

Oracle Warehouse Builder 10g Release 2

What is an Expert?

May 2006

Note:

This document is for informational purposes. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described in this document remains at the sole discretion of Oracle.

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. This document and information contained herein may not be disclosed, copied, reproduced, or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

Oracle Warehouse Builder 10g Release 2

What is an Expert?

INTRODUCTION

Oracle Warehouse Builder 10gR2 (OWB) provides a major leap forward in technology, aimed at helping users meet their organization's requirements to derive actionable knowledge from its data assets (infusing quality into the entire process).

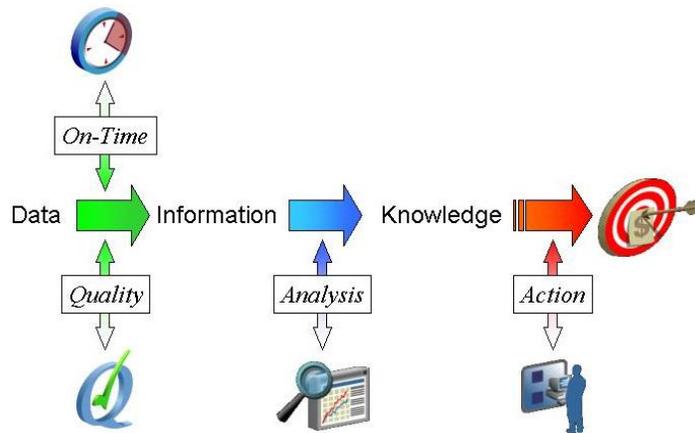


Figure 1. Data to Information to Knowledge to Results

As with all powerful technology, a certain level of knowledge is needed on the part of the user to fully leverage the tool's capabilities and thus their investment. This means that business requirements from end-users are passed through to individuals trained in the technology who translate and fulfill them, and typically in the context of OWB, these users are within the IT department. These business requirements range from the very simple to the very complex, but it's the IT department that typically needs to action them all.

This presents some interesting challenges. First: the IT department can rapidly become a bottleneck to fulfill requirements (and thus impedes the organization's ability to operate at peak efficiency). Thus there is an ever-present pressure to empower as broad an audience as possible to have them fulfill their simpler needs, allowing IT to focus on fulfilling the more complex. Second: all organizations

(either formally or informally) have guidelines/best practices and standard operating procedures (SOP's). These provide members of the organization a level of consistency and stability within the framework they execute in, which in turn helps the organization as a whole to operate more efficiently.

These standards and best practices extend not just to an individual's behavior, but often right down to the technology itself, e.g. coding style and the naming of variables in code (in this case to promote a broader audience capable of maintaining the code). Standards and best practices obviously can only be effective when used (and often there is some level of policing and enforcement), which in turn means it should be as easy as possible to adopt these standards.

So to recap, some of the main challenges raised are:

- empowering a broader audience (providing as much 'self-service' functionality as possible and lowering barriers to adoption such as lack of training)
- Facilitating easy adoption of best practices

Oracle Warehouse Builder 10g Release 2 enables users to deliver on the knowledge requirements of their organization by encapsulating expertise. This is done in the form of a feature called Experts – empowering broader audiences to use OWB functionality (without necessarily knowing that they are) to accomplish particular task flows, and also to help implement/enforce standards and best practices.

WHAT IS AN EXPERT?

Experts enable you to capture your knowledge (expertise) and best practices in Warehouse Builder and make them repeatable. Experts can be considered to be mini-applications or task-flows that developers build to carry out specific tasks e.g. load data. They are declaratively built using a drag-and-drop editor (just like all other functionality inside OWB) and have a unique capability in the industry: experts allow users to re-use the graphical UI components and capabilities of OWB itself within their experts. For example: OWB's flat file sampler wizard may be re-used as a component within an Expert, where the developer needs the (end) user to define the structure of a file. However in this case, the developer has **full control** over what the information/metadata the wizard returns is used for, and can call other components, write their own scripts, call SQL or Java, etc. etc.

