# ORACLE WAREHOUSE BUILDER 10G AND OLAP – WHAT'S NEW

*Chris Claterbos, Vlamis Software Solutions, Inc.*

## INTRODUCTION

With the use of the new features found in recently updated Oracle's Warehouse Builder (OWB) and Oracle 10g we can design, construct and load a data warehouse that is fully OLAP ready. This presentation will introduce the attendee to OWB 10g Release 2 and Oracle OLAP and how to design and construct a Oracle 10g database that can be used for OLAP analysis and presentation. The many new features that improve on the ability to build OLAP warehouses taking advantage of the many new features in Oracle 10g will be presented. Implementation and design issues will also be addressed. A step-by-step guide will be presented with the end result being a completed OLAP based datamart. A complete end-to-end demonstration will be presented.

## WHAT IS OWB?

The major functional areas of data warehousing from ETL tools to data storage is contained within the OWB framework. OWB is a key component in Oracle's Intelligent Webhouse initiative, enabling the design and deployment of data warehousing and business intelligence solutions. The core of OWB is the Common Warehouse Metadata, which is the open metadata interchange standard driven by the OMG partners such as Oracle, IBM, NCR, and Unisys.

OWB provides a comprehensive framework for designing and deploying data warehousing and business intelligence applications across the enterprise. It consists of the following functional components:

### REPOSITORY

All of the user's work is stored in an Oracle 10g based repository. The repository is based upon an interim release of the Common Warehouse Metadata (CWM) standard. This repository consists of a set of tables that are stored in the Oracle 10g database that all OWB users and runtime applications share.

### GRAPHICAL USER INTERFACE

The OWB client is used to perform all tasks necessary for the design and administration of the OWB project. It is written entirely in Java and can be run from any machine that can run the Java runtime environment.

### CODE GENERATION

OWB generates the code for creation of the database objects as well as the code to transform and load the data into the data warehouse.

### INTEGRATION

There are several components within OWB that function as data integrators. A majority of these integrators are dedicated to the extraction of data from a particular type of data source. The primary integrator provides support for Oracle relational data sources. Other relational, non-relational and flat file sources are also supported. Specialized integrators are also supplied to support Oracle Applications as well as SAP R/3. In addition to the data extraction integrators there is also integration with Oracle Business Intelligence applications, Discoverer and Oracle OLAP. It is the use of this functionality that will be addressed in this paper.

## OLAP INTEGRATION – WHAT IS NEW IN THIS RELEASE

Warehouse Builder enables you to prepare your data stores for Oracle10g OLAP processing. Using Warehouse Builder, you can design, deploy, and load online analytical processing (OLAP) objects that provide complex analytic power to your data warehouse. This release of OWB is much different than previous versions with respect to the OLAP integration and generation. In previous releases an external tool is used to "Bridge" to OLAP. The new release is fully integrated with the OLAP option.

## EXAMPLE SCENARIO

One of the easiest ways to illustrate how to build an OLAP database with OWB is to show the process using an example application. The following sections show the process that is used to create a database design, validate the design, create the tables in Oracle 10g, transfer the design to OLAP and create the OLAP database.

### CREATE THE DATABASE DESIGN

Once the main screen is loaded, you then select the module that you will work with. OWB uses a concept of a module, which is a logical grouping of related objects. The warehouse module is a collection of the definitions of the objects comprising the data warehouse. When this is selected the Module Edit screen will display the objects that are part of the module. Definitions for such things as dimensions, facts, staging tables are all contained in the warehouse module and are shown in Figure 2.

