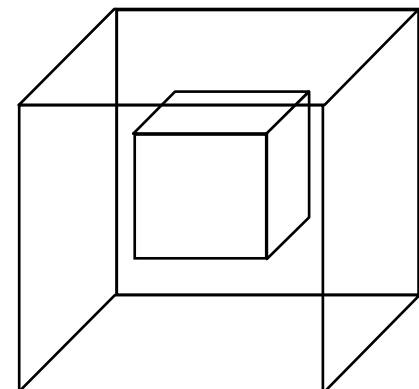
Product Mgr. View



Regional Mgr. View



Financial Mgr. View



Ad Hoc View

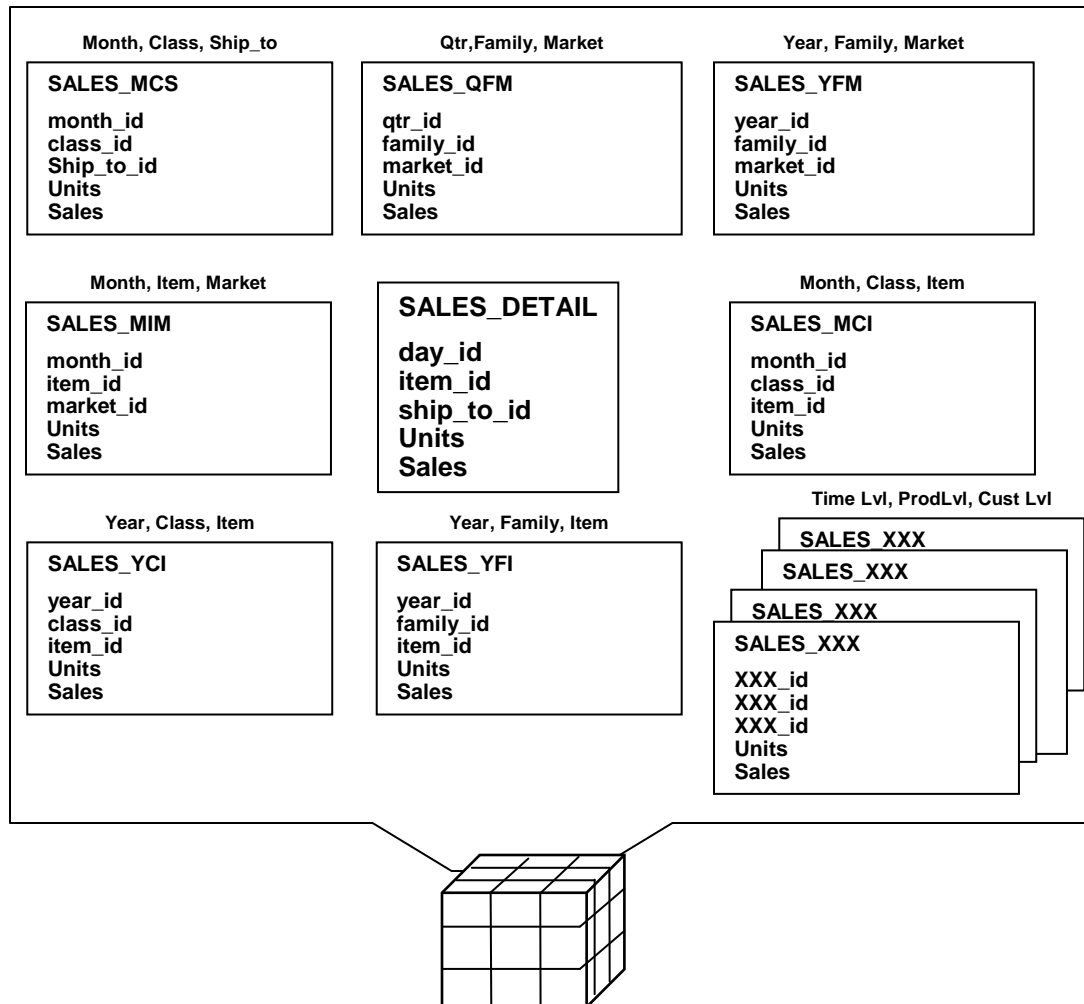


Cubes Defined

- **Definition:**
- **Cubes are collections of measures. They are a logical way to organize data. All measures in a cube share the same dimensionality, sparsity, and data type**
- **Examples:**
 - ❑ **Sales_Cube (with Units, Dollars, Profit)**
 - ❑ **Finance_Cube (with Actual, Budget, Variance)**



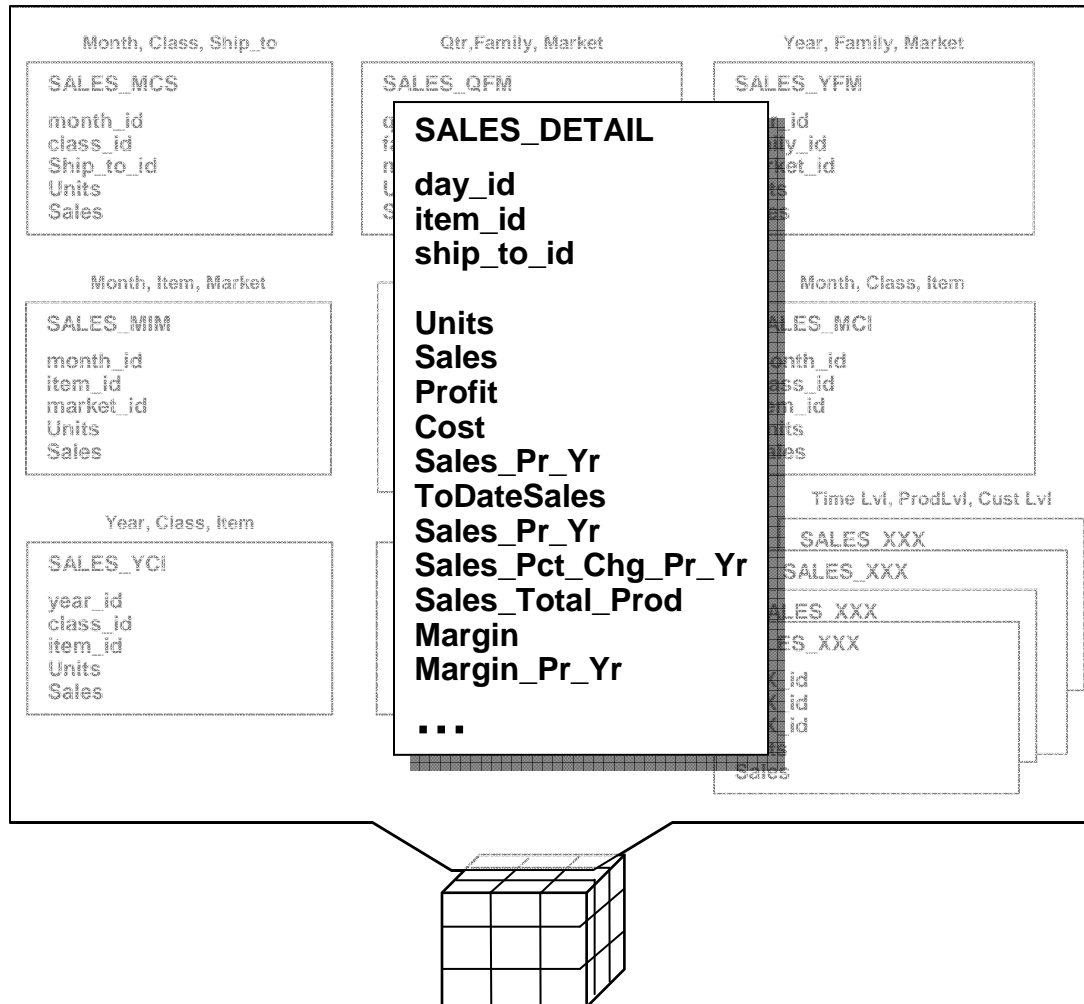
Classic ROLAP Approach



- Define appropriate summaries based on query patterns
- Each summary is typically defined at a particular grain
 - ☐ Month, Class, Ship_to
 - ☐ Qtr, Family, Market
 - ☐ Year, Family, Market
 - ☐ etc.