At runtime, Experts follow the same paradigm as traditional OWB functionality, in that code (in this case Tcl with OWB extensions) is generated from the model/logical design, and is then executed.

Building Experts

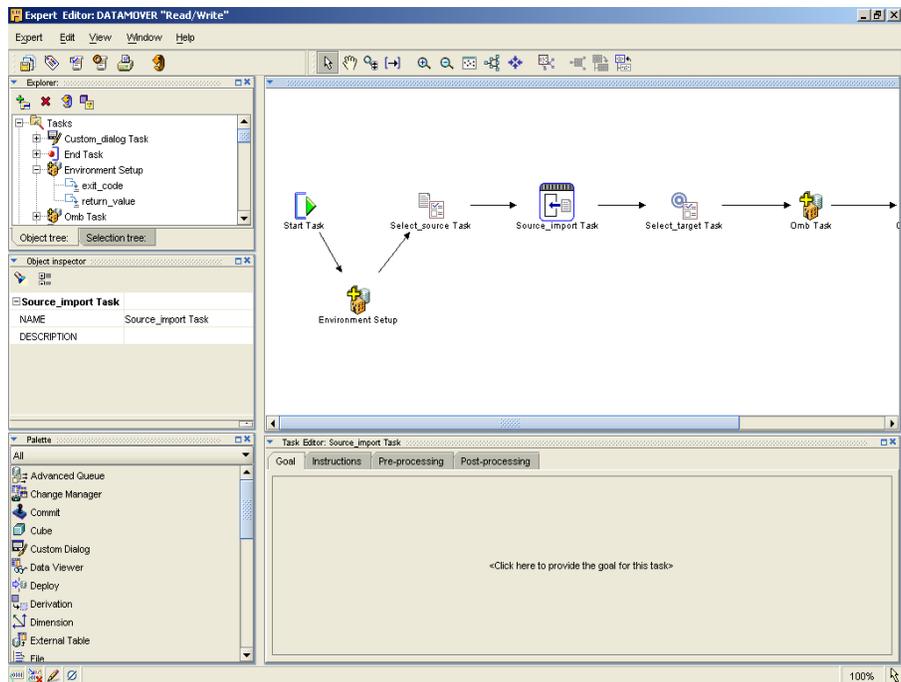


Figure 2 Expert Editor

An Expert in Warehouse Builder is implemented by a sequence of tasks. After creating a module (folder) to store the expert in, the developer is taken into the Expert editor where they drag and drop tasks from the task palette (Figure 2 – bottom left) onto the canvas (Figure 2 – top right).

Tasks are components of OWB and each task has parameters (Figure 2 – top left) that are either input parameters (so help define the behavior of the task) or output parameters (that supply information obtained during the course of executing that task).

