

# Oracle Business Intelligence 10g Technical Overview

*An Oracle White Paper  
December 2004*

# Oracle Business Intelligence Technical Overview

Executive Overview .....	3
Introduction .....	4
Phase 1: Consolidation .....	4
Step 1: Map transactional sources to a target data warehouse .....	5
Step 2: Generate the code to extract, transform and load data.....	5
Step 3: Generate the business area .....	6
Warehouse lifecycle management.....	6
Phase 2: Discovery .....	6
Business areas defined automatically .....	6
Authenticity is verified on demand .....	7
Phase 3: Sharing Data .....	7
Query results are shared and personalized.....	7
OracleBI Spreadsheet Add-in.....	8
Generate once and deploy in any format .....	9
Generate once and deploy anywhere .....	10
Expanding Business Insight.....	10
Rapid Development of Custom Applications .....	10
Integrated OLAP, Unified Access.....	11
Data Mining.....	11
Personalization .....	11
Portal.....	12
Oracle BI 10g for business intelligence architecture.....	12
Conclusion.....	12

## EXECUTIVE OVERVIEW

In an increasingly competitive world, as a business executive, you are constantly in need of business intelligence (BI) that empowers your decision makers to act on the information and keep your products and services competitive. No matter how you decide to compete, you need complete business intelligence about your company's operations and its customers to assess the potential opportunities and risks associated with various types of competitive response. To build a BI system for the enterprise, IT professionals would purchase point solutions piecemeal.

Unfortunately, such piecemeal solutions result in proliferating complexity and escalating maintenance costs. For example, each upgrade would require dismantling and recalibrating the system. Furthermore, when multiple disparate vendors upgrade their respective products they rarely account for any impact on the other products that are part of the overall solution. Often the customer is left stuck in the middle with a lot of finger pointing between vendors each time the system undergoes an upgrade, or maintenance cycle. The net result is more time and money to stitch together and maintain point solutions compared with a truly integrated solution.

More importantly, there can be a real business impact for un-integrated systems beyond just the maintenance costs. With various tools using different copies of metadata, you could have multiple users viewing the same data, but arriving at separate conclusions. Consider for a moment the impact of sales territory reorganizations. If the CRM application is updated with the latest territories, but the financial application still points to the outdated territories, the projected revenue reports would yield two different results. Imagine the spirited dialog between the VP of Sales and VP of Finance as they discuss future business directions based on their own reports.

Oracle offers an integrated BI solution that provides the business user with a complete picture across the entire organization. With an integrated solution from a single vendor, customers who implement Oracle Business Intelligence benefit with the ability to:

- **Make the right decisions faster**
  - Easy end-user customization
  - “Self-Service BI” gets the right information to the right person at the right time
- **Enable more employees with the information they need**
  - Consistent, intuitive end-user experience
  - Transparent access to relational and multi-dimensional data
- **Improve Information Quality**

Oracle business intelligence is designed to easily and quickly integrate diverse data sources, find information from the database, share the newfound information, and exploit the data to learn more about the business and its customers.

## **INTRODUCTION**

There are typically three phases in building a BI system: consolidation, discovery, and sharing. In addition, the BI solution should be designed with robustness to include expansion, as the business needs change. At first glance, the three phases may seem simple, but a closer look at each phase reveals potentially complex implementations as the system is integrated. For example in the first phase, consolidation, the number of complications escalates as the number of diverse data sources increases. In addition, the data usually requires a degree of transformation, or name and address scrubbing during data consolidation. Once the consolidated data is prepared for discovery, a myriad of reporting tools must be integrated to deliver business intelligence to the appropriate business decision maker. Such reporting tools include analytical applications, query & analysis, enterprise reporting and lineage. Finally, each tool may have its own high performance engine that must be integrated into the BI system.