Summary Strategies

Classic ROLAP approach



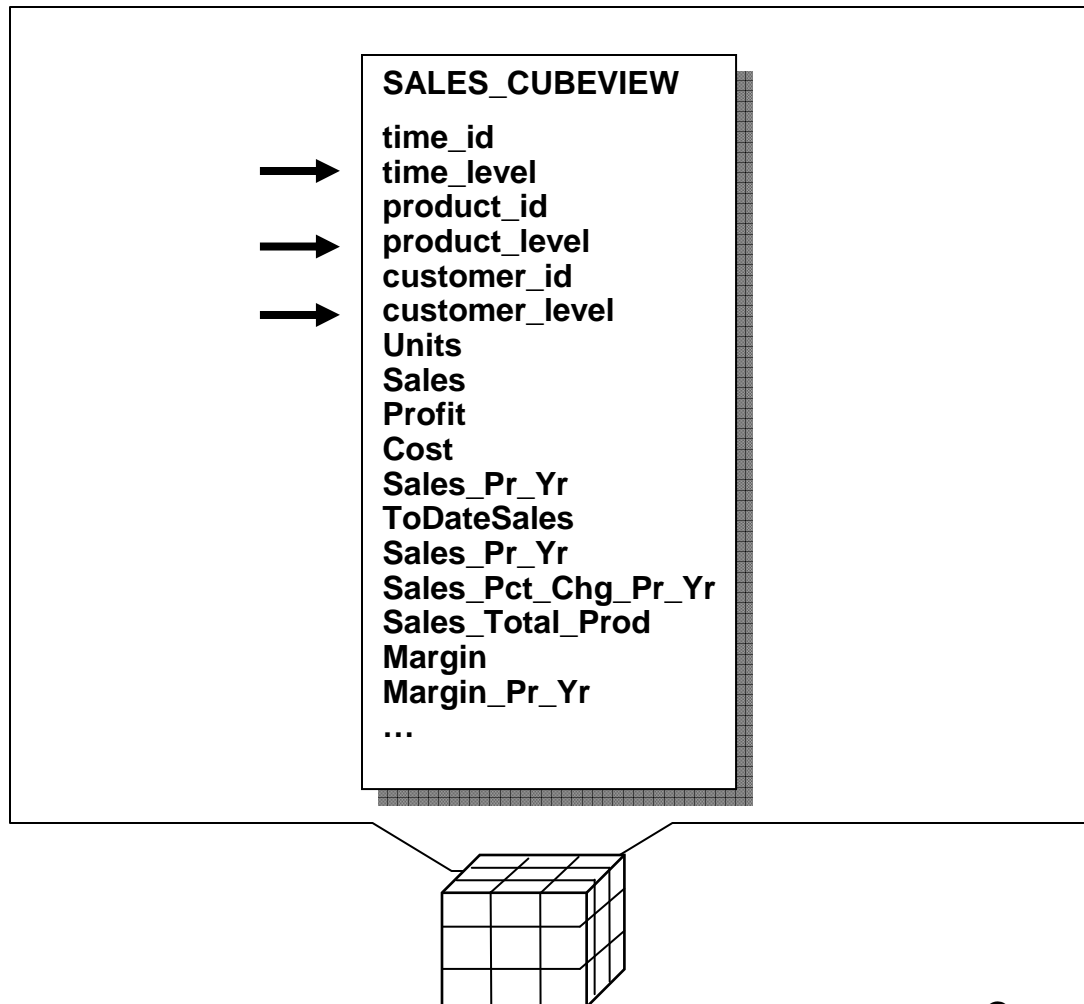
- Most OLAP calculations do not aggregate easily
 - ☐ Percentages
 - ☐ Ratios
 - ☐ Moving Averages
 - ☐ Etc.
 - Aggregation rules may be complex
 - Difficult to handle case where user queries a grain that is not supported by a view
- Summary view required for each grain that will be queried

Is this manageable?



MOLAP Approach:

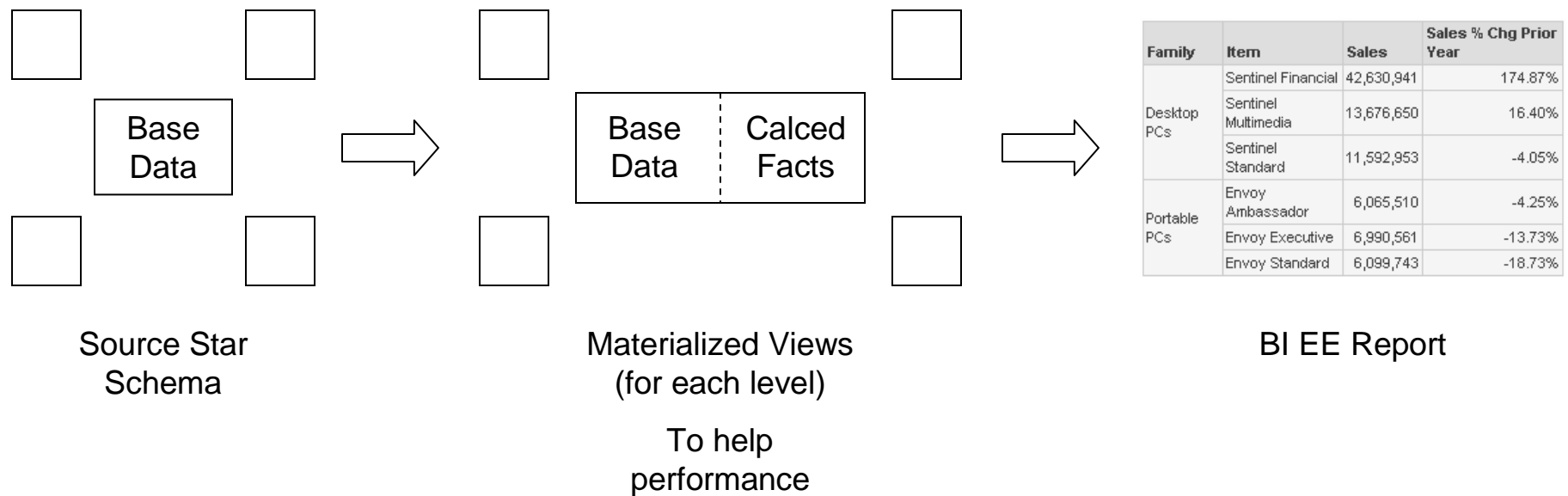
Single embedded total view for aggregations



- Single view contains data for *all* summary levels
- Multi-dimensionally *based* Oracle OLAP engine handles all of the calculations
- No Summary Tables or Materialized Views