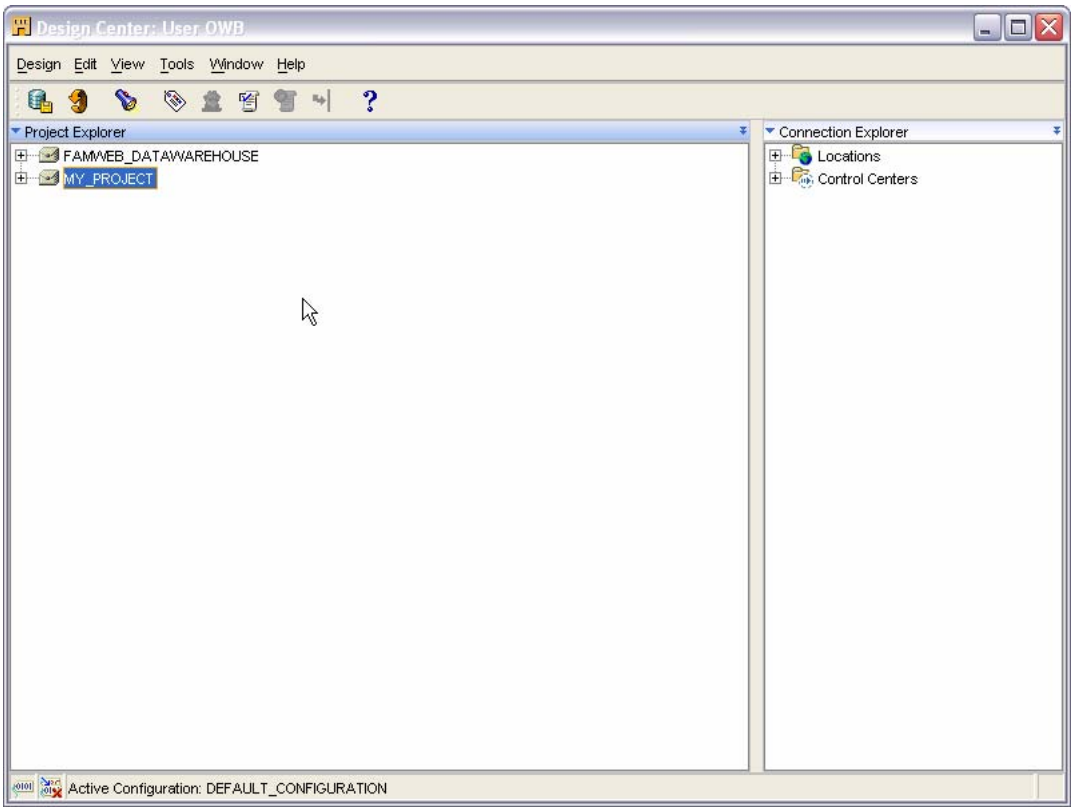


*Figure 1. OWB Main Screen*

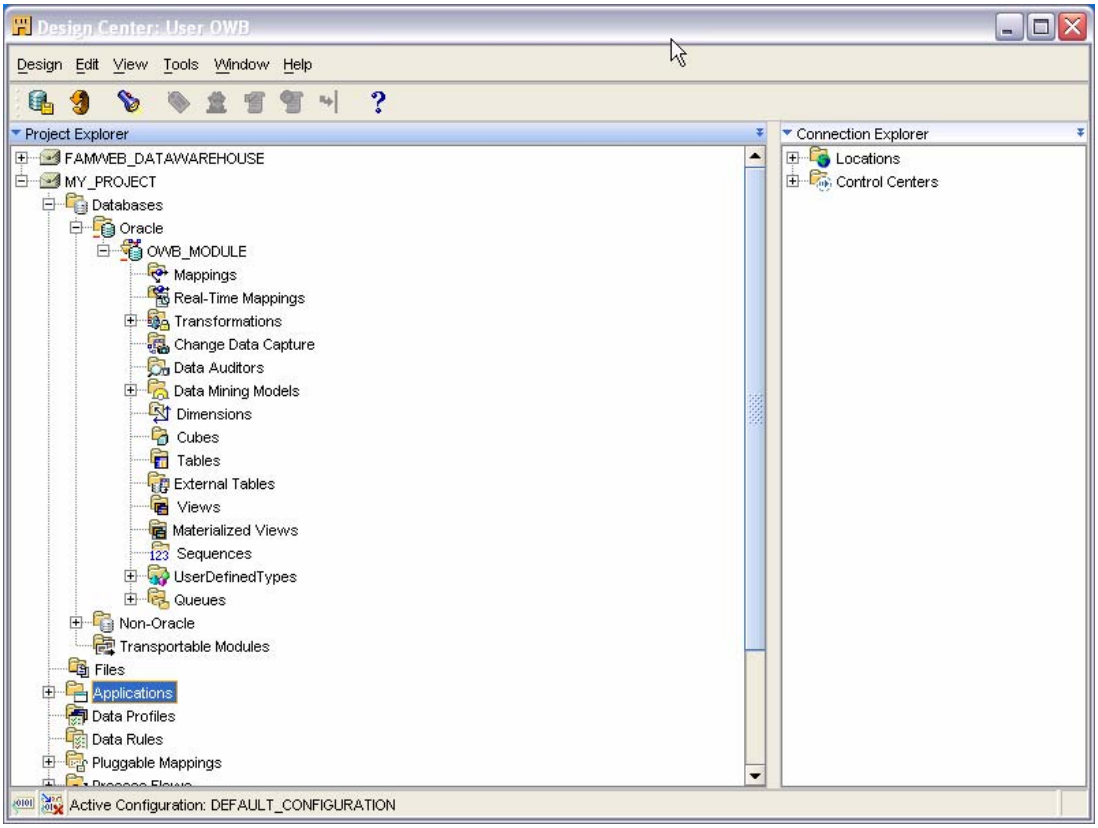


*Figure 2. Module Editor*

The first thing that is created in the design are the various dimensions that will be used by the fact tables. Once the dimensions are defined then the facts can be specified.

When you create a dimension, use the extensions listed in the table below while defining the implementing column for the dimension level attributes. Table 1 lists the level attribute names that are generated as OLAP compliant level descriptors when you deploy the dimension to the Oracle10g OLAP catalog using the Warehouse Builder Transfer Wizard. Don't worry, OWB creates all the attributes required automatically.

The default level attributes generated will include; ID and Name. The ID field represents the level key and the NAME field represents the level long description. Time Dimensions will have additional attributes for Timespan and End Dates. The names are associated to an attribute description. It is the attribute description that is critical in this process. It is the description that tell OWB is the attribute is one of the attributes that have reserved meanings. Table 1. contains the list of OWB Attributes what they represent in the resulting Analytic Workspace.

***Table 1 Dimension Attribute Suffixes***

| Physical Level Attribute Name Suffixes in Warehouse Builder | Dimension Attribute Created |
|---|---|
| _NAME or NAME | Short_Description or Long_Description |
| _END_DATE or END_DATE | End_Date |
| _TIME_SPAN or TIME_SPAN | Time_Span |
| _PRIOR_PERIOD or PRIOR_PERIOD | Prior_Period |
| _YEAR_AGO_PERIOD or YEAR_AGO_PERIOD | Year_Ago_Period |

It should be noted that the attribute object names created in the AW do not match exactly the names created when using Oracle Analytic Workspace manager. If this is a concern then OWB does allow you to rename the attribute. The critical thing is to make sure that the attribute type is not changed.

Once the dimensions are defined then the facts can be specified. Warehouse Builder enables you to define a cube using the New Cube Wizard. The fact contains all of the related measures that will have the same dimensionality. Figure 3. Illustrates a simple sales fact table design with one measures, amount_used, and three dimensions.

