PeopleSoft®

EnterpriseOne Xe Configurable Network Computing Implementation PeopleBook

September 2000

J.D. Edwards World Source Company
7601 Technology Way
Denver, CO 80237

SKU XeEACI

Table of Contents

Configurable Network Computing Implementation

The *Configuration Planning and Setup* suite is designed for use by Configurable Networking Computing (CNC) specialists, OneWorld system administrators, and network/server administrators. The assumption throughout these guides is that the initial OneWorld installation is complete and the standard data sources, path codes, and environments are defined. These guides tell you how to make changes or additions to the configuration setup after the initial installation.

The *Configuration Planning and Setup* suite consists of the following guides:

- *Configurable Network Computing Implementation Guide*. This guide is written primarily for CNC specialists and contains the following topics:
    - Understanding middleware
    - Understanding data sources and verifying that the necessary ones have been created
    - Understanding and creating path codes and environments
    - Working with the Object Configuration Manager
    - Understanding the different modes of processing
    - Understanding a typical OneWorld customer configuration
- *System Administration Guide*. This guide is written primarily for OneWorld system administrators and contains the following topics:
    - Understanding and setting up data replication
    - Setting up printers
    - Using the Work with Servers program
    - Setting up user profiles
    - Setting up OneWorld security
    - Understanding and working with data dictionary administration
    - Understanding vocabulary overrides
    - Understanding transaction processing
    - Working with media objects and imaging
    - Using the universal table browser
    - Understanding OneWorld naming conventions
    - Understanding the jde.ini file

- *Package Management Guide*. This guide is written primarily for OneWorld system administrators and others who manage custom modifications to the OneWorld environments. *Package Management* contains the following topics:
    - Package management planning and setup
    - OneWorld modification rules
    - Object management
    - Package Build
    - Deployment
    - Multitier deployment
- *Server and Workstation Administration Guide*. This guide is written primarily for network administrators and contains the following topics:
    - Understanding Snapshot (multiclient installer)
    - Server administration
    - Troubleshooting the workstation
    - Troubleshooting the server

Although every attempt has been made to organize the information in the *Configuration Planning and Setup* guides according to related tasks, you may find that the information needed to perform the various duties for a position is described in more than one guide. For example, the person who is responsible for setting up path codes, environments, and data sources (described in the *Configurable Network Computing Implementation Guide*) might also be responsible for building and deploying packages (described in the *Package Management Guide*).

The *Configuration Planning and Setup* suite is the central location for all CNC-related tasks except:

- Initial installation of OneWorld. See the *OneWorld Installation Guide.*
- OneWorld upgrade and cumulative updates. See the *OneWorld Upgrade Guide.*
- Network infrastructure and third-party software setup and maintenance. This information is provided by the applicable software or hardware vendor. J.D. Edwards does not provide documentation.

You do not need a complete understanding of the installation process to perform configuration planning and setup tasks. However, to use the *Configuration Planning and Setup* guides it is important that you understand what the installation accomplishes.

Understanding OneWorld Roles

The OneWorld implementation methodology defines specific roles:

- CNC consultant and CNC administrator
- Custom solution consultant and application developer
- Application consultants and application project leaders
- Hardware, network, and third-party software consultants and administrators

Each of these roles is performed by a consultant and a customer. After implementation, the role of the consultant is diminished. Therefore it is critical that the customer ensures that adequate training occurs for each of the roles to be assumed by their personnel.

CNC Consultant and CNC Administrator

The CNC consultant and CNC administrator are involved with installing OneWorld and setting up environments, users, security, distributed processing, and data replication. They are also responsible for setting up version control and testing various CNC configurations. The CNC consultant and CNC administrator control the deployment of OneWorld software throughout the company.

Custom Solution Consultant and Application Developers

OneWorld custom solution consultants resolve business issues by developing applications. Their primary responsibilities include designing the modifications with upgrades in mind, and developing, testing, and introducing the customized software. While the CNC administrator performs the version control functions that build and deploy software, the customer solution consultant must help develop the internal procedures for application development cycle for your business.

Application Consultants and Application Project Leaders

After OneWorld is installed, configured, and rolled out, the application consultants continue in their role as product experts. Although application consultants do not implement the CNC configurations, they must understand how OneWorld handles distributed processing, data replication, environments, and so on, because these application issues influence the CNC decisions. In addition, application consultants must become very good at troubleshooting potential problems.

Hardware, Network, and Third-Party Software Consultants and Administrators

Implementing OneWorld includes many tasks that are outside the scope of J.D. Edwards services. Third-party consultants provide these services as well as supplementing our staff as CNC consultants, network architects, custom modification consultants, and so on.

Understanding the *Configurable Network Computing Implementation* Guide

The *Configurable Network Computing Implementation* guide focuses primarily on how to set up your enterprise after you have loaded OneWorld. Although it is aimed primarily at CNC specialists, those with other job functions may find the information useful or essential to their positions as well.

This guide includes the following sections:

- ❑ Path Code Setup
- ❑ Object Configuration Manager
- ❑ Environment Setup
- ❑ Data Sources
- ❑ Middleware
- ❑ Modes of Processing
- ❑ Typical Customer Configuration

Path Code Setup

Path codes keep track of sets of objects and their locations in OneWorld. For every set of objects in your configuration, OneWorld requires a path code definition in the Object Path Master (F00942) table.

This section defines path codes and describes how to set up and maintain path codes in OneWorld.

This section contains the following:

- Understanding how OneWorld uses path codes
- Working with path codes

Understanding How OneWorld Uses Path Codes

A path code is a pointer to a set of objects. For each set of objects in your configuration, you must define a path code in the Object Path Master table. For example, J.D. Edwards recommends a separate path code definition for each of the following sets of objects:

- J.D. Edwards pristine objects
- Production objects
- Development objects
- Conference Room Pilot (CRP) objects

The Object Path Master (P980042) application uses the Object Path Master (F00942) table.