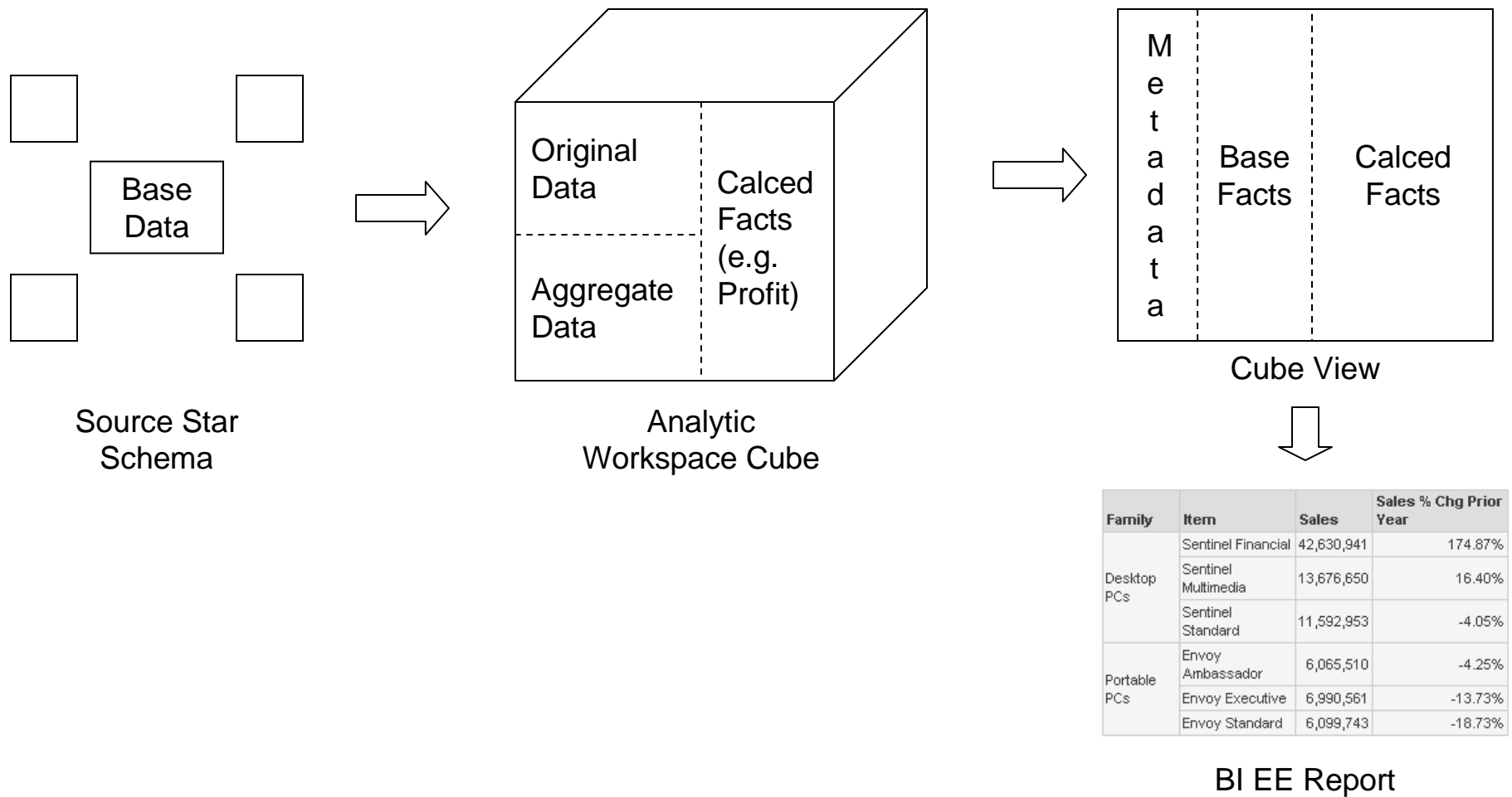


BI EE on ROLAP



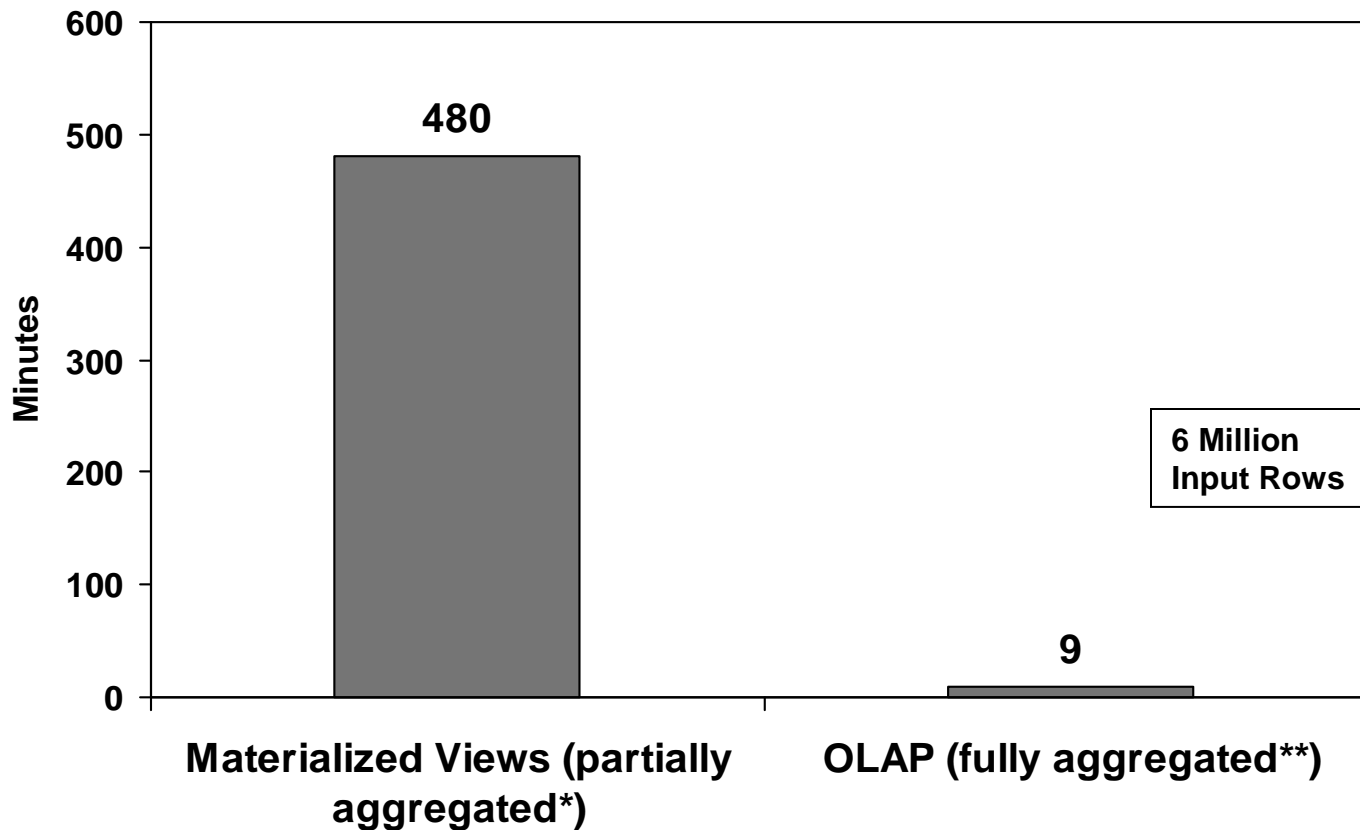


BI EE on MOLAP



Oracle Performance Case Study

Oracle Applications: Finance DBI



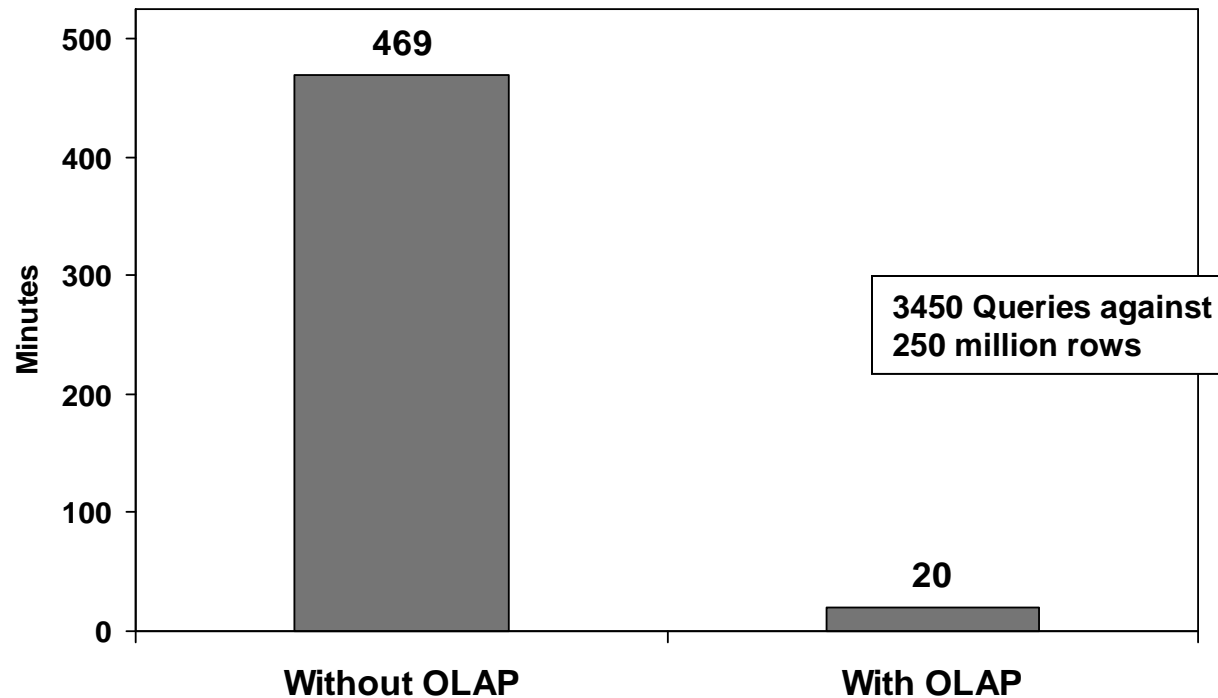
* MV aggregated 1 dimension and 1 measure