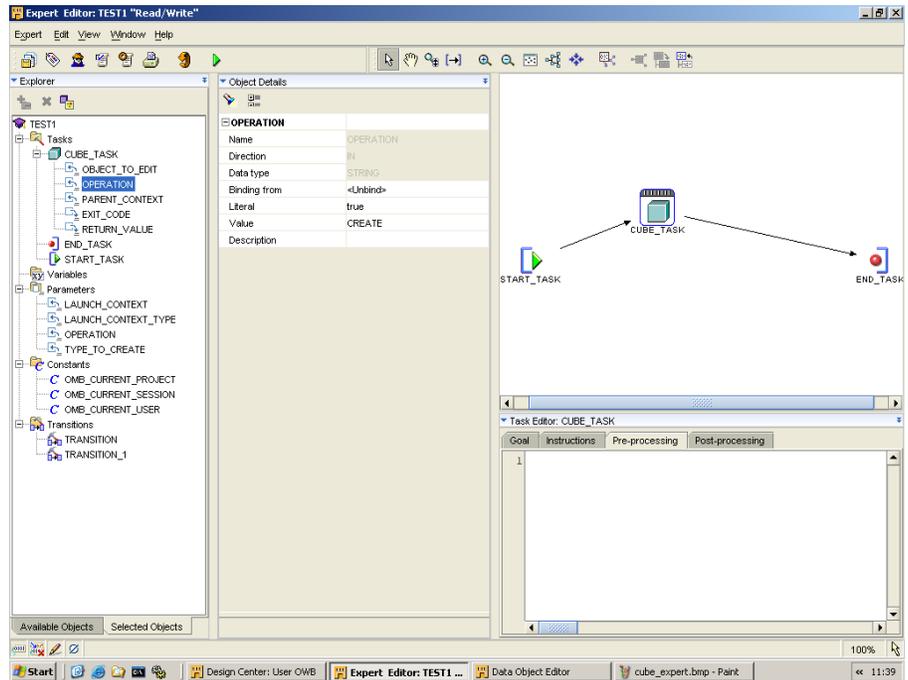


Figure 3 Cube task within an Expert

Figure 3 above shows the Cube task placed on the canvas. Note: Figure 2 and 3 both show the Expert editor (but with different Experts open). The difference in look and feel between them is explained by a new feature of OWB 10gR2: all editors contain dockable/floating panels, permitting developers to re-arrange (or hide) panels to organize the screen layout as they wish.

A closer examination of the Explorer window (Figure 3 - far left) shows the task-specific parameters available to control the behavior of this task:

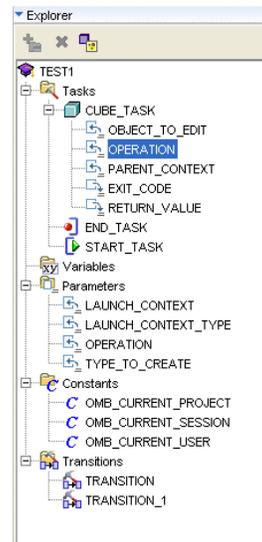


Figure 4 Expert Explorer

The task ‘Cube’ **either** brings up the Cube creation wizard if the argument supplied to the ‘OPERATION’ parameter is ‘Create’ **or** the data object editor if the argument supplied is ‘Edit’ (and the name of the cube to edit is passed to the parameter ‘OBJECT_TO_EDIT’). PARENT_CONTEXT supplies more information needed about where to execute the command (within the Metadata repository), RETURN_VALUE contains the name of the object (if cube creation was used and thus the user was prompted to enter the name) whilst the ERROR_CODE parameter gives the return status of the command (e.g. if the user pressed ‘Cancel’ or some abnormal condition occurred due to a system problem during the course of executing that particular task) allowing developers to handle these cases themselves.

Whilst still examining the Expert explorer, there are some other objects (not task-specific but shown in Figure 4) worthy of note at this point. Variables can be defined in order to store data generated at particular steps in the entire expert (and they can naturally be read also throughout the expert). The Parameters and Constants sections contain special ‘global’ variables (populated by OWB) and are used to communicate (programmatically) to developers how the expert was started (see section below on Running Experts) – allowing developers to change the behavior of their expert accordingly if they so wish.

Going back to each task, In addition to task-specific parameters, each Task has several standard (but optional) dialogs that developers may use (bottom right of Figure 3).

The goal and instruction dialogs contain text that is used at execution time by the Task Assistant (covered in the next section of this paper), to give users of the expert, information and assistance on what they are seeing. An example of a goal might be ‘Define the structure of a Flat File’. The instruction for that same task might be ‘Follow the prompts and click-through each step of the wizard to define

Running Experts

As previously mentioned, Experts follow the same paradigm as traditional OWB functionality in that code (Tcl with OWB extensions) is generated from the logical model the user has designed, and then executed. Those familiar with previous versions of OWB will be familiar with the scripting API (which is based on the Java implementation of Tcl called Jacl) that has OWB-specific extensions, the OMB-prefixed commands ('B' for Batch component) – e.g. OMBCREATE PROJECT 'TEST' will create (in batch) a project (called 'Test'). With OWB 10gR2, new scripting extensions have been added: OMU-prefixed commands ('U' for UI component). For example OMUCREATE PROJECT will show the same UI dialog that is shown when an OWB user clicks the Design → New... menu item and ask the user to specify the project name and description. It is code with both the OMB and OMU extensions that is generated and executed when an Expert is run.

