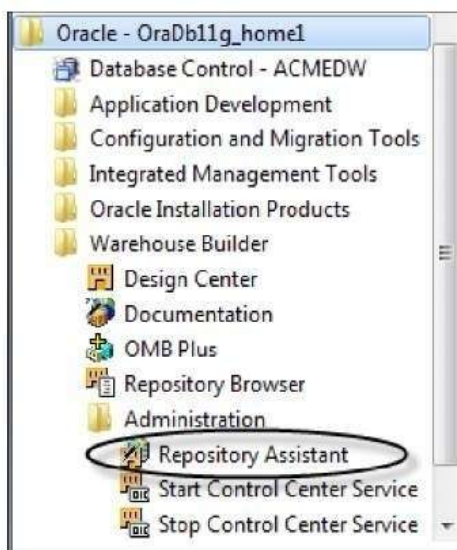


Configuring the repository and workspaces

We have talked about the OWBSYS schema that is created for us automatically during the Oracle 11g installation, and we have also looked at unlocking it and assigning a password to it. However, if we were to connect to the database right now as that user, we would find that as yet only a couple of objects exist in that schema. Filling out that schema is what will be done during this final installation step. We are going to use the Repository Assistant application to configure the repository, create a workspace, and create the objects in the repository that are needed for OWB to run. The OWBSYS schema is where the Warehouse Builder will store those objects. The Repository Assistant application is available from the Start Menu under the Warehouse Builder | Administration submenu of the Oracle program group as shown here:



These menu options will appear locally on a client if we've installed the standalone Warehouse Builder client, as well as on the server. So where should we run the Repository Assistant if we have both? The most common configuration is to run this application on the same machine where the repository is located and the Control Center Service is going to run, which is all on one machine. There are other less common options for where to run the Control Center Service and where the Repository is located in relation to the target schema. These options are documented in Oracle Warehouse Builder Installation and Administration Guide, topic 1 -Overview of Installation and Configuration Architecture.

The URL for the topic in the guide is the following:

http://download.oracle.com/docs/cd/E11882_01/owb.112/e17130/overview.htm#CEGEBHBI.

We want the runtime implemented on the server, which is the most common and simplest configuration. The Repository Assistant pops up an extra screen if it is running remotely from the client, which we will see next. We would see it during the installation if we were on a remote computer.

The steps for configuration are as follows:

1. We'll launch the Repository Assistant application on the server (the only machine we've installed it on) and the first step it is going to ask us for is the database connection information—Host Name, Port Number, and Oracle Service Name — or a Net Service Name for a SQL*Net connection. SQL*Net is Oracle's networking capability for communicating with databases in a distributed networked environment. A naming method is configured so that when using a Net Service name, SQL*Net will know what connection information to use for the connection. We have not configured a naming method, since we don't really need it just to connect locally, so we'll use the Host Name, Port Number, and Oracle Service name option as follows:

° The Host Name is the name assigned to the computer on which we've installed the database, and we can just leave it at LOCALHOST since we're running it on the computer that has the database installed.

° The Port Number is the one we assigned to the listener back when we installed it. It defaults to the standard 1521. This is an example of why the issue of changing or not changing that default port number was mentioned. If we changed it but can't remember what we changed it to, then the following tip will help out.

Determining what port your listener is listening on

There are a couple of options we have for this. One is to perform the following steps.

Open a command prompt window and type in the following command: C:\>lsnrctl

This will launch the Listener Control program, which is the command line utility Oracle provides for controlling the listener. Then enter the following command at the listener control prompt:

```
LSNRCTL> status
```

Look for the line that says:

Listening Endpoints Summary...

The next line will have the port number listed along with the protocol and host name such as the following:

```
(DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=computer) (PORT=1521)))
```

We can find information about the second option for determining the port number in the listener configuration file, listener.ora, in the Oracle home NETWORK\ADMIN directory. Open that file with Notepad and look for the above line.

° **For the Service Name**, we will enter the name we assigned to our database during step 3 of the database creation process. The name we used is ACMEDW. At the end of the database configuration assistant process, a detail screen was displayed. It was suggested that it would be a good thing to take a screen capture of it because it contained details about the database configuration, which would be useful later. One of the items on that screen was the database name that was assigned. If that is not available, then here's another tip to find the database name.

