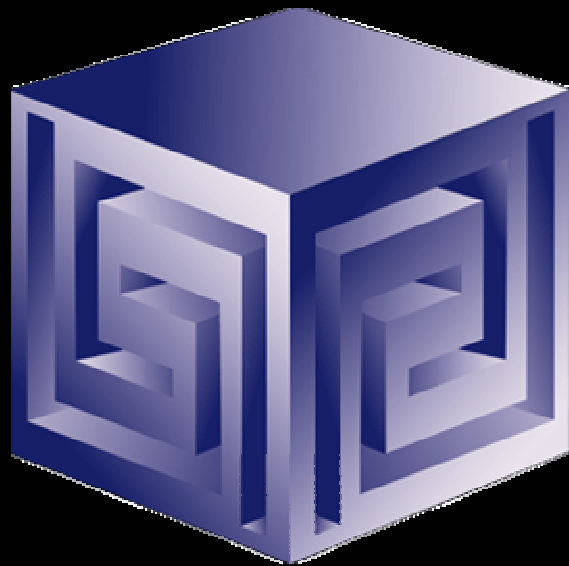


# **Effectively Using Oracle OLAP in Business Intelligence Applications**

**presented at**

**ODTUG 2004**



**Presented by:**

**Dan Vlamis (dvlamis@vlamis.com)**

**Vlamis Software Solutions, Inc.**

**(816) 781-2880**

**<http://www.vlamis.com>**



# Vlamiis Software Solutions, Inc.

- Founded in 1992 in Kansas City, Missouri
- Provides business solutions to international and domestic clients based on Oracle technologies.
- Authorized software reseller
- Creator of the first Oracle 9i Business Intelligence and Analytics tool.
- Core competency include:
  - Certified designers,
  - Developers,
  - Implementers
  - Nationally recognized technical authors, speakers and publishers.



# Agenda

- Historical background
- Comparison to Express
- Why Oracle9i OLAP?
- What is Oracle9i OLAP and how does BI Beans fit in?
- Oracle9i OLAP storage options
- Structure of Analytic Workspace
- Differences from Express
- Case studies



# Requirements of OLAP Systems

- Multidimensional user view
- Drill down, rotate
- User-created measures
- Iterative discovery process
- Multiple levels (embedded totals)

**Do these attributes imply a proprietary MDBMS?**

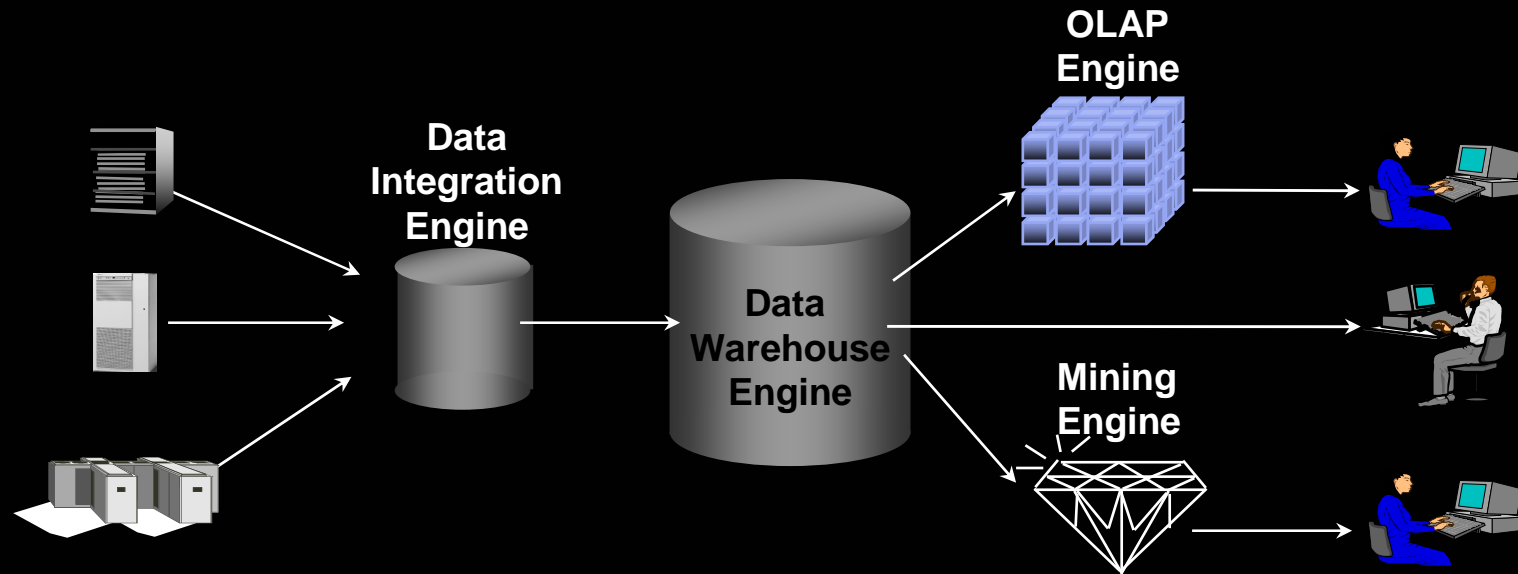
**No.**



## In the Past

- Previous development of BI and OLAP Applications required proprietary development environment
- Each deployment model required a different tool
- Development effort very labor intensive
- Concept to Deployment takes long time
- Requires specialized skills