By integrating such an intricately complicated system that includes data management, reporting tools, and performance engines from multiple vendors, the organization faces

- Very high software costs
- Expensive and time consuming training
- A wide variety of technology expertise
- Protracted and complex implementations
- Extensive system upgrades and maintenance
- The movement of data between disparate data stores

Oracle Business Intelligence is designed as a single integrated platform for business intelligence. This paper provides an outline of the integrated business intelligence functionality available from Oracle. Included in this paper are illustrations of the tight integration between individual components as they are applied to each phase in building a complete and integrated BI solution from a single vendor that is founded on open standards, takes full advantage of the BI capabilities embedded in the Oracle 10g database and is very competitively priced.

## **PHASE 1: CONSOLIDATION**

Consolidating data is becoming increasingly complex as businesses face mergers and acquisitions, reorganizations, and globalization. The data tends to become spread throughout the entire organization in multiple data sources, making it increasingly difficult to gather business intelligence. Oracle Warehouse Builder (OWB) is designed to consolidate disparate data sources, perform any required data

transformations, manage the warehouse lifecycle, and integrate with the analysis tools. OWB provides capabilities to ensure data quality.

### **Step 1: Map transactional sources to a target data warehouse**

Once the business questions and data requirements of the end-users have been determined, the IT professional starts identifying and connecting to the data sources containing the data to meet these requirements. Connections to flat file sources, or relational data sources such as Oracle, DB2, Informix, Microsoft SQL Server, Sybase, and SAP R/3 are made possible by simply selecting the appropriate OracleBI Warehouse Builder (OWB) integrator. The user then creates source modules that store data definitions containing both connection information and table definitions. Once connected to the data source, the integrator is used by OWB to extract data and metadata from that source. The other type of module the user creates with OWB is the warehouse module, which contains definitions of the facts, dimensions, staging tables, etc. that make up the data warehouse. Together, these two types of modules form the central elements used in warehouse design. The user can simply drag and drop objects from the modules to use as building blocks in designing a target warehouse. After the objects of interest are selected, OWB provides a graphical mapping environment where users can model all aspects of a mapping in the form of a dataflow diagram. These mappings include participating source and target tables and all other operations that apply to the extraction, transformation, and loading (ETL) process. OWB can perform the ETL operation in one complete step, making warehouse deployment fast and easy.

Using OWB to map data sources to targets is easy with a graphical Mapping Editor, where the user visually designs, or models all aspects of the ETL operation. Developer productivity is enhanced as the user designs complex transformations, inline expressions, multiple joins, aggregations, etc. with an intuitive interface, without requiring specialized SQL programming knowledge.

### **Step 2: Generate the code to extract, transform and load data**

Once the mapping models are completed, OWB can generate the SQL and PL/SQL code to instantiate and populate the data warehouse, both Relational and OLAP targets. This saves time and reduces the level of expertise required to program SQL code. Furthermore, OWB is open to allow users to inspect the code at any step in the ETL operation. This adds to the flexibility in the event a customized operation is required. In addition, the OWB Code Generator can generate code that's optimized with respect to the data source. For example, if the data is clean with no anticipated conflicts, the user simply clicks the set based code generation option and high performance code is generated to load the data as an entire set. On the other hand, if the data has high potential for constraint violations, or other load errors, the user clicks on the row based code generation option and instantly the code is rewritten to load data on a row-by-row basis. Because the PL/SQL packages are automatically generated and do not require any

specialized PL/SQL programming knowledge, the tool is available to more users, reducing the cost of ownership and increasing the productivity of the department.

### **Step 3: Generate the business area**

Now that the data is consolidated and loaded into a target warehouse, the multidimensional design is easily shared with Oracle reporting tools. All repository and runtime metadata is exposed via public views. This means that Oracle reporting tools can “understand” the cubes, dimensions and hierarchies as well as all table names and column headers. As a result, valuable programming time is saved because the user is not required to recreate the metadata when deploying the Oracle10g reporting tools, as would be the case with point solutions.

