EFFECTIVELY USING 9I OLAP IN BUSINESS INTELLIGENCE APPLICATIONS

Dan Vlamis, Vlamis Software Solutions, Inc.
dvlamis@vlamis.com

PREFACE

At the time of this writing (October, 2002), Oracle9i release 2 has just been released. This release adds significant additional capabilities over Oracle9i release 1. Limited access to information has prevented a thorough review of Oracle9i OLAP release 2.

INTRODUCTION

With Oracle9i, Oracle finally did it. It took seven years since Oracle purchased Express from Information Resources, Inc. (IRI), but Express is finally integrated into its flagship RDBMS engine. Release 2 of Oracle9i OLAP promises tighter integration, with the 9i OLAP engine being imbedded in the kernel of the RDBMS engine. Data can be stored in relational tables, or in a multidimensional "analytic workspace." In either case, the data can be presented multi-dimensionally.

HISTORY

To understand what has changed, it is best to start back at the beginning of Express. The Express language and environment has gone through many changes over the years. Express has been purchased by various companies searching for an analytic platform for decision support applications.

EXPRESS IS BORN

Express was started by several people from MIT who needed a platform to develop decision support applications. In the 1970's, their company, Management Decision Systems (MDS), developed this into a product on the Primos and IBM Virtual Machine (VM) platforms. This product was eventually called *Mainframe Express* (MFX) or "classic Express." In 1985 the product was rewritten in C and ported to the PC platform running MS-DOS. This product would be known as *pcEXPRESS*. The language went through several changes at this time, but the underlying concepts used remained the same—storing data in multidimensional arrays.

IRI ACQUIRES MDS

Information Resources, Inc. (IRI) had been using Express as a platform to analyze its consumer packaged goods supermarket scanner data. In 1985, IRI purchased MDS and eventually developed *DataServer* (later renamed Oracle Sales Analyzer) as a software product designed to deliver its scanner data and the analysis tools required by its clients. Along the way, IRI ported the new pcEXPRESS database engine to various mainframe environments and added many extensions still found in Express today.

ORACLE PURCHASES EXPRESS

In 1995, Oracle Corporation needed a tool to analyze its financial data and was impressed enough with Express and the product eventually renamed Oracle Financial Analyzer that they bought the Express product line from IRI. Oracle opted to keep the Massachusetts-based development team in a separate division rather than integrate the developers into its California-based organization. Oracle attempted to integrate Express into its flagship RDBMS product offering, especially with its Relational Access Manager offering, but never truly integrated the tools or databases. In 1998, Oracle integrated the development teams into the rest of its organization. Major enhancements necessary to support data warehouses in the

RDBMS made possible delivering OLAP functionality directly out of the RDBMS engine, and the 9i OLAP project was started as part of Oracle9i development.

Oracle9i OLAP Introduced

Oracle9i OLAP changes the way that Oracle-based OLAP data is stored and OLAP applications are built. Data can now be stored natively in the RDBMS. Access to the data is through a new Java OLAP API (application programming interface). BI Beans provides a new interface for accessing the data. Managing OLAP data no longer requires knowledge of an environment foreign to traditional RDBMS developers and DBAs.

Storing Data Relationally Instead of Multi-dimensionally

At its heart, Oracle9i OLAP is about storing data in a series of standard relational tables of rows and columns and accessing the data as if it is stored multi-dimensionally. This, of course, enables standard Oracle tools such as loaders, backup utilities, and data management tools to be used against this data. Materialized views enable higher-level totals to be stored in a data warehouse for efficient access, eliminating many of the traditional performance problems in ROLAP systems. By placing the responsibility for maintaining these materialized views in the server database (instead of application code), applications are greatly simplified and are much less error prone.

RDBMS Data Access Catches up to MDBMS Data Access

Traditionally, one of the key benefits of a MDBMS has been fast data access. Array access is inherently fast because a DBMS can calculate the exact position to locate a cell of data through simple multiplication and addition—no indexes required. See *How Does Express Really Work Anyway* at www.vlamis.com for more information. As larger multidimensional databases are stored with greater sparsity, the need to efficiently compress out NA values with structures such as conjoints and composites has mitigated this advantage. In addition, advanced index methods such as bitmapped indexes have improved RDBMS access for data warehousing purposes. RDBMS systems are finally up to the task of managing OLAP data.

Analytic Workspaces Provide Multidimensional Storage

So, what if you still want to store data multi-dimensionally? Oracle retained the ability to store data natively in arrays with analytic workspaces. Oracle did not advertise this heavily with Oracle9i OLAP release 1. Express databases are still there—they are just called "analytic workspaces." They are even still stored in separate files with a .db extension. This lack of full integration explains why Oracle didn't advertise this fact when Oracle9i OLAP first came out. In release 2, Oracle has changed the data storage mechanism to remove this disadvantage and will be storing the analytic workspaces in special structures in the RDBMS. This, along with access to data in a analytic workspace via SQL, will allow Oracle to take full advantage of multidimensional data

BI Beans Enable Rapid Application Development

While the Java OLAP API enables Java programmers to access 9i OLAP data, most developers will want to use Oracle's enterprise Java Beans for Business Intelligence (BI Beans) to develop applications. These enterprise Java Beans provide the visual interface to interact with Oracle9i OLAP data. Although BI Beans works in many standard Java Beans integrated development environments, Oracle delivers wizards with JDeveloper that simplify the task of developing OLAP applications.