Experts can be executed from within the OWB Design Center itself (select 'Start' off the right-mouse click menu on an Expert) or completely outside the OWB environment e.g. from an icon on the user's desktop (batch file calling OMB*Plus script client and using the OMUSTART EXPERT command) - but of course the OWB software must be physically reside on the user's machine, even if they're not aware they're running OWB.

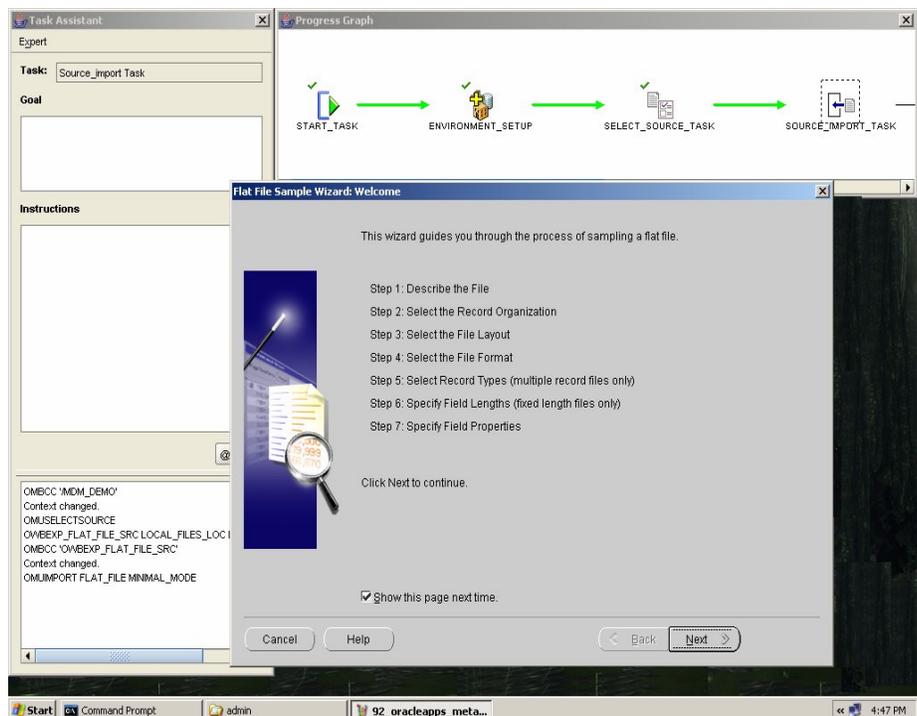


Figure 7 Expert being executed outside the Design Center

When an expert is executed, some additional ('supplementary') UI components/dialogs appear. A progress graph (the top-right dialog in the Figure 7 above) tells the user how far through the expert they are and how much is left to go (a concept sometimes called 'breadcrumbs'). The dialog on the far left of Figure 7 is

the Task Assistant which, for each task in the expert, shows the name of the task, the goal and any instructions (entered during expert development) to help the user understand what is needed to complete the task.

Also present in the Task Assistant (at the very bottom) is a log window that shows the actual Tcl/OMB scripting commands being executed (and any return results from the Tcl interpreter). This can be useful particularly at design-time when building/debugging an expert. By default it is hidden (but is shown in Figure 7).

Preferences control the expert's appearance – in particular whether any or all of the 'supplementary' dialogs mentioned above are shown. Switching off all the supplementary dialogs and executing the expert within the Design Center can make it appear to be a natural extension of the product – and this is key to developers using OWB to adopting best practices – the development environment they use in the course of fulfilling end-user requirements can **itself** be augmented with organization-specific steps encapsulated in an Expert!