### **Warehouse lifecycle management**

Once the warehouse is deployed, managing changes to the warehouse is essential. Inefficient maintenance of the warehouse can strain any IT budget. Changes, such as tables, columns or views in the source are added, dropped or modified. When these changes occur, most ETL tools today can only detect something changed between the source and target, but leave the users to reconcile the differences. OracleBI Warehouse Builder, on the other hand, reconciles any changes in the source and target objects. For example, if a change occurs after a source module was created in the OWB repository, the repository definitions are out of sync with their corresponding source objects. The re-import function of Oracle Warehouse Builder automatically reconciles and restores the synchronicity of definitions in the repository for the source objects. Or, the warehouse upgrade function synchronizes the definitions in the repository with the target objects. The user saves time maintaining the warehouse, constantly trying to keep track of source data changes and extending those changes to the target database.

## **PHASE 2: DISCOVERY**

Now that the data is located in one place, the powerful ad hoc query and analysis tool, OracleBI Discoverer unleashes potential opportunities and risks associated with your products, customers, and marketplace.

### **Business areas defined automatically**

Discoverer hides the complexity of underlying database structures like OLAP cubes, tables, columns, joins, etc. by presenting a business-oriented view of the data. Business users create reports by opening up subject oriented business areas represented by file drawers and folders, moving selected items onto a worksheet. Logical hierarchies, calculated Items, join definitions, custom sort orders and more enable users to perform otherwise complex tasks by simply clicking and dragging. You create a business area in OracleBI Discoverer Administrator using the Load Wizard or, as previously discussed, using OWB.

The tight integration between OracleBI Warehouse Builder and OracleBI Discoverer allows users to easily populate business areas by using a wizard-based interface. The dimensions and hierarchies created in OWB are understood by Discoverer, which improves productivity and provides faster deployment to the end users for ad hoc query and analysis.

### Authenticity is verified on demand

Providing lineage reports extends the integration with OWB even further. Discoverer users can drill from their queries to the metadata definition in OWB and can view the objects impacted by a particular data transformation. For example, if the aggregated revenue from a region abroad were questionable, the user could drill on the item to view a graphical representation of all the calculations, or other transformations used to calculate that revenue. Now, users can create their own queries and easily verify the validity of the results in the same session.

### PHASE 3: SHARING DATA

Often much time and effort by the users are expended in the discovery phase where just the right query and analysis is performed. Once the right information is generated, sharing the information across the enterprise before it becomes stale can be challenging.

### Query results are shared and personalized

OracleBI Discoverer promotes sharing the queries with the enterprise via its tight integration with Oracle's publishing tools. The tight integration between OracleBI Discoverer and OracleAS Portal empowers users to publish their favorite reports or list of reports to OracleAS Portal. Other users in the organization easily access the information by browsing these portal pages. Users publish their reports as portlets as a list or content. "List of workbooks portlet" provides a list of workbook/worksheet names, "Worksheet Portlet" contains data -- table or cross-tabular report, a graph, or both, and "Gauges Portlet" enables users to easily visualize the data in a speedometer as shown below.

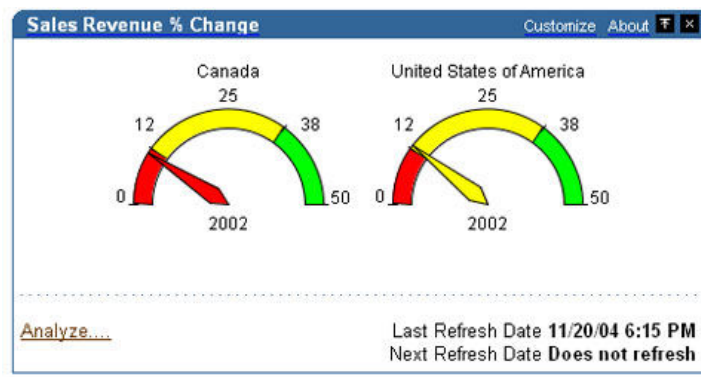


Figure 2: An example of a gauge portlet for designing performance dashboards

Information to be published is often derived from data in corporate data sources, which may be SQL based (relational databases) or non-SQL based, such as XML data. It is often necessary to combine data from additional sources like Oracle10g OLAP, Web Services, Text files, or JDBC data sources. Oracle Application Server Reports permits multiple queries in the same report, where each query can be based on a different data source. Developers can also add additional java-based data sources (called Pluggable Data Sources).