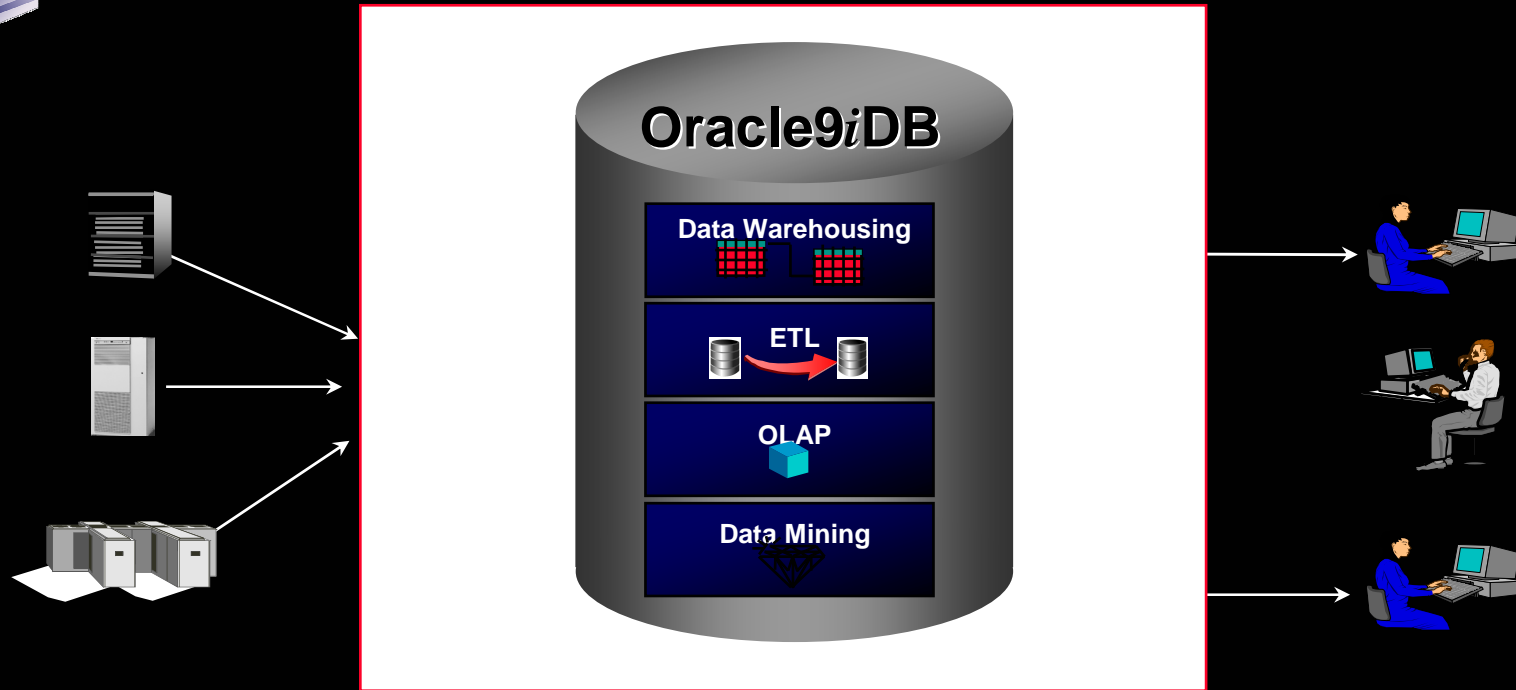
# Business Intelligence the Old Way



- Special purpose engines for differing tasks
- Metadata migration tools ease replication
- User interfaces generally different for different tools



# BI the New Way: Oracle9i DB



- Single business intelligence platform
  - Reduce administration, implementation costs
  - Faster deployment & Improved scalability and reliability



# Key Developments

- Integrating Express Server team into Oracle Server team
- RDBMS gets OLAP functionality in SQL
- Data Warehouse features in Oracle 8i:
  - Bitmap and bitmap join indexes
  - Materialized views
  - Query rewrite
  - “N-pass” functions
- Increasing use of very sparse data
- Oracle focus on integration



# Express Features and Limitations



- Multidimensional data store provides quick access
- All data in single proprietary file
- Express SPL powerful for calcing and extending apps
- GUI environment uses Visual Basic language
- RAM moves data from RDBMS into Express
- Designed for multiple read/only users
- Difficult for IT to adopt, Max database size
- Difficult to integrate and schedule back ups
- Long learning curve for new developers
- GUI not consistent with Oracle direction
- Star and snowflake only, performance concerns
- No multi-writer support

# **Express Has Not Gone Away... It has just been absorbed**



- Analytic Workspaces are Express DB files
- Oracle marketing will downplay Express
- Design considerations for Analytic Workspaces same as for Express DBs
- Express engine still there
- Express SPL still there (but only operates on storage in analytic workspaces)
- Analytic Workspaces better at complex calcs (for now)
- Use OLAP Worksheet for OLAP DML commands



# Terminology Map

## Express

## Oracle OLAP

Express database	Analytic Workspace
Oracle Express Server	Part of Oracle OLAP
Express Instance Manager	Part of Oracle OLAP
Express SPL (or 4GL)	OLAP DML
RAA	OLAP folder of OEM
RAM	Automatic*