Working with 9i OLAP—Server Side

Oracle9i OLAP is part of Oracle9i. It is so integrated with the rest of the RDBMS that it is not possible to install just Oracle9i OLAP without the RDBMS.

Managing 9i OLAP

The Oracle9i OLAP management tools are integrated into the rest of Oracle Enterprise Manager. What used to be the Express service is now part of the Oracle kernel.

JAVA OLAP API EXAMPLES

The Java OLAP API is the application programming interface that provides multidimensional access to data warehouse data. The API includes primitives that handle typical OLAP needs, such as calculated measures and selection scripts. The following code fragment demonstrates selecting the products where the dollars measure is greater than 1,000,000 for geography Orlando for time period May2001:

```
Source geogSel = geography.selectValue("ORLANDO");
Source timeSel = time.selectValue("MAY2001");
Source result = prodSel;
NumberSource DollSel = (NumberSource) dollars.join(geogSel).join(timeSel).join(prodSel);
Source prodSel = product.select(DollSel.gt(1000000));
```

When data is stored in relational tables, Oracle9i OLAP translates this OLAP API language into SQL, using new SQL primitives such as concatenated rollup, scrollable cursors, and query rewrite for efficient access.

When data is stored in analytic workspaces, Oracle9i OLAP translates this OLAP API language into OLAP DML (a new name for Express SPL).

Since the OLAP API translates into either SQL or OLAP DML, the same application can access RDBMS or analytic workspace data with a few simple changes.

WORKING WITH ORACLE9I OLAP—CLIENT SIDE

While the Java OLAP API is available, few developers will actually use it. Oracle primarily built the Java OLAP API for its own internal use to provide a foundation for future development. Rather than using the Java OLAP API, applications will be developed using Java using Oracle's BI Beans, delivered with JDeveloper 9i. As of the writing of this paper, BI Beans is still in beta, so descriptions are necessarily vague.

BI BEANS ENABLE ACCESS TO OLAP DATA FOR THE MASSES

BI Beans are enterprise JavaBeans that provide access to Oracle9i OLAP calls through a standard interface. Some Beans are "back-end" beans that supply services such as database connections, while others are user interface beans such as Table, Crosstab, and Graph. These form the heart of an OLAP user interface.

CONCLUSION

So what does the future hold? Clearly, Oracle9i OLAP represents a major leap forward for integration between Oracle's RDBMS and OLAP offerings. This integration makes it possible for all Oracle databases to benefit from OLAP technology. Many of the concepts applicable to Express design carry over to Oracle9i OLAP, but the changes in storage and new user interface possibilities will cause most users of existing Express-based applications to reevaluate what can and can't be done. Oracle9i OLAP release 2 extends the technology to effectively re-introduce multidimensional storage in analytic workspaces. This enables better storage mechanisms for "personal data" such as forecasts, what-if analyses, and scenario management. Adoption of Oracle9i OLAP has been slow because without BI Beans, there was no "interface" to take view Oracle9i OLAP data. With BI Beans release, the only item missing is the analytic applications based on BI Beans. You can wait for Oracle to develop these applications, develop them yourself, or purchase one from a third party. Why wait? The power of OLAP in the relational database is here today.

Effectively Using 9i OLAP in Business Intelligence Applications

presented at

OracleWorld 2002

paper number 32066

Presented by:

Dan Vlamis (dvlamis@vlamis.com)

Vlamis Software Solutions, Inc.
(816) 781-2880

http://www.vlamis.com

Vlamis Software Solutions, Inc.

- A Member of Oracle Partner Program since 1995
- Member of many Oracle Beta Programs, including Express, 9i OLAP, BI Beans
- Designs and implements databases/data marts/data warehouses using RDBMS and Multidimensional tools
- Specializes in Data Transformation, Data Warehousing, Business Intelligence, Applications Development
- Founder Dan Vlamis is former developer of Sales Analyzer Application
- Creator of first ad hoc query tool to access 9i OLAP
- Oracle Certified Solutions Provider

ORACLE® CERTIFIED SOLUTION PARTNER

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

Requirements of OLAP Systems

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

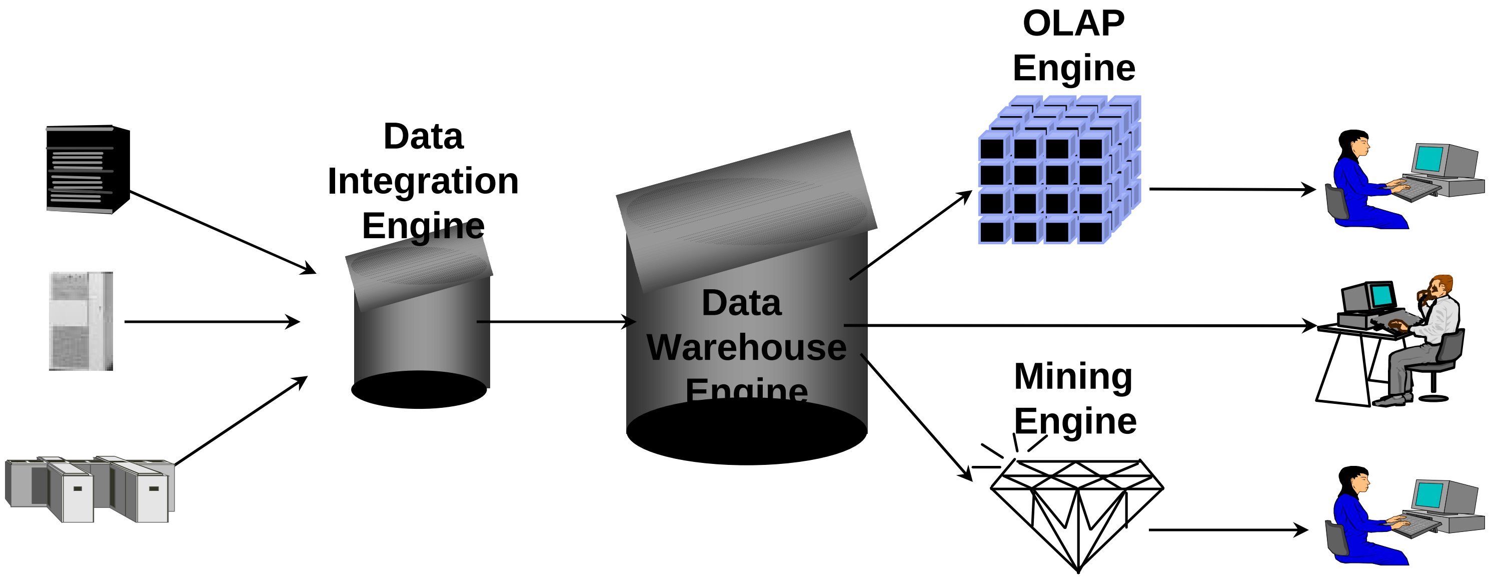
Do these attributes imply a proprietary MDBMS?