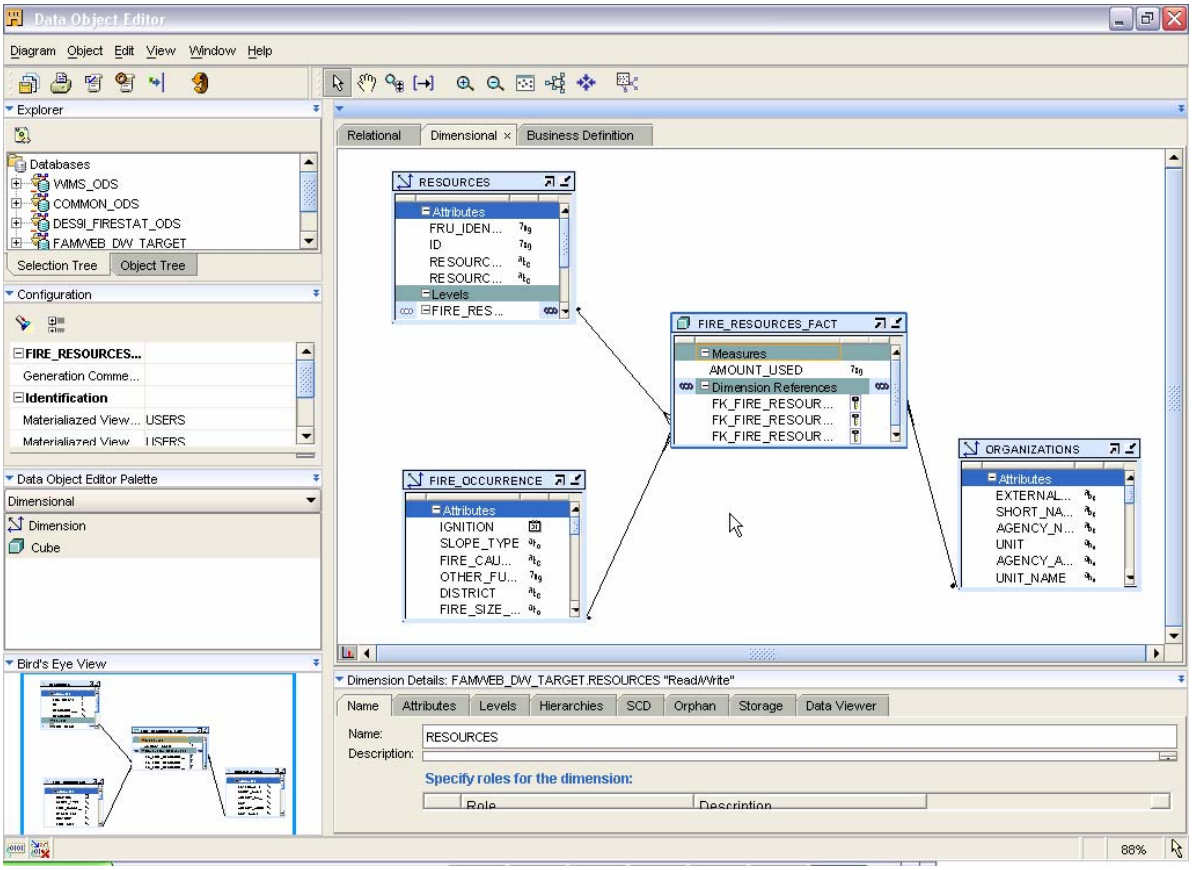


*Figure 3. Fact Editor*

When the design is complete the work is saved to the repository and if desired the design can be validated to make sure that the specifications are correct and consistent. The physical database scheme DDL can now be generated as well creation of all of the physical database objects. After all of this has been completed the design is now ready to create a database that can be made OLAP aware. The relational database is now generated and the relational tables populated with data.

## OWB AND 10G OLAP INTEGRATION

The integration of OWB and 10g OLAP is performed through OWB as well. In previous version the OLAP integration was done externally. OLAP is not fully integrated with OWB! The designer can now choose how the OLAP cubes and dimensions will be implemented, ROLAP or MOLAP. If the Cubes will be implemented via MOLAP, i.e. stored in an Analytic Workspace (AW), then additional implementation details can be made to control storage and data loading. If a 10g database will be used new OLAP features such as compressed composites and data partitioning can be utilized in the design. Such features can have a major impact in the storage and load times. Some of the screens supporting this can be seen in Figure 4.