Storage of Experts

Experts are regular Warehouse Builder objects and they are persisted as metadata objects in the OWB repository (just like maps, process flows and all other OWB objects). Thus all OWB's advanced metadata management features and facilities such as security, import, export and snapshots/version management apply also to experts.

There are actually two types of Experts: Private Experts are built and saved within a regular OWB Project (Left hand panel in Figure 8). These experts are typically used only in the context of that project. Public experts (Bottom right hand panel in Figure 8) on the other hand are not stored within a specific user's project, but are stored in a special, global project (actually called 'PUBLIC_PROJECT') which is always available to users (in addition to having any specific project open). Public Experts typically encapsulate functionality useful across different types of projects (eg providing some utility functionality such as publishing maps as a web service).

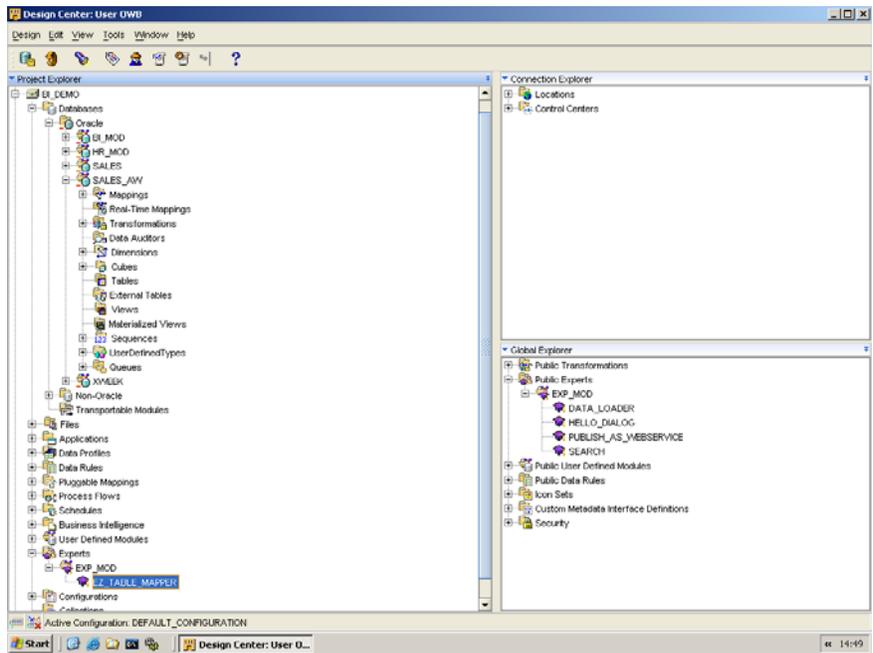


Figure 8 Public and Private Experts

Summary and conclusion

As has been outlined in this paper, Experts are a new and revolutionary technology introduced with Oracle Warehouse Builder 10g Release 2. Major benefits of Experts and the Experts development framework include:

- Improve overall productivity by providing directed guidance for tasks
- Facilitate knowledge management by encapsulating and managing best practices and domain-specific solutions
- Ability to automate parts of a repetitive task
- Simplify the use of Warehouse Builder by exposing only the strictly necessary tasks and encapsulating any perceived complexity
- Declarative approach to code definition
- Run standalone or augment the OWB development environment

Experts are an extremely powerful feature. There are virtually limitless possibilities as to what can be done with them – and all with the goal of improving productivity through encapsulation of expertise and empowering users through guided access. Experts truly vault OWB 10gR2 from being ‘just’ an ETL tool into being a platform and fundamental component of the Oracle Business Intelligence stack, helping users throughout the organization solve their data and metadata integration and quality needs.



Oracle Warehouse Builder 10gR2 Experts
May 2006
Author(s): Paul Narth

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

Worldwide Inquiries:
Phone: +1.650.506.7000
Fax: +1.650.506.7200
www.oracle.com

Copyright © 2006, Oracle. All rights reserved.

This document is provided for information purposes only and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission. Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.