### **OracleBI Spreadsheet Add-in**

The OracleBI Spreadsheet Add-In enables end-users to display and navigate Oracle OLAP data directly from within Microsoft Excel™. The users can treat the Oracle OLAP data as regular Excel data, for example create formulas and graphs, enabling them to combine the powerful analytic capabilities of Oracle OLAP with standard Excel functionality. Here, Excel acts as an intelligent front end connected directly to an Oracle OLAP enabled database.

The OracleBI Spreadsheet Add-in is installed similarly to other Excel Add-ins - but the real application power and flexibility remains contained within the Oracle server and requires no local maintenance. The add-in does not stop Excel being used in the normal way, and it is usually possible to have several add-ins installed without interfering with each other. Most add-ins provide their own main menu entry and relevant, right-click context menus. Many also have one or more tool bars, which can be turned off when not required.

The main advantage of using OracleBI Spreadsheet Add-in is it allows users to create, manage and execute queries directly against the OLAP dimensions and measures using all the processing power of the database. This allows users to query extremely large database warehouse instances, which would normally be beyond the capabilities of Excel, due to its well-documented structural limitations. Once a user has built a query the data is presented within Excel as a normal spreadsheet, with additional controls for dimension paging. Since the OLAP query looks and feels exactly like a normal Excel spreadsheet the user is free to enhance and extend the query using the normal Excel business tools such as adding a chart, as can be seen below in Figure 1.

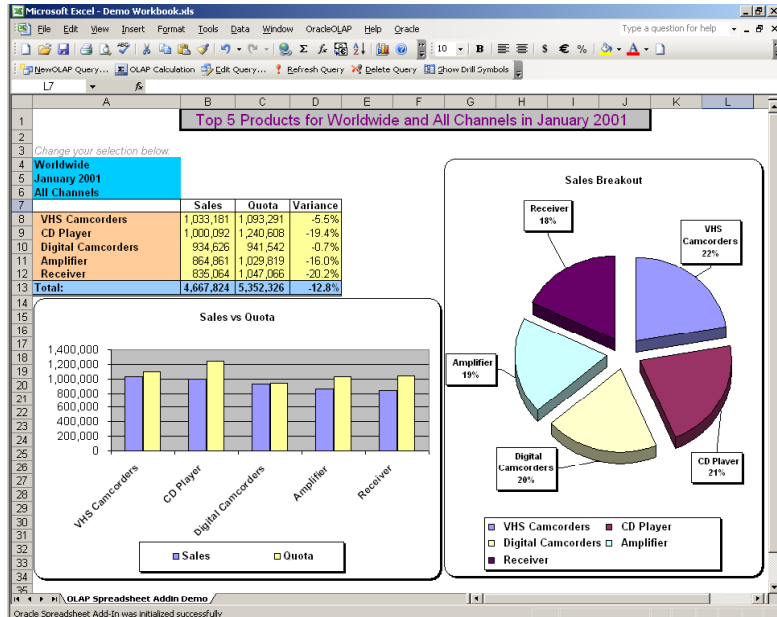


Figure 1: OLAP Spreadsheet Add-in report using OLAP data and Excel charts

Having created a complex report, users do not want to rebuild their queries each time new data is loaded into the warehouse. By using query conditions spreadsheets can become dynamic in nature, providing the user with the ability to continually refresh with the latest data without the need to rewrite the query. Using Oracle's OLAP Query Builder users can construct queries using business terms and definitions. Both calculations and queries are based around measures and their related dimensions. Using a wizard driven interface, users can select data from Oracle OLAP simply by choosing from a list of values, or by creating advanced selections, such as exceptions, top/ bottom or hierarchy-based queries. In addition, the user can create Oracle OLAP based calculations using a similar wizard.

The built-in Query Wizard is based on the Oracle Business Intelligence Beans Query Wizard. This wizard is used across all the Oracle OLAP products, ensuring a consistent interface with other Oracle Business Intelligence Beans based applications.