\* Not necessary to use RAM any more since relational data directly accessible



# Terminology Map (continued)

## Express

## Oracle OLAP

oesdba	olapdba
SNAPI	Java OLAP API
Oracle Express Objects	JDeveloper
Express Basic	Java
Express Administrator	AW Manager
OESCMD and Administrator command line	OLAP Worksheet



# Components of Oracle OLAP

- OLAP folder of OEM defines multidimensional structure (replaces RAA)
- Query Processor processes Java OLAP API
- OLAP Worksheet provides command line to OLAP DML
- BI Beans provide linkage to Java OLAP API
- JDeveloper is environment for building apps (replaces OEO)

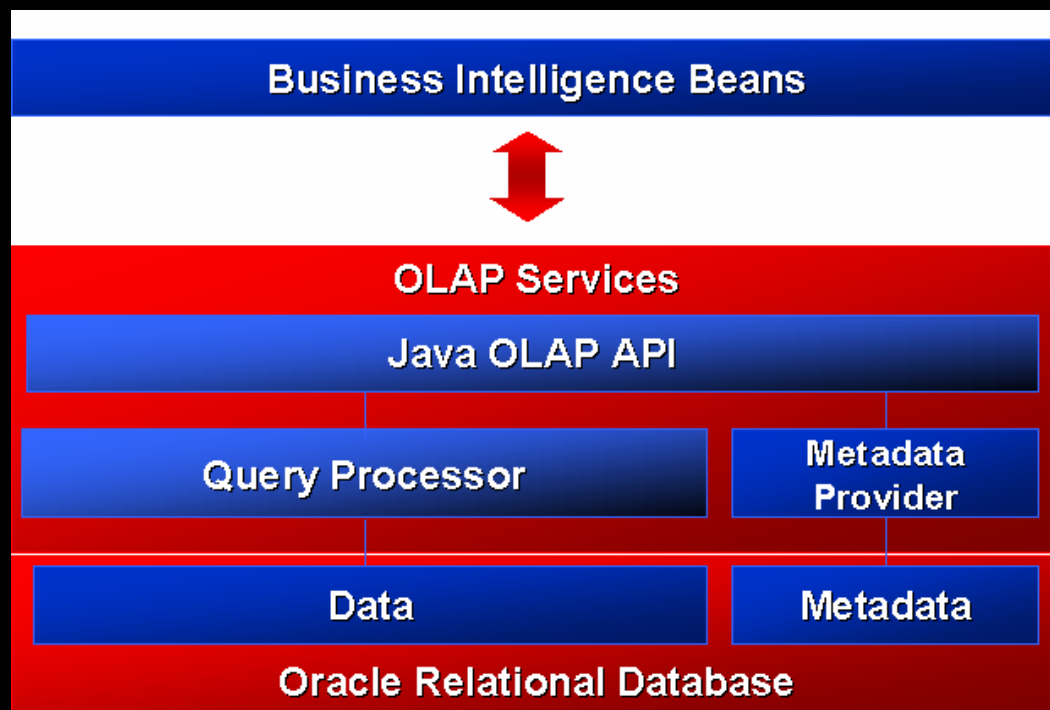


# What Does Oracle OLAP Add?

- 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
- What-if, forecasting
- Platform for extensions
- Analytics (e.g. forecasting) built into server



# Oracle OLAP Architecture



**Rapid application development**  
**Analysis ready**

**Java OLAP API**  
**Predictive analysis functions**

**Scaleable data store**  
**Integrated meta data**  
**Summary management**  
**SQL analytic functions**



# Java OLAP API

- Object-oriented
- Mathematically consistent
- Java (industry standard)
- Compatible with JOLAP standard
- Declaratively (not procedure) based
- Data in relational or analytic workspace
- Multidimensional cursors
- Really designed for low-level access
- Most developers will use BI Beans instead



# OLAP API Calculation Capabilities



- Multidimensional object model
- Totals broken out by multiple attributes
- Row and column calculations
- Union dimensions
- Measures as dimensions
- Calculated dimension members  
(e.g. income 0-20K, 20-50K, 50-75K, >75K)
- Asymmetric queries
- Multiple measures per cell (e.g. color-coding)



# Simple Java OLAP API Example

## English

**Select the products where the dollars measure is greater than 1,000,000 for geography Phoenix for time period Jun2004.**