** OLAP aggregated 7 dimensions and 11 measures



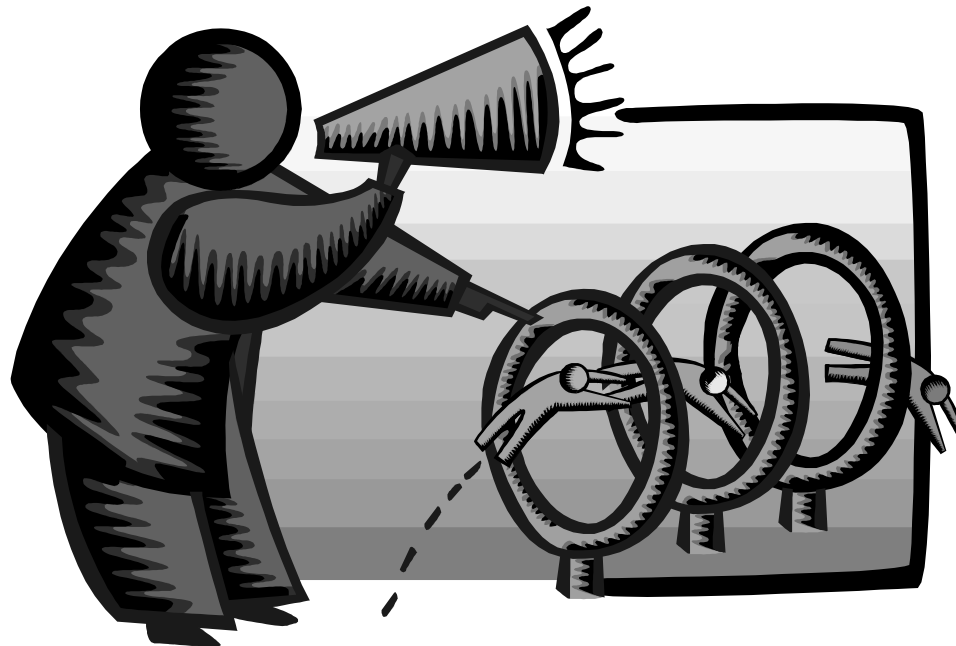
Oracle Performance Case Study

Ad Hoc Queries Across Summary Levels



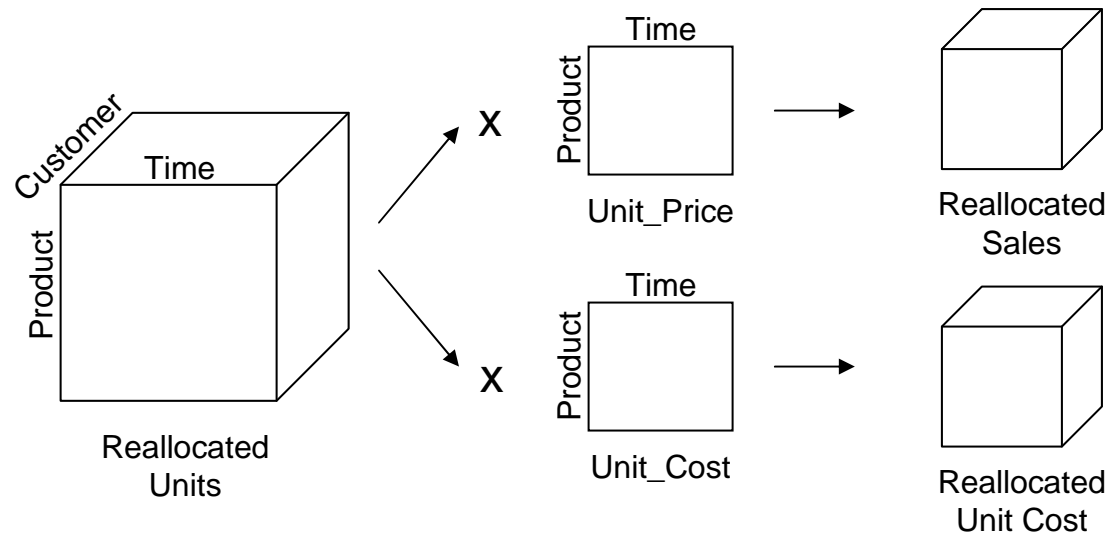
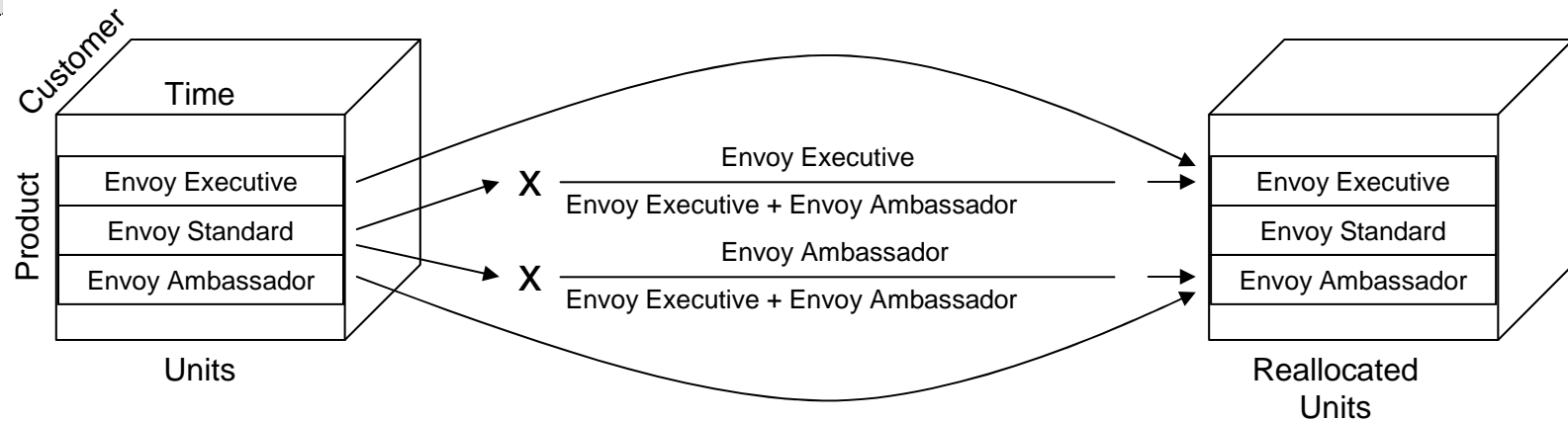