### Generate once and deploy in any format

Reports has always been a powerful tool to publish data in a variety of formats: PDF, XML, HTML, HTMLCSS, Postscript, PCL, Delimited text, and RTF. Now, users can publish data using industry-standard JSP's. This means that developers can create a Web page template using an HTML editor, then bring it into the OracleAS Reports environment and integrate data from various data sources into that Web page. This includes using business logic and your own corporate styles to fashion and control a report's appearance. Of course, a JSP doesn't have to return

HTML, as it can be used to create any text-based output format (e.g. WML). Finally, because JSP's are used, when a report is updated, all the Web pages tagged with that report are updated instantly.

### **Generate once and deploy anywhere**

Printer, Web, file-system, Email, and OracleAS Portal are all destinations that Oracle Application Server Reports Services has supported for some time now on a wide variety of platforms. With OracleAS Reports Services, as well as greatly enhancing the product's out-of-the-box emailing capabilities, developers are now able to create their own java-based extensions to the Reports Server to open it up to previously unsupported destinations such as Fax, FTP, WebDAV, etc. The broadcast of this information can be done on-demand or scheduled, or even as a reaction to an event that has occurred within an Oracle10g Database, Oracle Workflow, or an application that communicates via Web Services. Email and wireless notifications of the report success/failure are available, and once again, developers can plug-in their own java-based notification mechanisms. Of course information security is of paramount importance, and with OracleAS Reports, administrators have the flexibility to secure Reports, Report Servers, and printers for fine-grade control of who can do what, where, and when.

### **EXPANDING BUSINESS INSIGHT**

Reacting quickly to business needs is critical. Sometimes it is necessary to build custom applications to allow insight into real-time business performance in order to make better business decisions. OracleBI Beans in Oracle JDeveloper (JDeveloper) is designed specifically to develop powerful business intelligence applications that are integrated with all Oracle business intelligence tools. Developers can quickly assemble an integrated BI application from high level, reusable components while taking advantage of Oracle10g OLAP capabilities. Oracle development teams make extensive use of OracleBI Beans in the building of the query analysis and reporting tools, these beans are also provided to your developers so that any custom application can keep the same look and feel.

### **Rapid Development of Custom Applications**

JDeveloper and OracleBI Beans provide the most productive development environment for building custom business intelligence applications. OracleBI Beans provide high-level components that encapsulate OLAP business logic for querying and presenting data. Data can be presented in tables, crosstabs, and graphs. Using BI Beans QueryBuilder and CalculationBuilder, developers easily access the advanced analytic capabilities of Oracle10g OLAP for generating queries. The combination of these flexible query and presentation options make it very easy to visually design OLAP objects. Wizards are provided in JDeveloper that package these components into applications in the form of either servlets or Java applications. A BI Beans tag library is also available enabling the rapid development of JSPs.

## **Integrated OLAP, Unified Access**

Advanced analytics are the heart of any business intelligence application. Oracle10g OLAP provides centralized analytic processing in a scalable and secure environment. The BI beans make it easy to formulate those business questions involving complex multidimensional queries. The results of such queries are then displayed in both crosstab and graphical presentations using the presentation beans. The QueryBuilder is a powerful, patented tool that enables users to specify query properties without expertise in SQL. Complex multidimensional queries are made simple by presenting the query definition in business terms, which end users can modify to meet their needs. For example, a default query definition called “Top 5 Products based on Sales” can be modified to “Bottom 10 Products based on Sales” by selecting the appropriate query commands from a drop down list. With unlimited query possibilities, users quickly leverage advanced database analytics to ask complex business questions of their data. Utilizing the BI Beans Catalog, users save and share their discoveries with others across the enterprise. Users are provided the same unified access to their relational and OLAP data sources regardless of which tool they use - OracleBI Discoverer, OracleBI Spreadsheet Add-in and OracleAS Reports Services provide support for a transparent access to all your data.