Finding your database instance name

There are a number of places where the database name appears on the database server without us having to log in to the database. One is in the listener control program. Open a command prompt window and type in the following command: C:\>lsnrctl

This will launch the Listener Control program. Then enter the following command at the listener control prompt:

```
LSNRCTL> service
```

Look for the instance name in the list of services that appears.

Another option is to check the name of the Windows service that is started for the database. The database service name is a part of that name. Open Control Panel | Administrative Tools | Services. The Windows service names for the Oracle processes all start with Oracle. The service that runs the actual database is named OracleService<dbname>, where <dbname> is the name of the database instance that you are looking for. The name says OracleServiceACMEDW for a database name of ACMEDW. We can also check the Oracle base folder, which is the folder where the Oracle software was installed. The Admin folder contains a folder named for the database instance if we followed the default naming conventions for folder names during the installation. That is one reason to stick with the OFA standard when installing Oracle products.

2. Now that we've determined the connection information for our database, we'll move along to step 2 of Repository Assistant. It asks us what option we'd like to perform of the following:

- ° Manage Warehouse Builder workspaces

- ° Manage Warehouse Builder workspace users

- ° Add display languages to repository

- ° Upgrade Repository to current release of Warehouse Builder

- ° Manage J2EE User account ° Manage Optional Features

We're going to select the first option to manage workspaces and move along to the next step.

3. This step asks us what we'd like to do with workspaces: create a new workspace or drop an existing one. We'll select the first option to create a new workspace.

4. This brings us to step 4 of the process, which is to specify an owner for the workspace. We are presented with two options: to create a new user or to use an existing user as the owner. To perform the first option, we will need to specify a database user who has DBA privileges that are required to be able to create a new user in the database. The second option is to specify an existing database user to become the owner of the workspace. This user must have the OWB_USER role assigned to be able to successfully designate it as a workspace owner. That is a database role required of any user who is to use the Warehouse Builder. If the existing user who is selected does not have that role, then it must be assigned to the user. An additional step will be required to specify another user who has the ability to do that assignment (grant that roll) or has DBA privileges. This second user must have the Admin Option specified for the OWB_USER role to be able to grant it if he or she does not have DBA privileges.

The user specified here, whether new or existing, will become a deployment target for the Warehouse Builder. This means that the user will be able to access the Design Center for building the ETL processes and the Control Center Manager for deploying and auditing. We'll specify a new user for the ACME Toys and Gizmo's warehouse, since we've just installed this database and no other users are created yet.

5. This step will depend on which option we specified in step 4. If we are creating a new user, it will ask us for an existing user with DBA privileges in the database. The SYSTEM account is the default provided there, but if we have a different account that is a DBA in the database, we can use that. If we have specified an existing user in step 4, then step 5 will ask us for the username and password for that user, as well as the name of the new workspace to create.

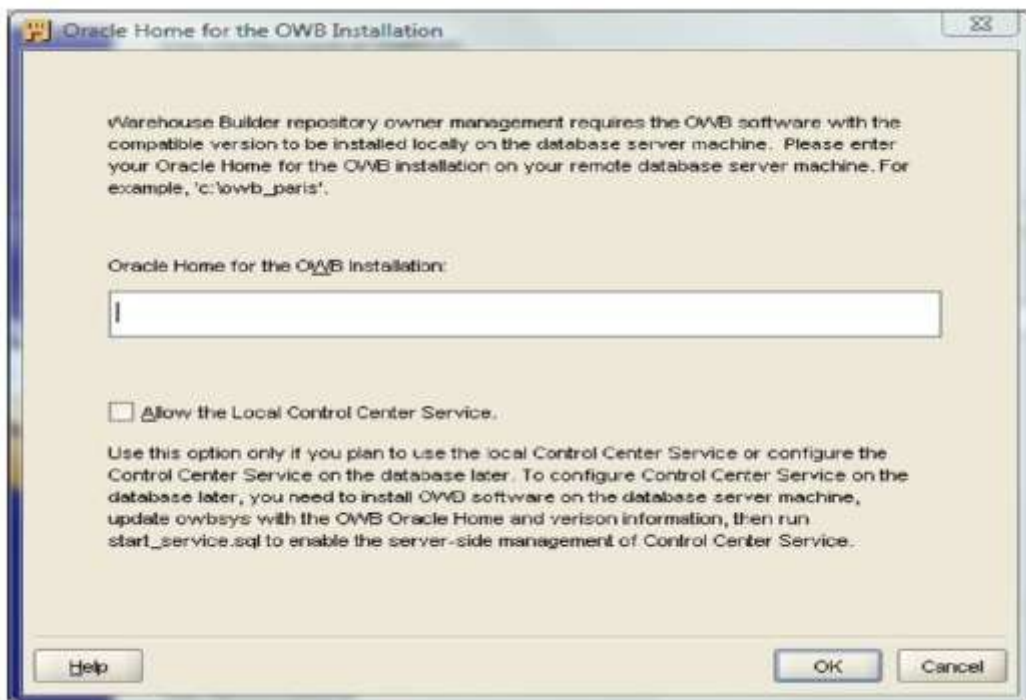
Since we're specifying a new user, we will put in the password for the system user and proceed to the next step. The password used here is the one we previously defined for the system accounts when we created our database.