Demo of BI EE on Oracle OLAP

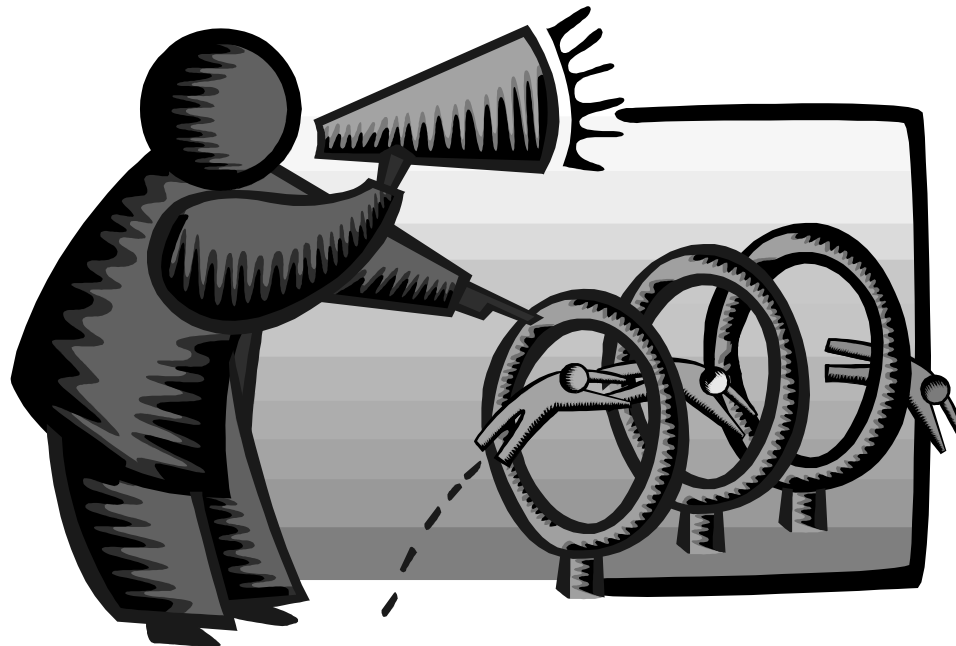




Reallocate Unit Sales



This concludes the demo!





Sneak Peek at BETA OLAP 11g

- Oracle 11g is currently in Beta, and Oracle OLAP has many NEW things Coming!
 - ☐ New CUBE_TABLE function in SQL
 - ☐ Tight integration with SQL
(automatically generated views)
 - ☐ Tight integration with data dictionary
 - ☐ New Calc Wizard in AWM!
 - ☐ Easier to use and deploy
 - ☐ Ability to use OLAP for Materialized views
(get MUCH FASTER response times!)



OLAP 11g Changes

- New CUBE_TABLE function simplifies access to AW data (replacing OLAP_TABLE)

The screenshot shows the Oracle SQL Developer interface. On the left, the 'Connections' tree is expanded to show the 'stack07 - global - main3' connection, with the 'Views' folder expanded to show 'CUSTOMER_SHIPMENTS_VIEW'. The main window displays the 'Enter SQL Statement' editor with the following query:

```
SELECT * FROM TABLE(CUBE_TABLE('GLOBAL.CUSTOMER;SHIPMENTS'));
```

Below the editor, the 'Results' tab is active, showing a table with 6 columns: REGION, WAREHOUSE, SHIP_TO, LEVEL_NAME, and LONG_DESCRIPTOR. The table contains 4 rows of data.

	REGION	WAREHOUSE	SHIP_TO	LEVEL_NAME	LONG_DESCRIPTOR
1	(null)	(null)	REGION	Europe	
2	(null)	(null)	REGION	North America	
3	(null)	(null)	REGION	Asia Pacific	
4	20	99	SHIP_TO	UK Env Dept Glasgow	

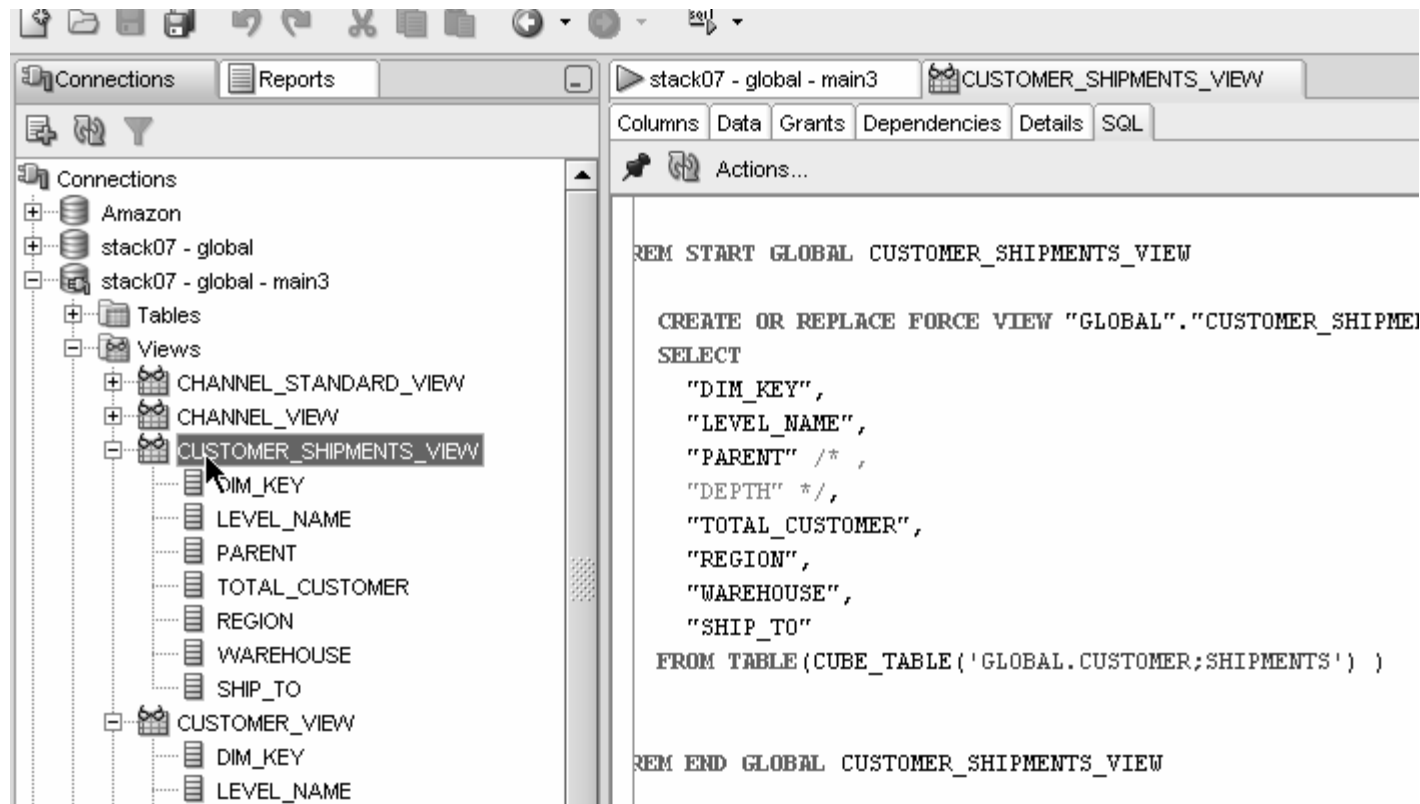
Standard Disclaimer – Beta software! No promises!