## Express

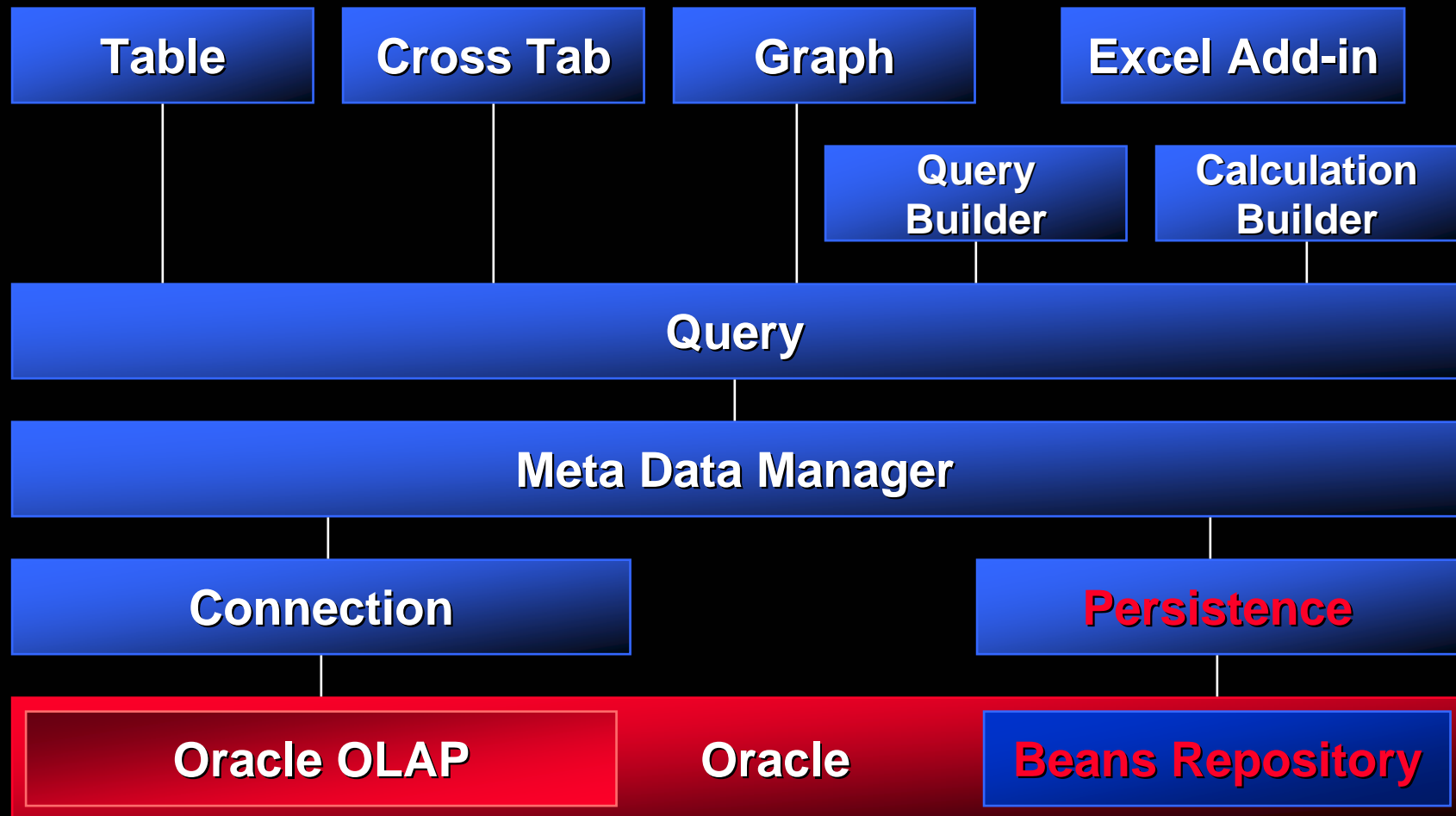
```
limit geography to 'PHOENIX'  
limit time to 'JUN2004'  
limit product to dollars gt 1000000
```

## Java OLAP API

```
Source geogSel = geography.selectValue("PHOENIX");  
Source timeSel = time.selectValue("JUN2004");  
Source dolByProd = dolSrc.join(geogSel).join(timeSel);  
Source prodSel = product.select(dolByProd.gt(1000000));  
Source dolGT1Mill =  
    dolSrc.join(geogSel).join(timeSel).join(prodSel);
```