6. Step 6 is new with release 11gR2 of the Warehouse Builder. It is for selecting optional features of OWB. For the purposes of this topic, only the first option is required, the Data Integrator Enterprise Edition, since it includes the Code Templates that we'll be discussing in topic 10. This feature was named Warehouse Builder Enterprise ETL in previous releases. We'll uncheck the rest. These are separately licensed options also, and we now have the ability to enable or restrict them for users depending on whether we've licensed them or not.

7. In this step, we specify the new username, password, and workspace name. We'll use acmeowb for the username and acme_ws for the workspace name.

8. This step will ask for the password for the OWBSYS user. This schema was installed for OWB to use for the repository. The password it's looking for is the one we set up back on the final database configuration screen at the end of running Database Configuration Assistant to configure the database. This step will only be required upon first running Repository Assistant to create a new workspace since it also has to perform the process of initializing the repository in the OWBSYS schema first. That is a one-time process, which is why subsequent runs of Repository Assistant to manage workspaces will not require this step.

After putting in that password, if we were running the Repository Assistant on a different machine than the database was installed on, then we would encounter the following screen. We referred to it earlier when talking about running the Repository Assistant remotely.

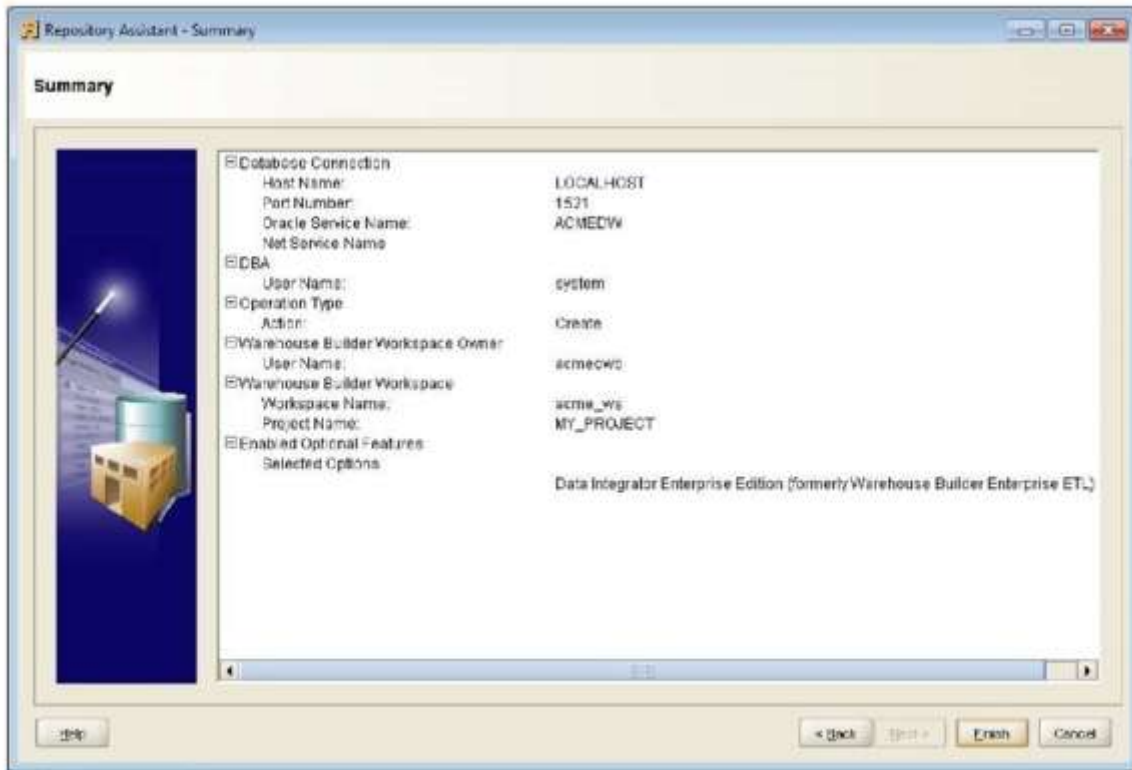


It doesn't know the location of the Oracle Home on the server, and so must prompt for it. It also provides the option for a Local Control Center Service that is for the remote runtime option discussed in the installation guide. Since we're running our database on the same machine as our client, we won't see this screen.

9. This step asks for tablespace names for the OWBSYS schema. A tablespace is a logical entity in an Oracle database for storing data. All objects created are assigned to a tablespace, which stores the data physically in a datafile or datafiles assigned to the tablespace. The administration of tablespaces in an Oracle database is more than we have room for here, so we won't be creating any new tablespaces to hold the OWBSYS data. We'll just leave the defaults selected—the USER tablespace for data indexes and snapshots, and the TEMP tablespace for temporary data. For advanced production databases, it would be a good idea (at a minimum) to specify a separate tablespace for OWBSYS, and actually think about using three new tablespaces for those three that have the USER tablespace assigned.

10. This step is to select a base language for the repository, so we'll make the appropriate selection. Once the repository is created, we cannot change the base language and there can only be one base language assigned to the repository. Physical names of repository objects are assumed to be in the base language. The Repository Assistant will automatically assign the base language depending on the locale that is assigned to the computer we're installing on. We also have the option of selecting one or more display languages that will allow users to assign a business name to physical objects in their own language. Unlike the base language, we can assign display languages after the repository is created. Select any of those that apply.