Copyright © 2007, Vlamis Software Solutions, Inc.



OLAP 11g Changes

- Views automatically created for SQL access to AWs – Dimensions and Cubes!



Standard Disclaimer – Beta software! No promises!

Copyright © 2007, Vlamis Software Solutions, Inc.



OLAP 11g Changes

- Views easily accessed from SQL Developer

The screenshot shows the SQL Developer interface. On the left, the 'Connections' pane shows a tree structure with 'Amazon' > 'stack07 - global' > 'stack07 - global - main3' > 'Views' > 'CUSTOMER_SHIPMENTS_VIEW'. The main pane shows the 'Data' tab for this view, displaying a table with 8 columns: DIM_KEY, LEVEL_NAME, PARENT, TOTAL_CUSTOMER, REGION, WAREHOUSE, and SHIP_TO. The table contains 16 rows of data.

DIM_KEY	LEVEL_NAME	PARENT	TOTAL_CUSTOMER	REGION	WAREHOUSE	SHIP_TO
1 9	REGION	1	1	9	(null)	(null)
2 10	REGION	1	1	10	(null)	(null)
3 8	REGION	1	1	8	(null)	(null)
4 99	SHIP_TO	20	1	9	20	99
5 46	SHIP_TO	21	1	10	21	46
6 89	SHIP_TO	21	1	10	21	89
7 59	SHIP_TO	21	1	10	21	59
8 91	SHIP_TO	20	1	9	20	91
9 90	SHIP_TO	21	1	10	21	90
10 49	SHIP_TO	16	1	9	16	49
11 95	SHIP_TO	21	1	10	21	95
12 72	SHIP_TO	11	1	8	11	72
13 47	SHIP_TO	14	1	9	14	47
14 60	SHIP_TO	18	1	8	18	60
15 74	SHIP_TO	15	1	8	15	74
16 75	SHIP_TO	16	1	9	16	75

Standard Disclaimer – Beta software! No promises!

Copyright © 2007, Vlamis Software Solutions, Inc.



OLAP 11g Changes

- Automatic views accessible from AWM

The screenshot displays the Oracle OLAP 11g interface. On the left, the 'abases' tree shows the hierarchy: stack07 (global) > Schemas > GLOBAL > Analytic Workspaces > GLOBAL (attached RW) > Dimensions > CHANNEL. The 'Views' folder under 'CHANNEL' is expanded, showing 'CHANNEL_VIEW - [Dimension ET]' and 'VIEWNAME - [Hierarchy: STANDARD]'. On the right, the 'Specify View Information' dialog box is open, showing the following fields:

Dimension Name: CHANNEL
Hierarchy Name: STANDARD
View Name:

Column Name	Data Type	Object Type
DIM_KEY	VARCHAR2	Key
LEVEL_NAME	VARCHAR2	Level Name
PARENT	VARCHAR2	Parent
TOTAL_CHANNEL	VARCHAR2	Hierarchy Level
CHANNEL	VARCHAR2	Hierarchy Level

Standard Disclaimer – Beta software! No promises!

Copyright © 2007, Vlamis Software Solutions, Inc.



OLAP 11g Changes

- Optimizer pushes joins down to AW
- Enables efficient non-OLAP-aware SQL queries

The screenshot displays the Oracle SQL Developer interface. The top pane shows an SQL query:

```
FROM time_view t,  
product_view p,  
customer_view cu,  
channel_view ch,  
units_cube_view f  
WHERE t.dim_key = f.TIME  
AND p.dim_key = f.product  
AND cu.dim_key = f.customer  
AND ch.dim_key = f.channel  
AND t.long_description = '2000'  
AND p.long_description = 'Total Product'  
AND cu.long_description = 'All Customers'
```

The bottom pane shows the execution plan for the query. The 'Explain' tab is selected. The plan is as follows:

Operation	Optimizer	Cost	Cardinality	Bytes	Part
SELECT STATEMENT	ALL_ROWS	1028	1	520	
HASH JOIN		1028	1	520	
MERGE JOIN(CARTESIAN)		407	1	380	
MERGE JOIN(CARTESIAN)		305	1	240	
MERGE JOIN(CARTESIAN)		203	1	160	
CUBE SCAN(OUTER) GLOBAL.CHANNEL					
BUFFER(SORT)		102	1	80	
CUBE SCAN(OUTER) GLOBAL.PRODUCT					
BUFFER(SORT)		102	1	80	

Standard Disclaimer – Beta software! No promises!

Copyright © 2007, Vlamiis Software Solutions, Inc.



OLAP 11g Changes

- Views are stored in Oracle Dictionary
- Notice in SYS.USER_DIMENSION_VIEWS

The screenshot shows the Oracle SQL Developer interface. On the left, the 'Connections' pane displays a tree structure with 'stack07 - global - main3' selected. Under 'Views', 'CUSTOMER_SHIPMENTS_VIEW' is highlighted. The main window shows the 'Enter SQL Statement' area with the query: `select * from sys.user_dimension_views;`. Below the query, the 'Results' tab is active, displaying a table with 4 rows and 4 columns: DIMENSION_NAME, VIEW_OWNER, VIEW_NAME, and VIEW_TYPE.

	DIMENSION_NAME	VIEW_OWNER	VIEW_NAME	VIEW_TYPE
1	TIME	GLOBAL	TIME_VIEW	ET
2	CHANNEL	GLOBAL	CHANNEL_VIEW	ET
3	PRODUCT	GLOBAL	PRODUCT_VIEW	ET
4	CUSTOMER	GLOBAL	CUSTOMER_VIEW	ET

Standard Disclaimer – Beta software! No promises!

Copyright © 2007, Vlamis Software Solutions, Inc.



OLAP 11g Changes

- **Cost-based presummarization balances aggregation time with performance**

Create Cube

General Translations Implementation Details Materialized Views Rules Summarize To Cache

Presummarization

Select the type of presummarization you wish to use

☐ No presummarization

☒ Cost-based presummarization

Percentage: 0 25 50 75 100 21

☐ Level-Based Presummarization

Choose the regions of the cube to be presummarized and stored in the analytic workspace.