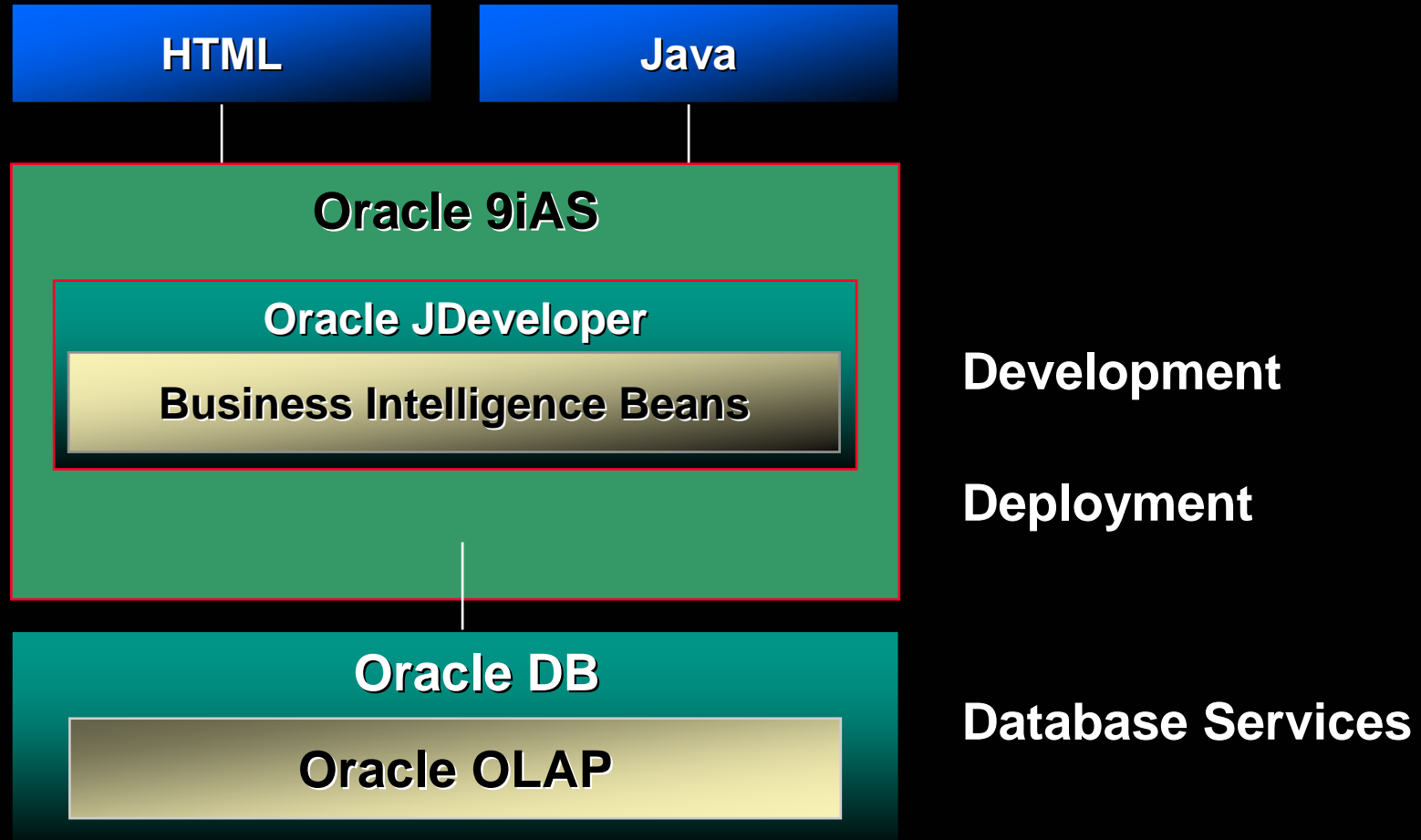


# Business Intelligence Beans





# Java Development Environment





# What Development Tool?

- Java OLAP API designed for products
- BI Beans for custom applications (using JDev)
- Oracle Reports for highly formatted reports
- Discoverer for ad hoc analysis
- Oracle Apps for analysis of Apps data
- 3rd Party tools fill in gaps



# Executing OLAP DML From SQL

- DBMS\_AW.EXECUTE procedure
- DBMS\_AW.INTERP function (returns a CLOB with the log from the commands)
- DBMS\_AW.GETLOG() function gets the log from the most-recently executing OLAP DML command or program or OLAP\_TABLE function



# OLAP DML Changes

- All SPL functionality there:
  - Some operating and file system commands changed
  - XCA and SNAPI no longer applicable
  - External Call (EXTCALL) command
  - SQL CONNECT command and SQL.DBMS option not necessary
- Added support for:
  - Parallel aggregate
  - Allocate
  - Data conversion functions, new data types
  - Concat (union) dimensions
  - objects with same name attached multiple times
  - dynamic model execution
  - bulk data transfer with RDBMS