## **Data Mining**

Oracle10g Data Mining allows companies to build advanced business intelligence applications that mine corporate databases to discover new insights and integrate those insights into business applications. The Oracle10g Database has embedded data-mining functionality like classifications, predictions, and associations. This allows application developers to integrate data-mining capabilities into their business intelligence applications to support such activities as:

- Preventing customer attrition
- Cross-selling to existing customers
- Acquiring new customers
- Detecting fraud
- Identifying the most profitable customers
- Profiling customers with more accuracy

Oracle10g Data Mining opens the door for integrating sophisticated data-mining capabilities, once the domain of specialized servers, into mainstream business intelligence applications.

## **Personalization**

OracleAS Personalization is more than “business rules” or collaborative filtering. It is delivering “individualized” recommendations, advertisements, and content delivered with the touch and timing of a close personal friend. Its scope extends

beyond recommending similar items based on a collection of clicks from potential buyers. Personalization brings together the entire customer experience, which includes “clicks”, purchases, returns, complaints, support requests, ratings, wish lists, and demographics. An automated, timely, and non-intrusive recommendation based on a complete view of the customer is true personalization.

Developers use OracleAS Personalization’s Recommendation Engine API to equip a Web site to tag, or capture a visitor’s “clicks” and to request real-time recommendations. These “click” data are combined with historical data, if available within the OracleAS Personalization schema, and passed to the OracleAS Personalization Recommendation Engine. The Recommendation Engine searches for “rules” or recommendations that best fit the current session and historical data scenario and passes the recommendations to the Web application in a fraction of a second. Integration with the Oracle10g Database delivers power and scalability with a minimum of data redundancy. OracleAS Personalization collects and stores the data, then builds predictive models all within the Oracle10g Database. Next, the powerful data mining technology embedded in Oracle10g Database automatically discovers individualized behavior patterns to generate highly accurate personalized recommendations in real time.

## **Portal**

OracleAS Portal is a complete framework for development and deployment of Web-based portals. It includes user administration, security, content customization, and development features to create and maintain basic reports, charts, and form-based applications. Creating a Business Intelligence dashboard personalized by job role is easy with OracleAS Portal. Charts and/or reports representing key performance indicators (KPI’s) can be rapidly developed. These charts and reports are deployed as Portlets. Individual users may customize their portal presentation by selecting the KPI portlets that are most relevant to their management focus. If users are given the authority, they can change the KPI’s to suit their own requirements and portal will handle the individual customizations seamlessly.

## **ORACLE BI 10g FOR BUSINESS INTELLIGENCE ARCHITECTURE**

Oracle provides a complete, integrated, open and affordable Business Intelligence solution out of the box, as well as an open and extensible architecture for developing analytical applications. All tightly integrated with Oracle10g Database for superior reliability and scalability.

## **CONCLUSION**

Oracle Business Intelligence continues to enhance the industry leading Oracle database platform with capabilities to support business intelligence systems. Today’s managers and data analysts are inundated with information and under time

constraints to make key business decisions. They need visual cues to identify the latest business trends and then make strategic management decisions. All of which must be shared with the right people throughout the enterprise. When the tools used for data consolidation, data analysis and enterprise reporting are integrated and available from one vendor, Oracle, it is clear that companies deploying Oracle Business Intelligence are armed for success. They have the capability to deliver the right information to the right person at the right time, at the right price.



Oracle Corporation  
World Headquarters  
500 Oracle Parkway  
Redwood Shores, CA 94065  
U.S.A.

Worldwide Inquiries:  
+1.650.506.7000  
Fax +1.650.506.7200  
<http://www.oracle.com/>

Copyright © Oracle Corporation 2003  
All Rights Reserved

This document is provided for informational purposes only, and the information herein is subject to change without notice. Please report any errors herein to Oracle Corporation. Oracle Corporation does not provide any warranties covering and specifically disclaims any liability in connection with this document.

Oracle is a registered trademark of Oracle Corporation.

All other company and product names mentioned are used for identification purposes only and may be trademarks of their respective owners.