Dimension:

Dimension	Levels
TIME	<input type="checkbox"/> ALL_TIMES
CUSTOMER	<input type="checkbox"/> CALENDAR_YEAR
PRODUCT	<input type="checkbox"/> MONTH
CHANNEL	<input type="checkbox"/> QUARTER

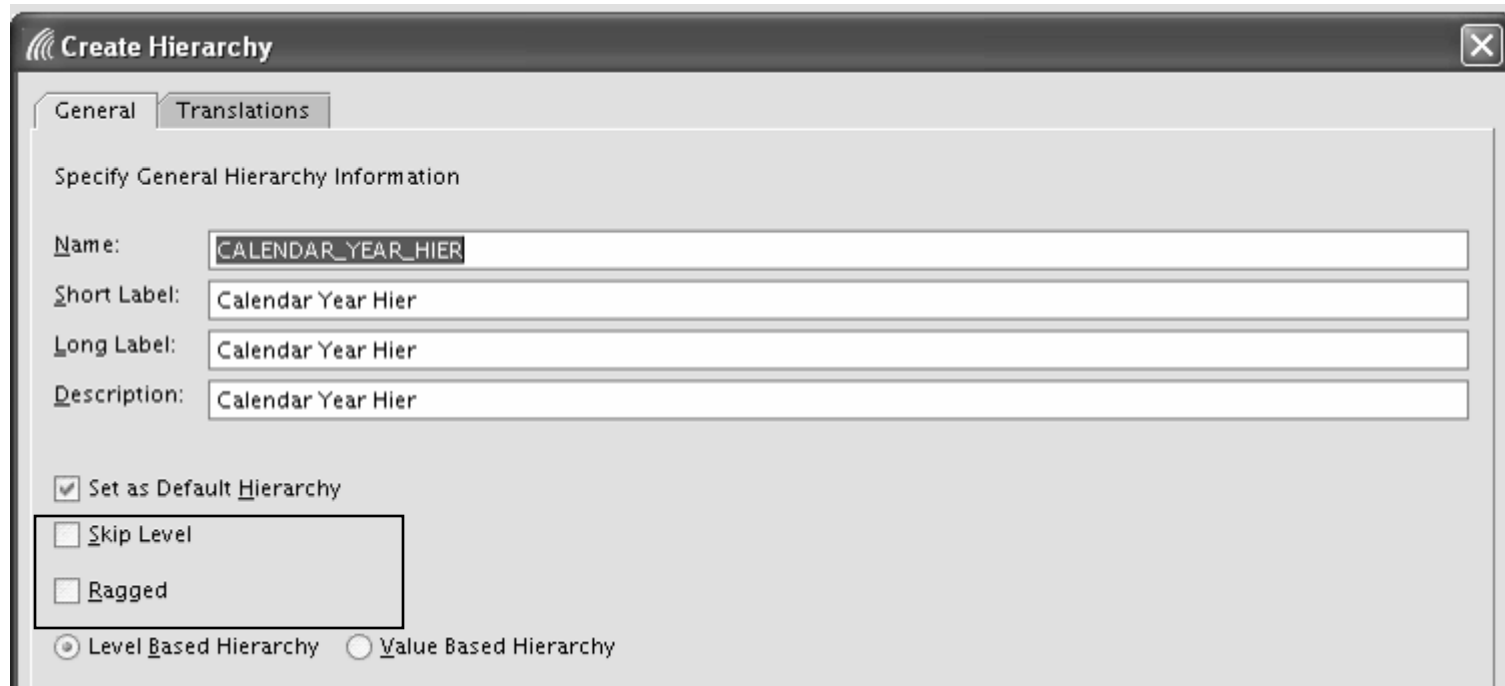
Standard Disclaimer – Beta software! No promises!

Copyright © 2007, Vlamis Software Solutions, Inc.



OLAP 11g Changes

- Native support for AWs with skip level and ragged hierarchies



The image shows a screenshot of the 'Create Hierarchy' dialog box in Oracle OLAP 11g. The dialog has two tabs: 'General' and 'Translations'. The 'General' tab is selected. The title bar says 'Create Hierarchy' with a close button. The main area is titled 'Specify General Hierarchy Information'. It contains several text input fields: 'Name' (with value 'CALENDAR_YEAR_HIER'), 'Short Label' (with value 'Calendar Year Hier'), 'Long Label' (with value 'Calendar Year Hier'), and 'Description' (with value 'Calendar Year Hier'). Below these fields are three checkboxes: 'Set as Default Hierarchy' (checked), 'Skip Level' (unchecked), and 'Ragged' (unchecked). At the bottom, there are two radio buttons: 'Level Based Hierarchy' (selected) and 'Value Based Hierarchy' (unselected).

Create Hierarchy

General Translations

Specify General Hierarchy Information

Name: CALENDAR_YEAR_HIER

Short Label: Calendar Year Hier

Long Label: Calendar Year Hier

Description: Calendar Year Hier

☒ Set as Default Hierarchy

☐ Skip Level

☐ Ragged

☒ Level Based Hierarchy ☐ Value Based Hierarchy

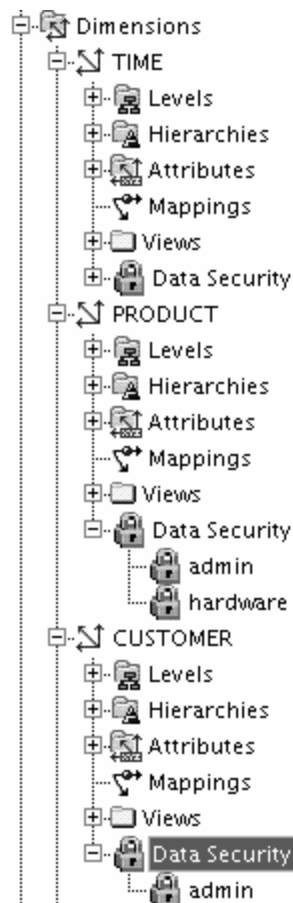
Standard Disclaimer – Beta software! No promises!

Copyright © 2007, Vlamis Software Solutions, Inc.



OLAP 11g Changes

- Create security policies based on hierarchies



Condition Expression:

GLOBAL.PRODUCT.DIM_KEY IN ('2') OR '2 GLOBAL.PRODUCT.PRIMARY LEVEL GLOB

User or Role	Type	Select	Insert
SCOTT	User	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Standard Disclaimer – Beta software! No promises!

Copyright © 2007, Vlamis Software Solutions, Inc.



OLAP 11g Changes

- Calc Wizard replaced by powerful "complete the sentence" wizard
- Expression language more SQL-like
- EQs of Calculated Measures in 11g-format AWs "read-only"

Choose a calculation type:

Rank

Calculation:

Rank members of the PRODUCT dimension and PRIMARY hierarchy based on measure UNITS_CUBE.UNITS (...)