11. We're almost finished. The final step is the optional step 10 to specify any workspace users from existing database users. We specified the workspace owner as a new user earlier in the install process, and now it's asking for any additional users who we might want to have access to the workspace. The workspace owner is allowed to add and remove database users from the workspace. Removing a database user from the workspace does not delete that user account from Oracle. It only removes him or her as a valid user of the workspace. After selecting any user, the Repository Assistant will present us with a summary screen of the actions it will take and the information we entered, as shown in the following image:



Notice the name of the project toward the end. There was no option to specify that project name, so it's just using a default name. It always sets up a default project in a new workspace by that name, but we can change it later when we actually start designing our data warehouse and working with the workspace in the Design Center.

Click on the Finish button and it will begin the installation, presenting us with a scroll bar moving to the right as it progresses through the installation. The very first time it runs, it will take around 5 to 10 minutes to run before reporting the success pop up, as it has to initialize the repository in the OWB-SYS schema. Creating new workspaces after the first time will be very quick, taking no more than a few seconds to complete.

Summary

That's it. We've gone through the install process of the Oracle 11g database. It automatically installs the Warehouse Builder components as well as the OWBSYS database user. We've also gone through a standalone installation of the OWB client on a separate workstation and have run Repository Assistant to configure our first workspace. We've also discussed the architecture of the Warehouse Builder components as they are now installed on our system. OWB is now installed and ready to use, so we can begin our project of designing and installing a data warehouse.

The general process we're going to follow throughout the rest of the topic to actually build our data warehouse is to start by defining our data sources—where we will import the data from. We will import or define definitions of those sources, so that the Warehouse Builder knows about them. Then we will define our target data structures—where we will be loading data into during ETL and validate those structures. They will have to be generated and deployed to the target schema, which is the process of building the target. After that comes the process of designing and implementing our ETL to load the target from the sources.

Now that we have the software and database loaded, it's time to begin defining our sources of data.