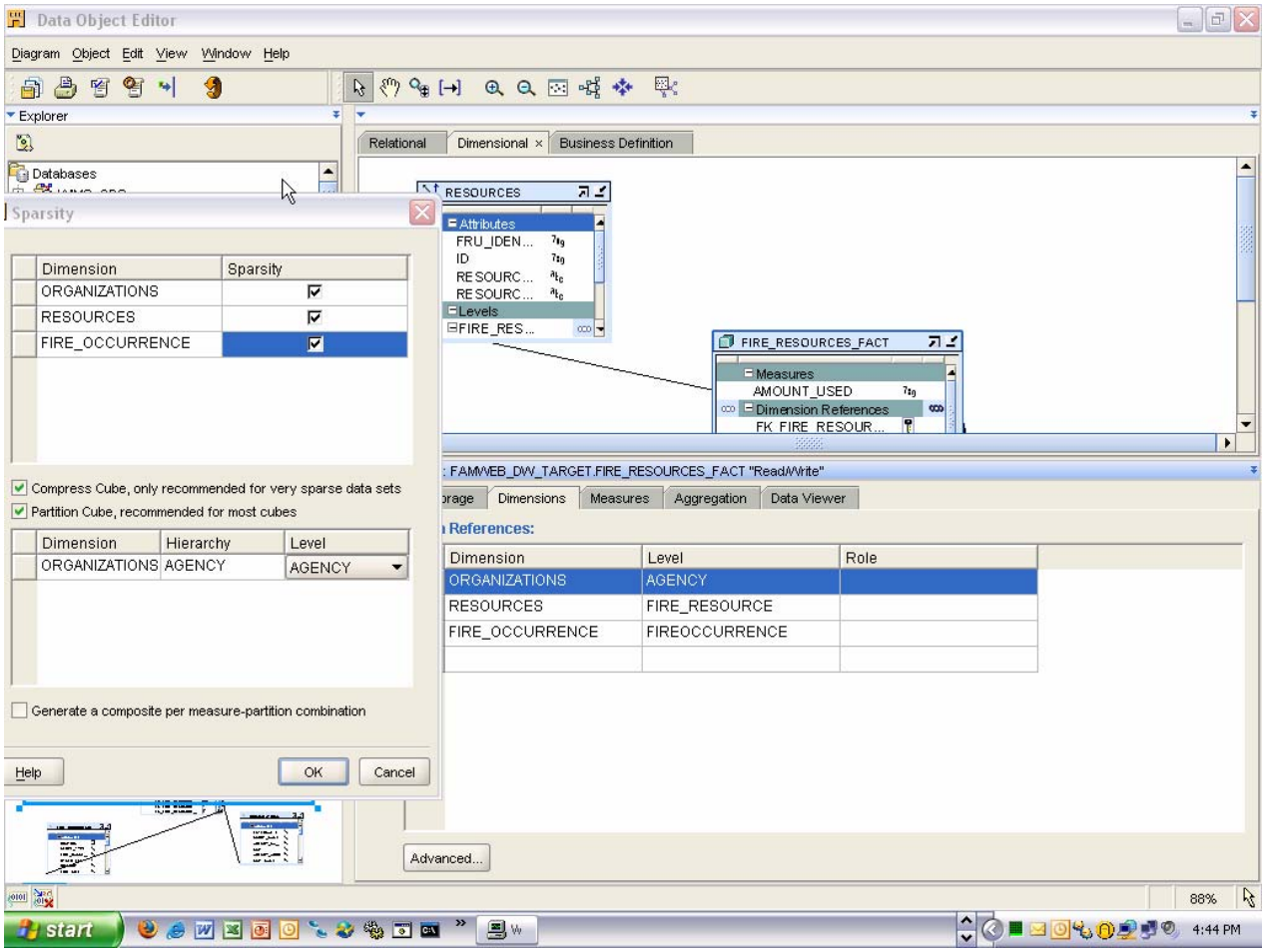


*Figure 4. OLAP Cube implementation details in OWB*

## OLAP MANAGEMENT

Once OWB have accomplished their tasks and the AW is created OWB allows for the creation of simple presentations to view the data. OWB also has screens that allow for viewing the data in dimensions and cubes without having to leave the tool. This is a major improvement over previous versions.