No.

In the Past

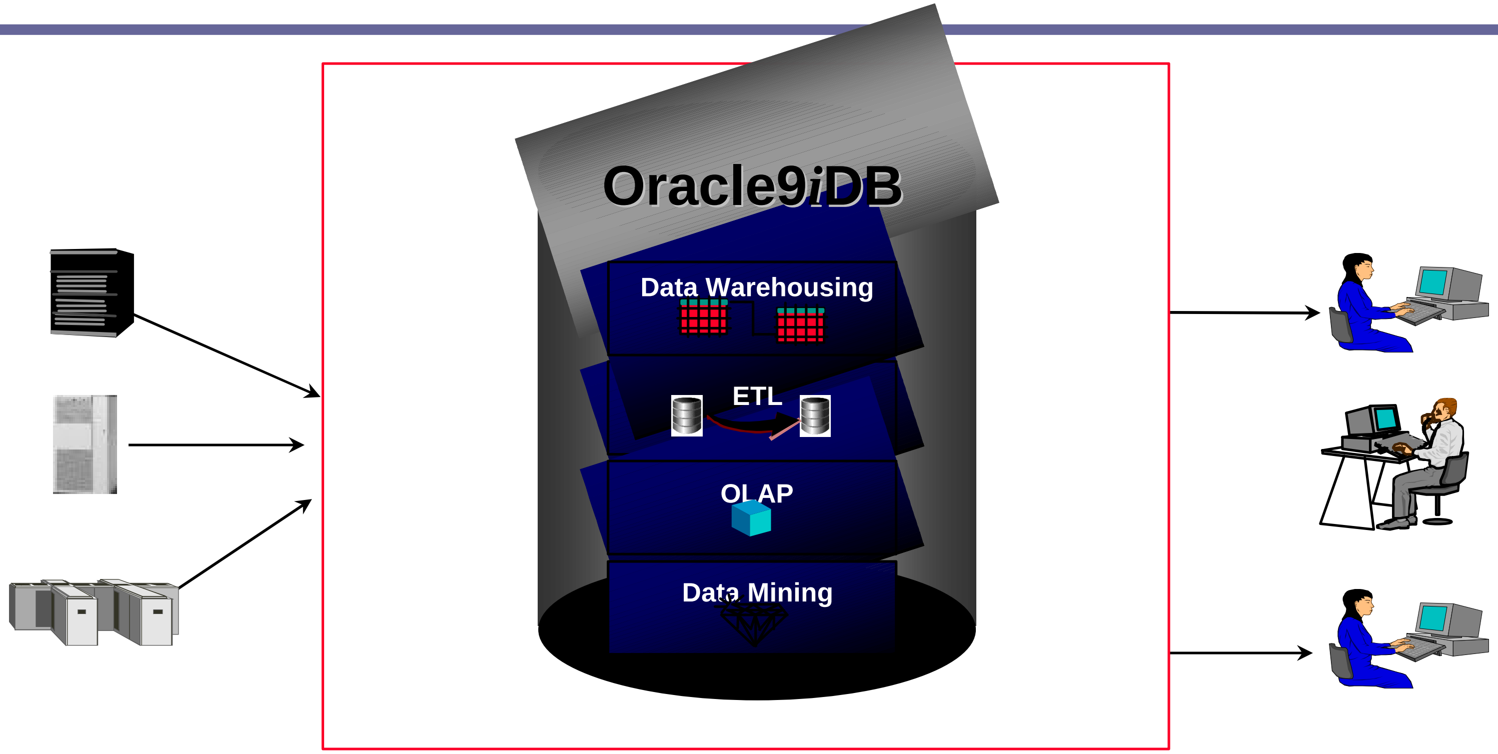
- Previous development of BI and OLAP Applications required proprietary development environment
- Each deployment model required a different tool
- Development effort very labor intensive
- Concept to Deployment takes long time
- Requires specialized skills

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 calculations (for now)
- Use OLAP Worksheet for OLAP DML commands

Terminology Map

Express	Oracle9i OLAP
Express database	**Analytic Workspace**
Oracle Express Server	**Part of Oracle9i OLAP**
Express Instance Manager	**Part of Oracle9i OLAP**
Express Agent	**OLAP Agent**
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