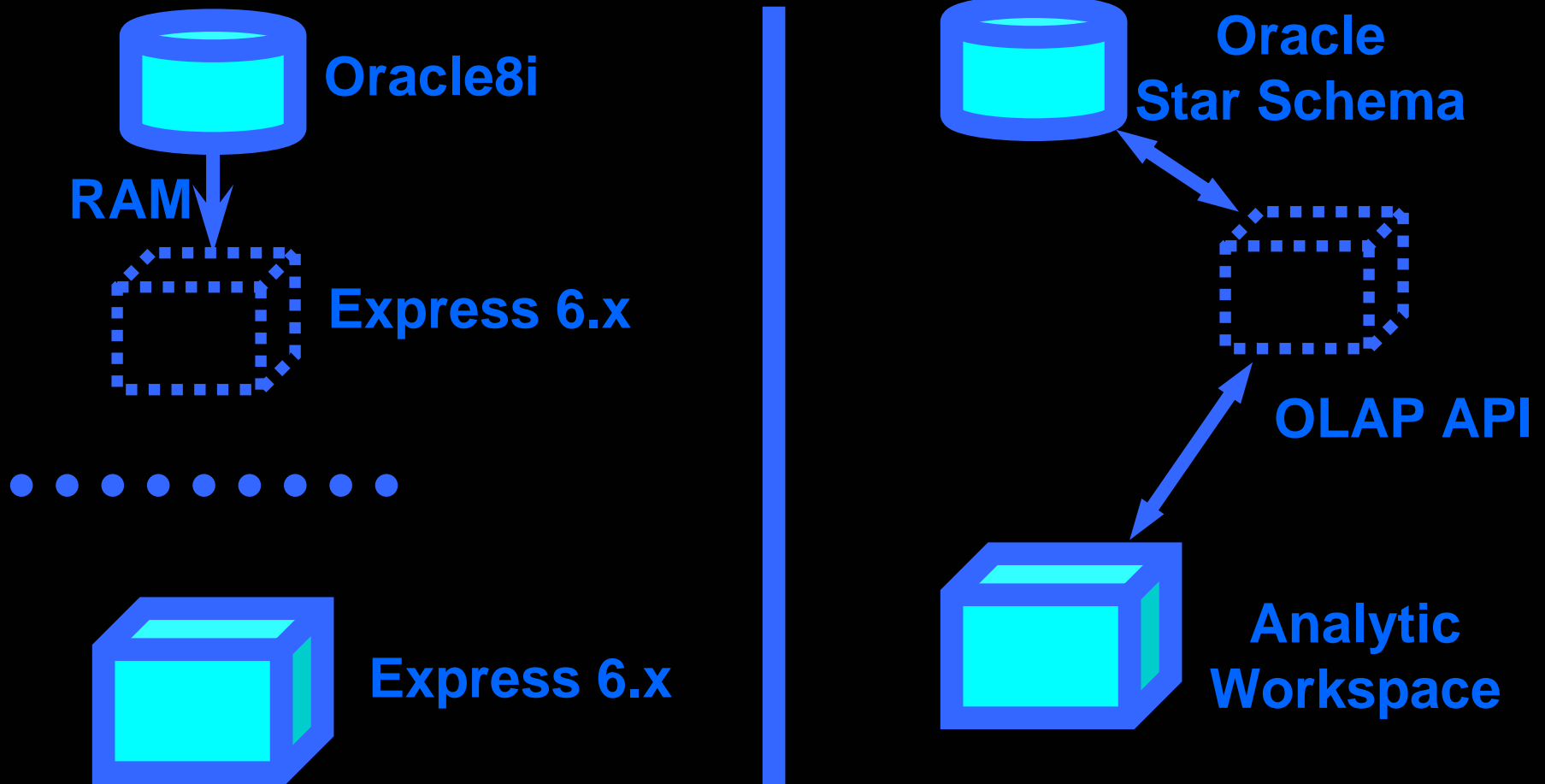


## **OLAP DML Changes (continued)**

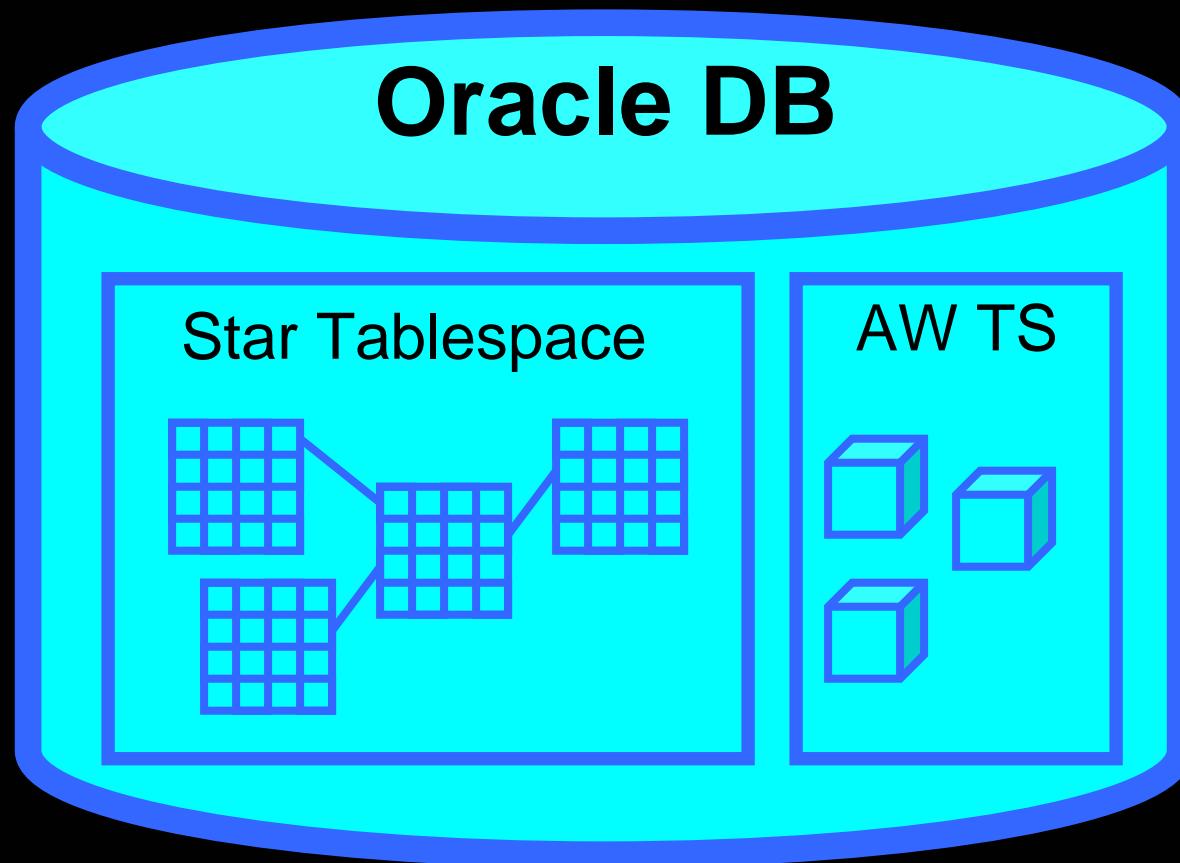
- Database is now Analytic Workspace
  - Stored in LOB in RDBMS
  - AW create command creates AW's
  - Can attach multiple AW's
  - Non-unique object names OK now
- Update AND THEN COMMIT
- SQL PREPARE for high-speed AW->RDBMS
- SQL access to AW via OLAP Table Function