. Calculate rank using RANK method with member's level in order lowest to highest.

member's level
member's parent
member's ancestor

Expression:

RANK() OVER HIERARCHY (GLOBAL.PRODUCT.PRIMARY ORDER BY GLOBAL.UNITS_CUBE.UNITS WITHIN LEVEL)

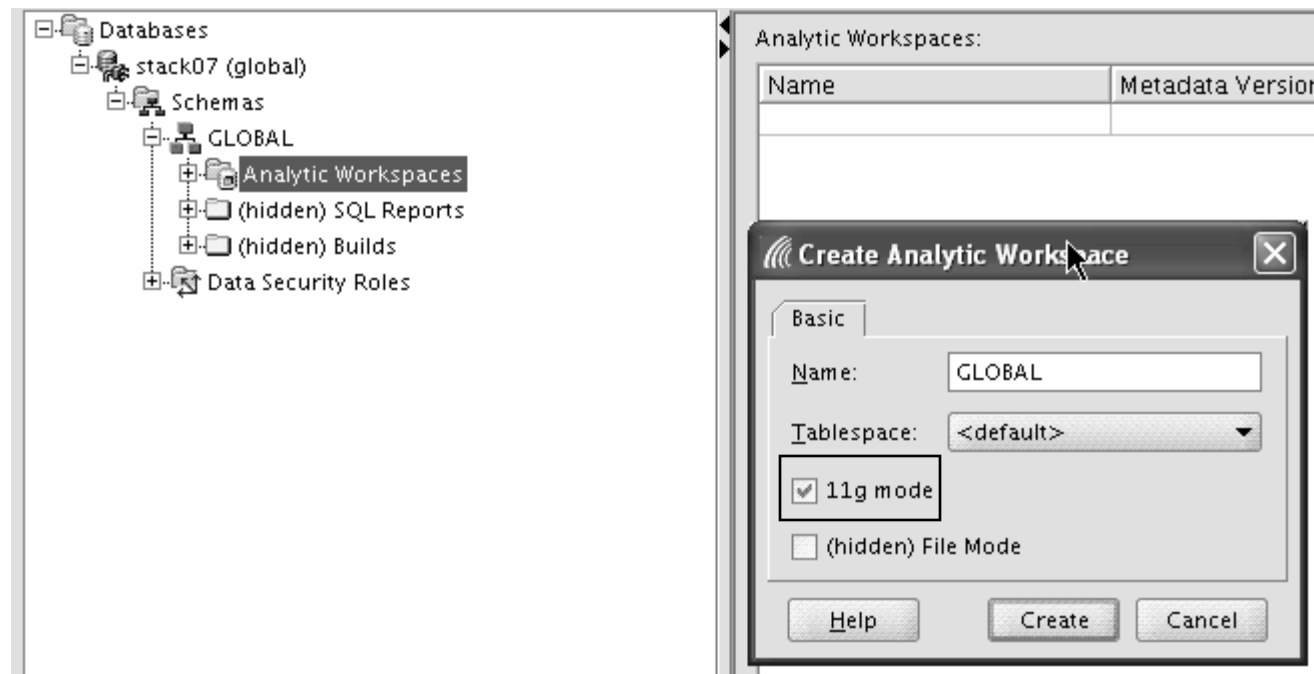
Standard Disclaimer – Beta software! No promises!

Copyright © 2007, Vlamis Software Solutions, Inc.



OLAP 11g Changes

- Can Create AWs in 11g mode (automatic views)
- If no 11g mode, have same flexibility as 10g



Standard Disclaimer – Beta software! No promises!

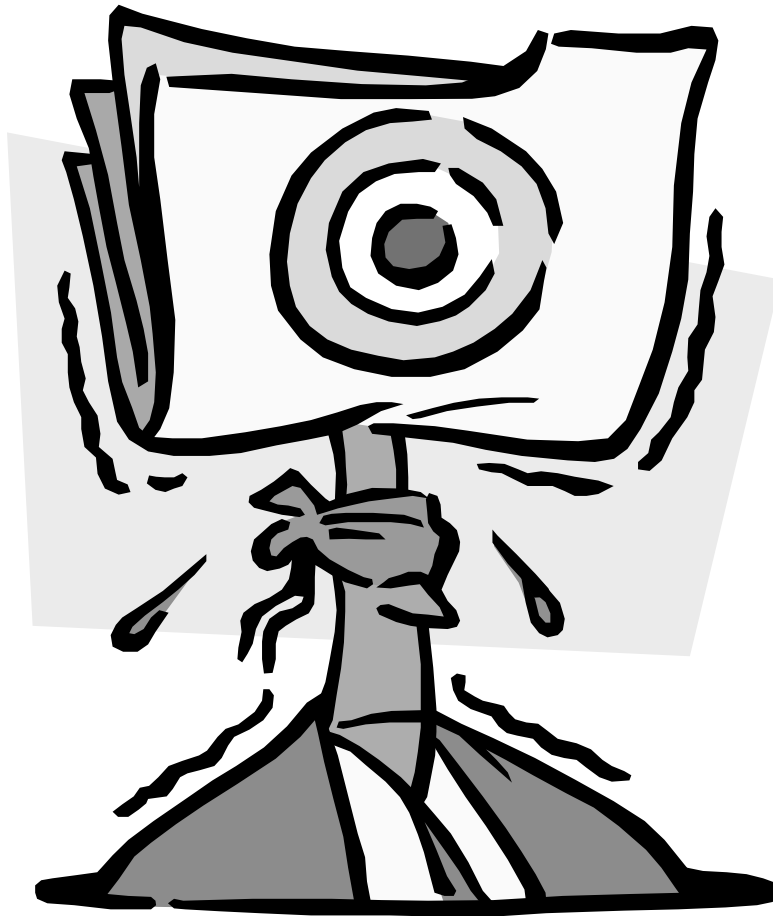
Copyright © 2007, Vlamis Software Solutions, Inc.



Further Information

- Oracle BI Sales
 - ❑ <http://www.oracle.com/bi>
- Oracle BI Technical
 - ❑ <http://www.oracle.com/technology/tech/bi/index.html>
- Oracle BI EE on top of Oracle OLAP
 - ❑ Collaborate 208: Using Oracle BI EE with Oracle OLAP Cubes on www.vlamis.com/papers.html
 - ❑ http://www.oracle.com/technology/obe/obe_bi/bi_ee_1013/olap/index.html
- VMWare image with Demo environment
 - ❑ www.bic2q.com
- Oracle OLAP and AWM Sales
 - ❑ http://www.oracle.com/solutions/business_intelligence/olap.html
- Oracle OLAP Technical
 - ❑ <http://www.oracle.com/technology/products/bi/olap/index.html>
- This Demo
 - ❑ <http://www.vlamis.com/demo>

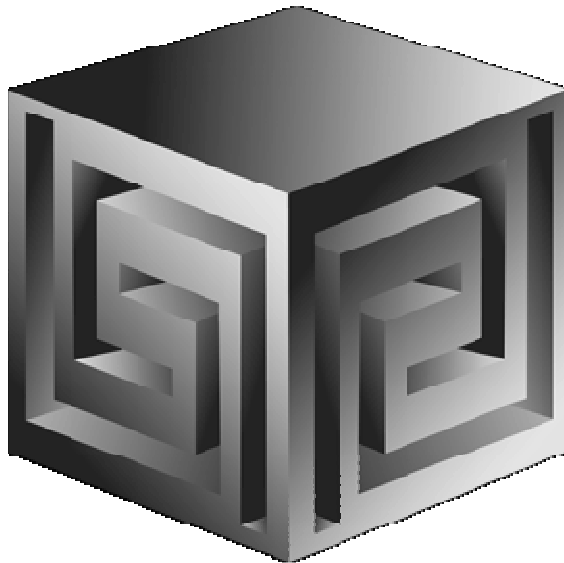
QUESTIONS?





Implementing Oracle BI EE Using Oracle OLAP Cubes

**Dallas Oracle User Group
Oct 12, 2007**



**Cathye Pendley & Mark Thompson
Cpendley or mthompson @vlamis.com**

Vlamis Software Solutions, Inc.

816-781-2880

<http://www.vlamis.com>

Copyright © 2007, Vlamis Software Solutions, Inc.

Copyright © 2007, Vlamis Software Solutions, Inc.