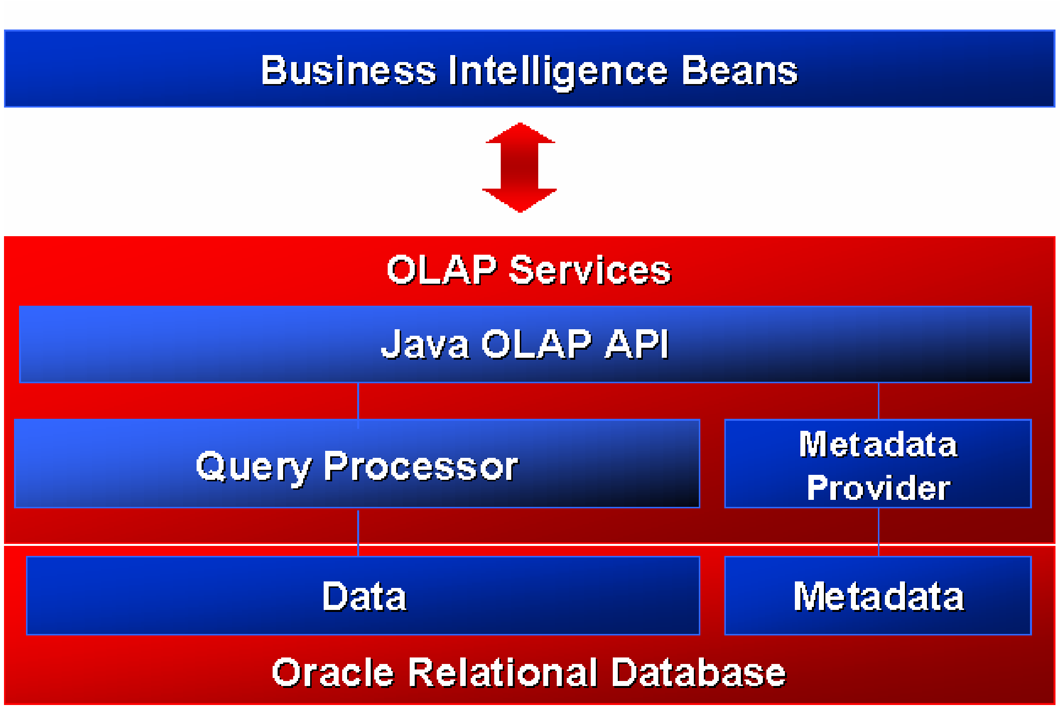
Terminology Map (continued)

Express	Oracle9i 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 9i 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)**

Oracle 9i 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 Orlando for time period May2001.

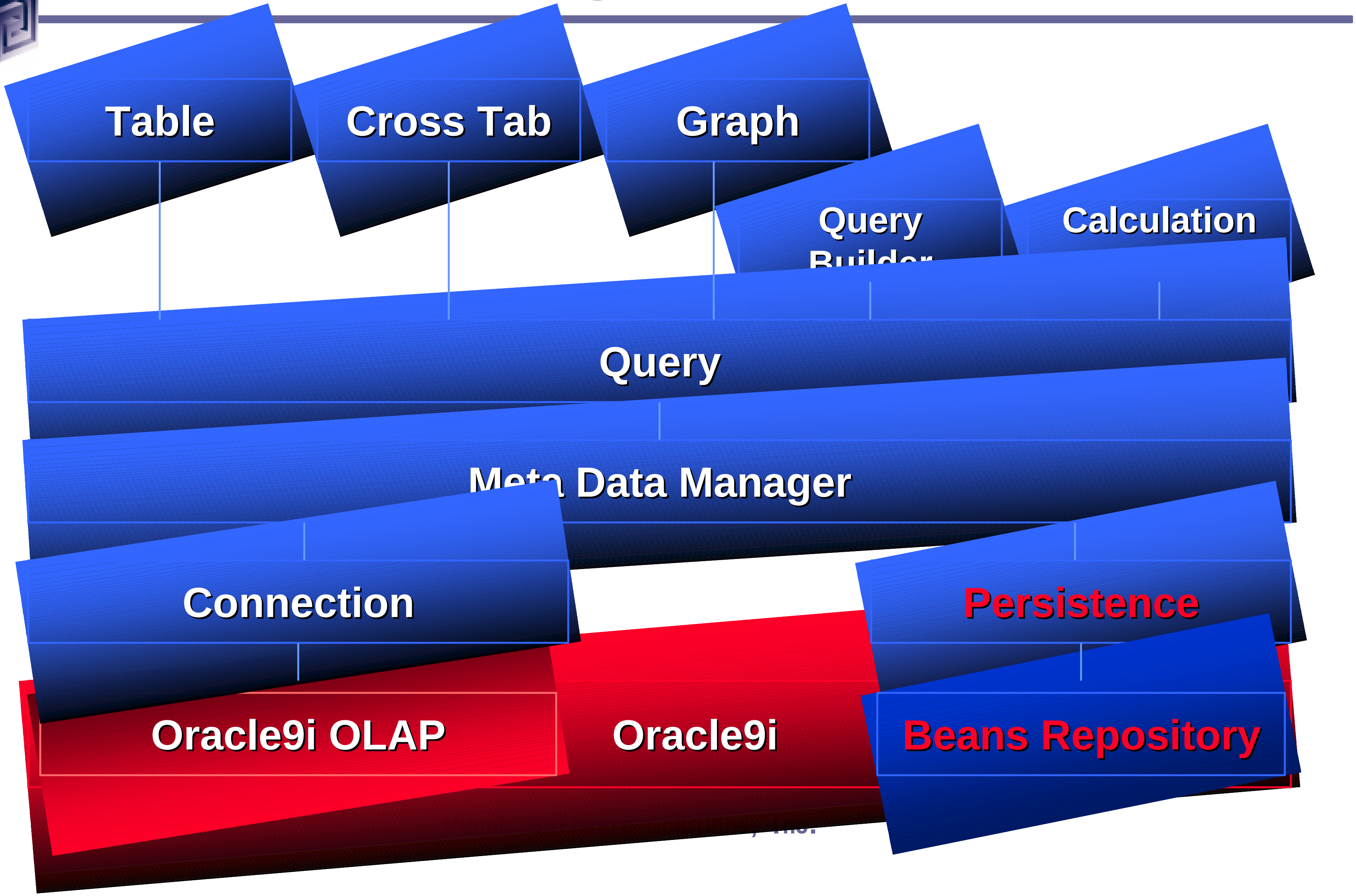
Express

```
limit geography to 'ORLANDO'
limit time to 'MAY2001'
limit product to dollars gt 1000000
```