# New Data Storage Possibilities in Oracle OLAP



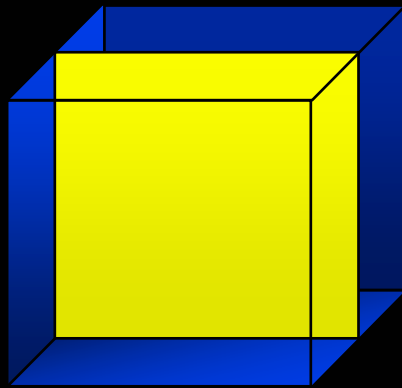
# Analytic Workspaces Are Stored in Tablespaces in Oracle OLAP



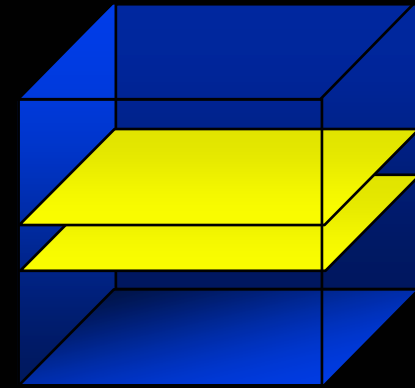
# Oracle OLAP AW Stores Data in Cubes



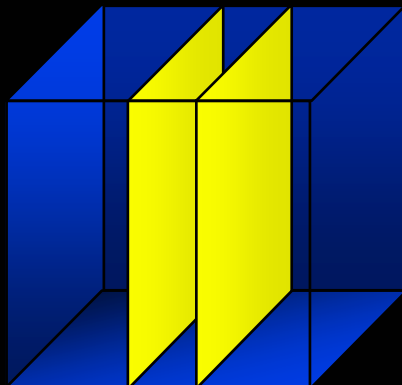
*Fast Flexible Access to Summarized Data*



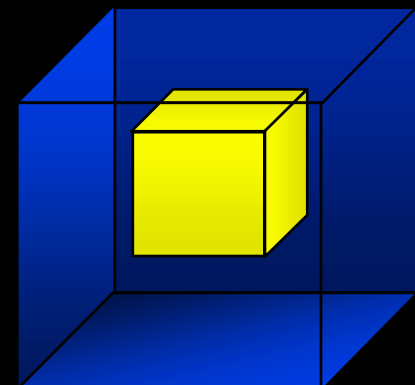
**Product Mgr. View**



**Regional Mgr. View**



**Financial Mgr. View**

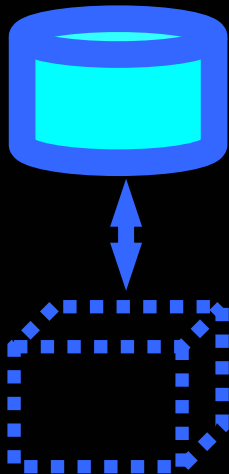


**Ad Hoc View**



# Advantages of RDBMS Storage

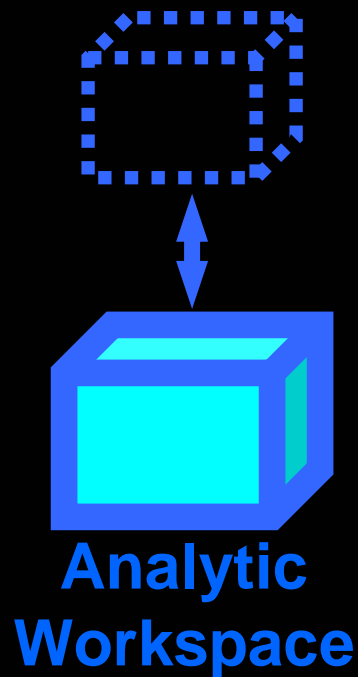
Oracle  
Star Schema



- Store data in familiar RDBMS
- Easy access to data using SQL
- Can use materialized views
- Best for read-only applications
- Model with OWB
- Data may already be in schema



## Advantages of AW Storage



- Faster multidimensional access
- Personal user workspaces
- Best for read/write applications
- Best for heavier analysis
- OLAP DML language



## Oracle AW Tips

- Use separate tablespaces for AWs
- Keep out of SYS tablespace
- Stripe across drives with multiple pathnames
- Make sure they autoextend
- Users should use TABLESPACE keyword when creating AWs

# AWs Allow Session Level Commit



- Want to concentrate on territories that underperformed last year (complex calc)
- Compute forecast for those territories
- Let me spread this forecast down organization
- Now adjust this value + 10%
- Re-run report
- Never mind
- Go back to prior scenario
- Submit THAT forecast for approval



## **AWs Allow for What-if**

- Modeling organizational changes
  - territory realignments
  - product hierarchy changes
- Product new launches
  - model new products after established product
- Forecasting
  - multiple scenarios
  - personal overrides of forecast
  - spread down of higher-level overrides
  - locks