OWB 10g R2 introduced a new set of features under the Business Intelligence topic areas. This includes Business Definitions and Presentations. The Presentations area allows for creation of Presentation Templates in the tool without having to go to other tools. Figure 5 Shows the tree of the various items that are now available.

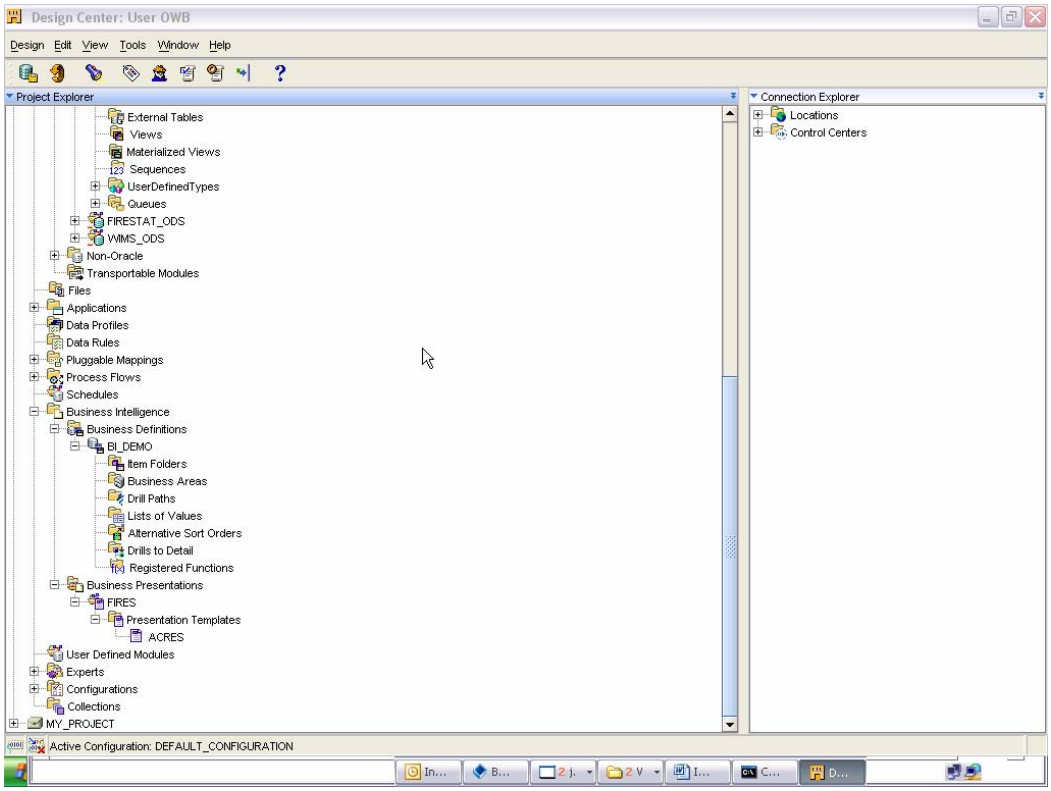


*Figure 5. List of Business Intelligence elements*

The users now view the Cubes, which represent the tying of the Fact Tables and the Dimensions together. The completed Cube contains measures that are ready to be viewed via OLAP front ends such as BI Beans.

Additionally, the user can group the data into Folders that allow multiple cubes of data to be presented to users as in a single folder. This makes it much easier to present measures and other objects to users in a more organized fashion as well as implement various security models.

After the data has been loaded into the relational fact and dimension tables it is immediately available to be viewed using any of the various OLAP front-end Query tools like BI Beans and Oracle Discoverer for OLAP.

## CONCLUSION

OWB is the cornerstone of Oracle's current and future data warehouse and business intelligence initiatives. The future promises more integration and tools that will make the designer and applications developer easier and provide a fully integrated view of enterprise data.

The current implementation of OWB represents a major improvement in the integration with the OLAP analytical engine. It now includes creation of Analytic Workspaces in addition to the creation of relational dimensions and cubes. Warehouse designers can now use one design tool to design, build, populate and maintain the entire enterprise data warehouse and data mart.