Java OLAP API

```
Source geogSel = geography.selectValue("ORLANDO");
Source timeSel = time.selectValue("MAY2001");
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



Table

Cross Tab

Graph

Query Builder

Calculation

Query

Meta Data Manager

Connection

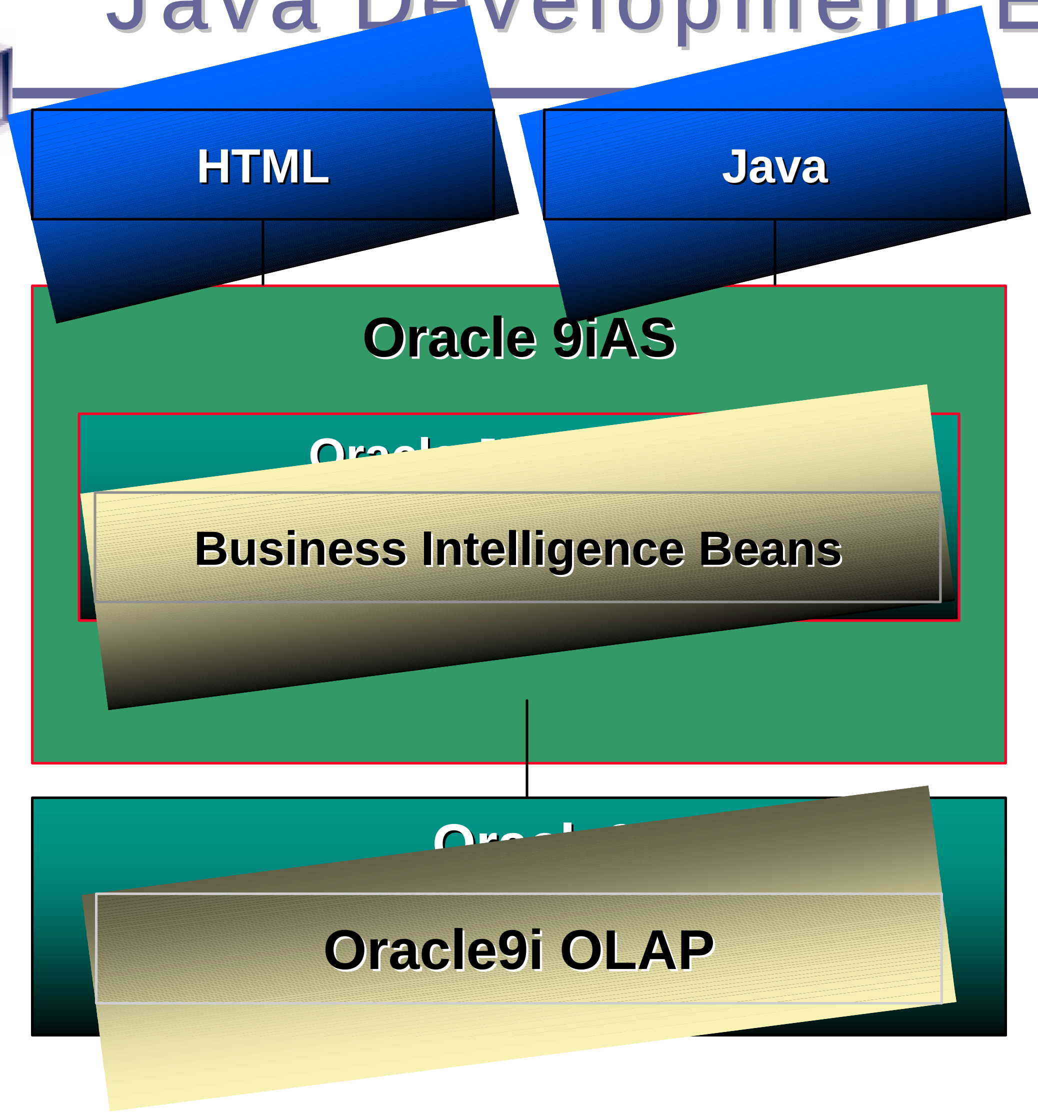
Persistence

Oracle9i OLAP

Oracle9i

Beans Repository

Java Development Environment



Development

Deployment

Database Services

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 of RDBMS data**
- **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**

Analysis Functions

Oracle9i Supports

- **Ranking family**
- **Window Aggregate**