## **AWs for Personal Data**

- DBAs often don't want users modifying data
- Modeling data
- User-created calculations
- Multiple scenarios for forecasts
- Allows for collaboration between users



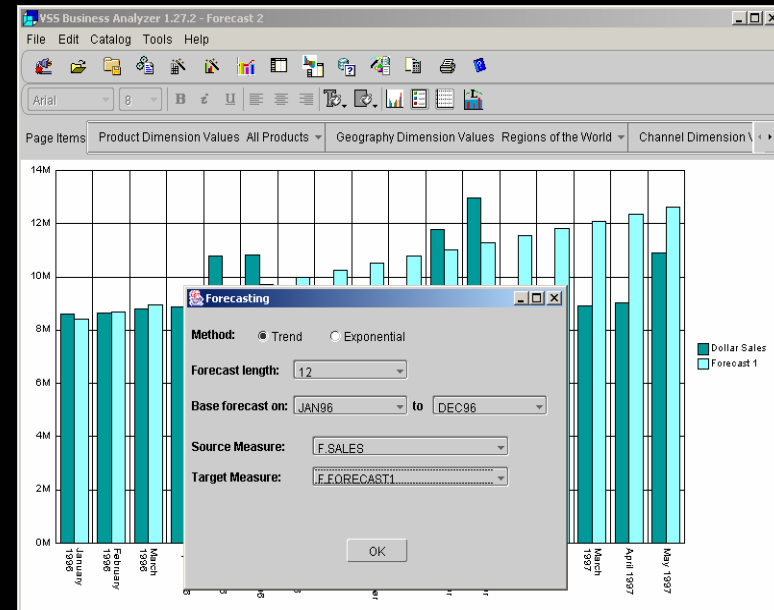
## **AWs Allow for Rules Based Apps**

- OLAP DML for manipulating data in DB
- Aggregation
- Allocation
- Decision Trees
- Transform data via complex OLAP DML logic
- High-power statistical functions built-in
- Rules or logic that differs by organization
- Expert systems



# What Can We Really Do?

- VSS Business Analyzer is an example of what can be done
- Developed in less than 4 months!
- Comprehensive Application that provides Ad-Hoc Query capabilities in both Thick and thin clients
- Utilizes a central report catalog available to all client types
- Provides an extensible framework for more complex applications



**See [www.vlamis.com](http://www.vlamis.com) for more info...**



# Managing a BI Project

- Involve end-users early on
- Prototype, pilot, then phase 1
- Recruit "champion" users
- Lead from user community, not IT
- Develop in phases
- Provide value early on
- Keep it simple (at first)
- Need forum for users to share ideas
- Provide user guide with user's data



# Oracle OLAP Case 1

- Company needs to reduce inventory levels
- Uses OLAP DML Forecast command based on orders
- Users can override forecasts and add their own promotional campaigns
- Computes more accurate forecasts of production needs, reducing inventory levels
- Can compare accuracy of monthly forecasts by comparing various “scenarios” each month with actual shipments
- Application presented as JSP for business forecasters / managers



## Oracle OLAP Case 2

- Customer is service organization
- Company wants to minimize hold time but not increase headcount
- Solution is to analyze hold time and customer resolution time for each support analyst
- Can rank support engineers / departments by customer satisfaction / resolution / callback rates



## Oracle OLAP Case 3

- Company has complex GL and needs “business rules engine” for allocating costs and income
- Uses Oracle OLAP engine to develop models to allocate data based on rules analysts develop
- Users can develop their own way of analyzing the data rather than relying on IT
- IT sets up infrastructure, users develop actual analyses

# How Get Started? / More Information

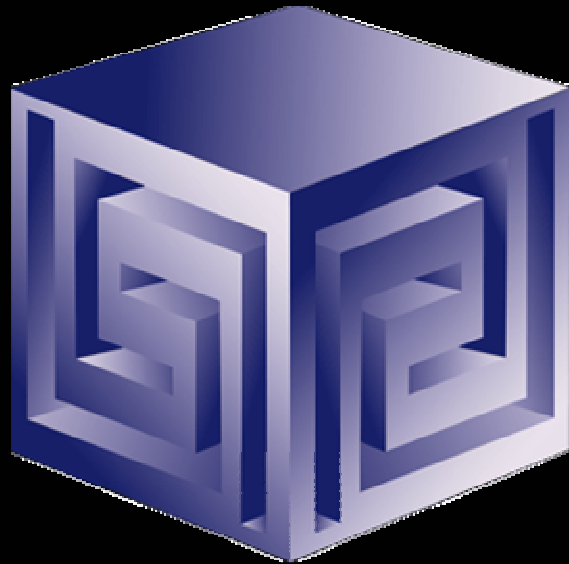


- Java programming
  - Start with JDeveloper 9i (download from OTN)
- BI Beans
  - Tutorials (In JDeveloper)
  - Samples (on OTN)
  - Training (Web and Instructor Led)
- Oracle Technology Network
  - <http://otn.oracle.com/products/bib/content.html>



# **Effectively Using Oracle OLAP in Business Intelligence Applications**

**presented at  
ODTUG 2004**



**Presented by:**

**Dan Vlamis (dvlamis@vlamis.com)**

**Vlomis Software Solutions, Inc.**

**(816) 781-2880**

**<http://www.vlomis.com>**