- **Reporting Aggregate family LAG/LEAD**
- **Linear Regression family**
- **Inverse Percentile family**
- **Hypothetical Rank and Distribution family**
- **FIRST/LAST Aggregates family**

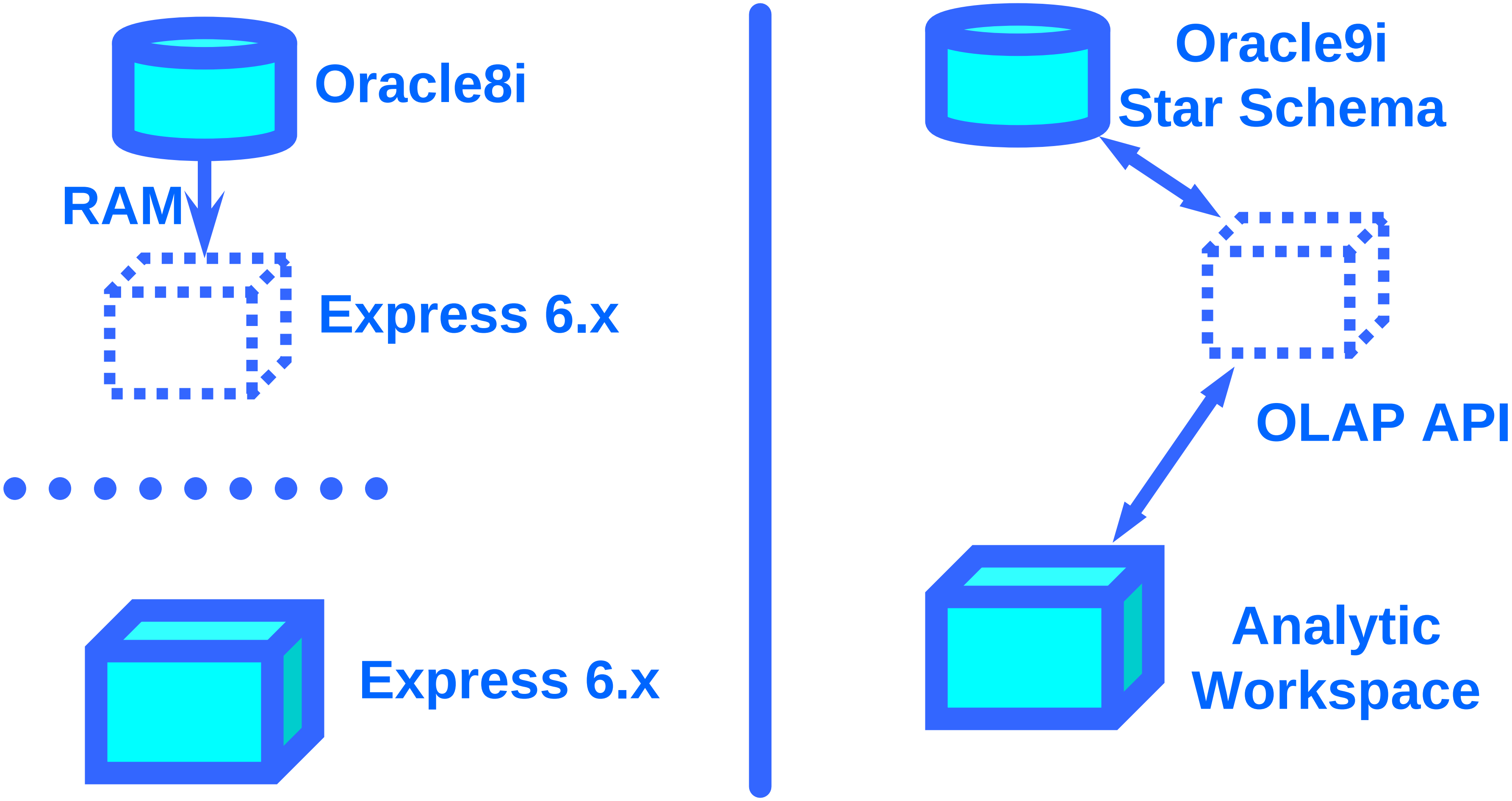
OLAP DML Changes

- 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

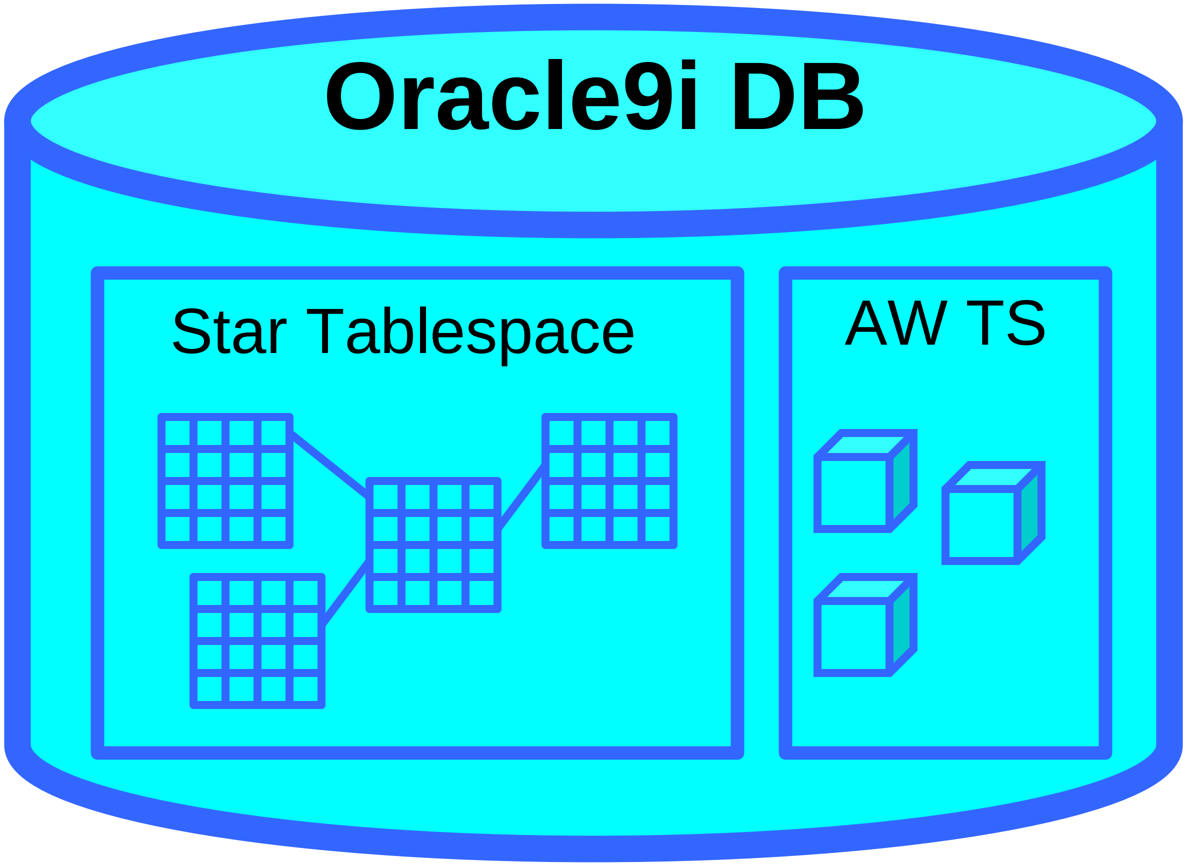
OLAP DML Changes (continued)

- Database is now Analytic Workspace
  - Stored in LOB in RDBMS (9.2)
  - 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 (9.2)

New Data Storage Possibilities in Oracle OLAP 9.2

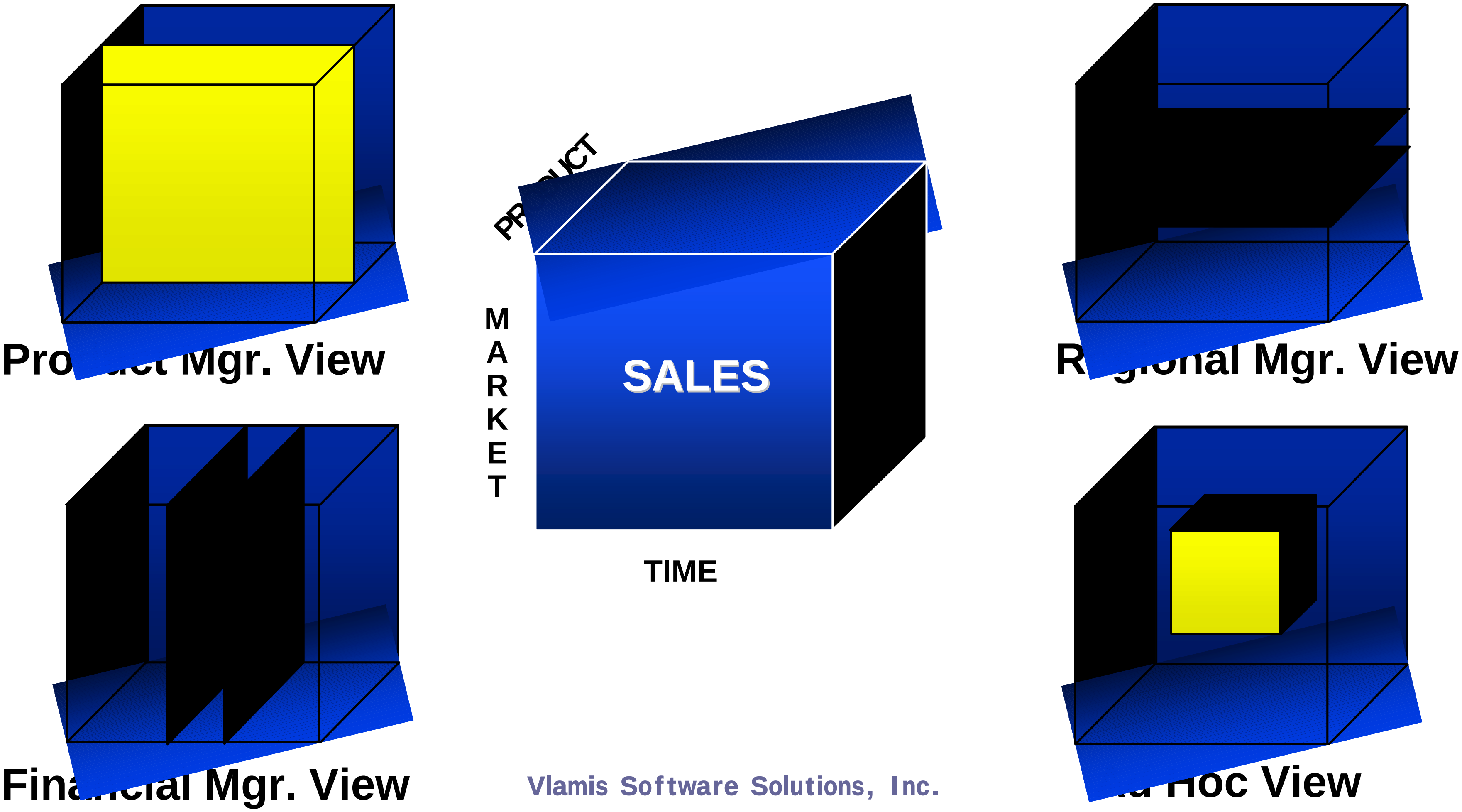


Analytic Workspaces Are Stored in Tablespaces in 9i OLAP 2



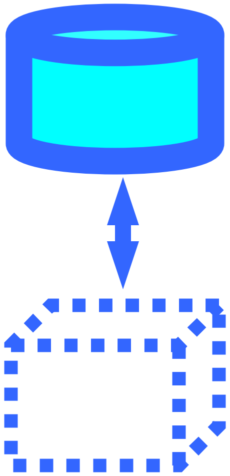
9i OLAP AW Stores Data in Cubes

Fast Flexible Access to Summarized Data



Advantages of RDBMS Storage

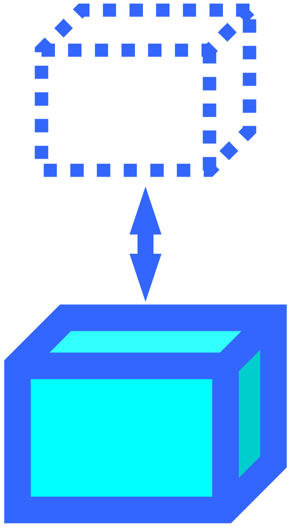
Oracle9i
Star Schema



- 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



Analytic Workspace

Oracle9i 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

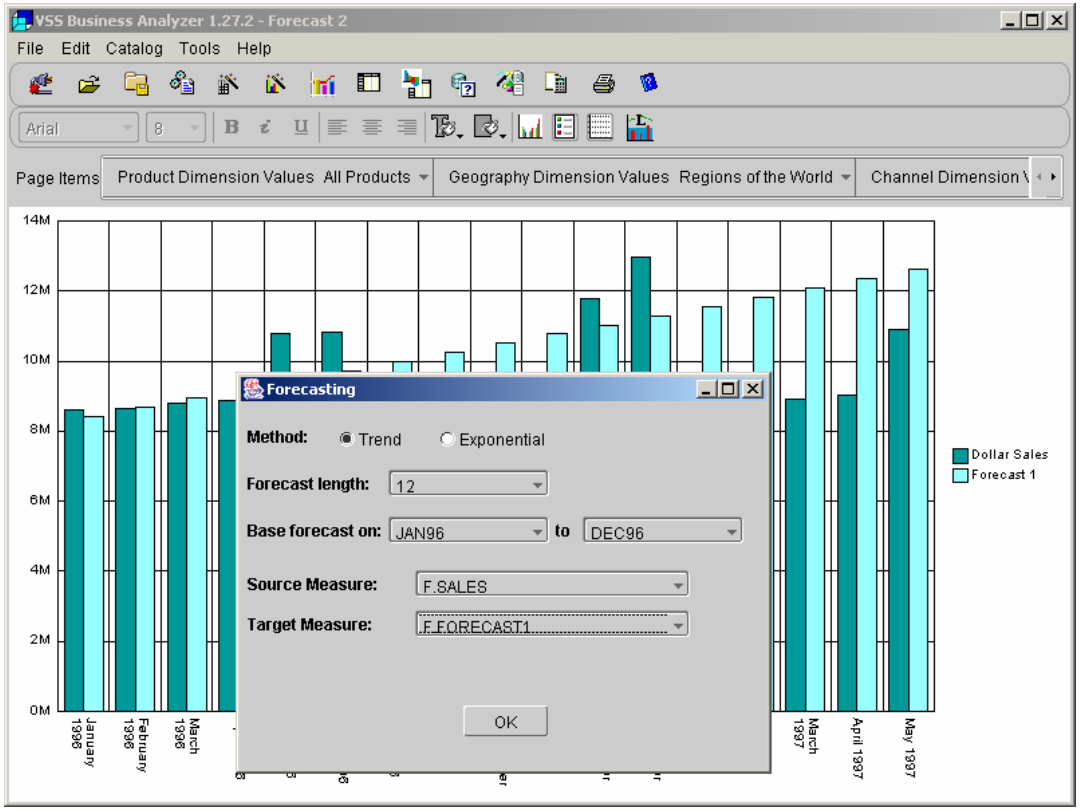
- DBAs often don't want users modifying data
- Modeling data
- User-created calculations
- Multiple scenarios for forecasts
- Allows for collaboration between users

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



VSS Business Analyzer 1.27.2 - Forecast 2

Page Items: Product Dimension Values All Products; Geography Dimension Values Regions of the World; Channel Dimension V

Forecasting

Method: ● Trend ○ Exponential

Forecast length: 12

Base forecast on: JAN96 to DEC96

Source Measure: F.SALES

Target Measure: F.FORECAST1

See 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**

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
- VSS Business Analyzer
  - www.vlamis.com

Effectively Using 9i OLAP in Business Intelligence Applications

presented at

OracleWorld 2002

paper number 32066

Presented by:

Dan Vlamis (dvlamis@vlamis.com)

Vlamis Software Solutions, Inc.
(816) 781-2880

http://www.vlamis.com