To properly configure OneWorld you need to understand how OneWorld uses path codes during different phases. This chapter describes the following:

- ❑ Understanding path codes at installation
- ❑ Understanding path codes at runtime
- ❑ Understanding path codes at development

Understanding Path Codes at Installation

OneWorld requires that you define a path code in the Object Path Master for each set of central objects. A set of OneWorld objects consists of a central-objects data source and a directory of objects, which include business function source and include files, object files, and dynamic link libraries (DLLs). A path code definition contains the data source name of the central-object specifications and the directory path to the objects.

When you build a package for the workstation, you must specify a path code. OneWorld uses this path code to determine which set of central objects to use as the source for the package and the directory to use as the destination for the package.

When you deploy objects to an enterprise server or to a workstation, OneWorld uses that path code to determine which set of replicated objects to push to the logic server. J.D. Edwards refers to objects stored on enterprise servers and workstations as sets of replicated objects. OneWorld stores replicated objects in directories using the same name as the path code from which they were deployed. OneWorld executes only replicated objects at runtime. If OneWorld opens an application that does not reside on the workstation and Just-In-Time Installation (JITI) is set for the workstation, the deployment data source will install the needed objects to the workstation at runtime.

Understanding Path Codes at Runtime

OneWorld uses path codes at runtime in the following ways:

- To validate available environments

  When you sign onto OneWorld, it checks what path codes you have defined in your environments against what path code directories are physically installed on the workstation you signed on to. If that workstation does not have a path code that you defined in one of your environments, that environment is not displayed when you sign onto OneWorld.

- To determine the directory location of a requested object

  OneWorld determines the directory on your workstation or server for the replicated objects based on the path code of the environment chosen when the user signs onto OneWorld.

Understanding Path Codes at Development

When you check out an object for development, you use the Object Librarian to specify a path code. OneWorld uses the path code to determine where the central objects are stored and checks out the object from those locations (both database and file server). The path code you choose also determines the directory on your workstation into which OneWorld stores the replicated objects.

When you check in an object, you use the Object Librarian to specify a path code. OneWorld uses the path code to determine the workstation directory from which to pull the replicated objects and the location of the central objects in which to place the objects.

Working with Path Codes

You may want to create path codes to supplement the one you created during installation. The tasks in this chapter describe how to add a path code and create a path code definition. When you add a new path code you must also modify certain Object Librarian tables.

See the following for information about adding a new path code:

- ❑ Adding a new path code
- ❑ Creating a path code definition
- ❑ Modifying tables for the new path code

Adding a New Path Code

During installation, you created one path code (PD7333). You can use the information from this path code to create another one (such as DV7333). When adding a new path code, always do so by copying existing objects, such as from your production path code. You can copy from any existing path code. You cannot add a new path code with an incomplete set of objects.

The following is an overview of the steps necessary to add a new path code. For example purposes, the new path code is DV7333.

▶ To add a new path code

1. Create a new directory on your deployment server for your new path code. For example, create a directory called DV7333.

   Each central-object data source should have an associated directory path on the deployment server. The PD7333 directory on your deployment server contains all of the OneWorld objects (including business function source and include files, object files, and DLLs) and standard packages. Use this directory as a template for creating new central-object directories.

   See *Copy Path Code Objects* in the *OneWorld Installation Guide*.

2. Copy the subdirectories of the PD7333 path code into the new directory (DV7333) that you created.

3. Create a new Oracle or SQL database (or increase the size of your existing Oracle or SQL databases) to hold a set of central-object specifications for the new path code. Remember that the central-object database cannot be on an AS/400 machine.
4. Create a new table owner for the new set of central-object specifications (DV7333).
5. Modify the Object Librarian - Status Details (F9861) and Versions List (F983051) tables for the new DV7333 path code.

   See *Modifying Tables for the New Path Code*.

6. Copy your production path code (PD7333) set of central-object specifications to that database.
7. Verify that the installation process created a new OneWorld data source for the newly copied central-objects database (Central Objects - DV7333). If the process did not create this data source, you must create it.

   See *Adding a Database Data Source*.

8. Using the Object Path Master, modify the DV7333 path code definition. Verify that the Deployment Data Source field specifies your new data source for the newly copied central-objects data source (DV7333). Verify that the Location and Server Share Path fields specify the location of your new DV7333 path code.

   See *Creating a Path Code Definition*.

9. Verify that the package definitions (DV7333_A and DV7333_B) exist that were provided with the installation.

   See *Overview of Creating and Deploying a Package* in the *Package Management Guide*.

10. Copy existing packages from your PD7333 path code to your new DV7333 path code. Modify your new path code package INF files by changing the path code directories to be that of your new path code.

    See *Package Build* in the *Package Management Guide.*

11. Verify that the installation process created the DV7333 environment. If not, create the environment by copying an existing environment. Make sure you change the path code to the new path code.

    See *Adding an Environment*. Remember, you must add new environments to a person's user profile before the environment will be available for selection when he signs onto OneWorld.

12. Modify your Object Configuration Manager mappings for the new environment. For tables mapped to the previous central-objects data source, add records that point the tables to the newly created central-objects data source. Map the Versions List table (F983051) and Processing Option Text table (F98306) to point to the Central Objects - DV7333 data source. You will want the default data source to be Business Data - Test.

    Note: OCM does not determine the location of the F7987* series tables. The deployment data source determines the location of these tables, which includes the Central Objects specifications tables.

    See *Working With the Object Configuration Manager*.

Path Code Naming Conventions

When you add a new path code, be sure to observe the following naming conventions:

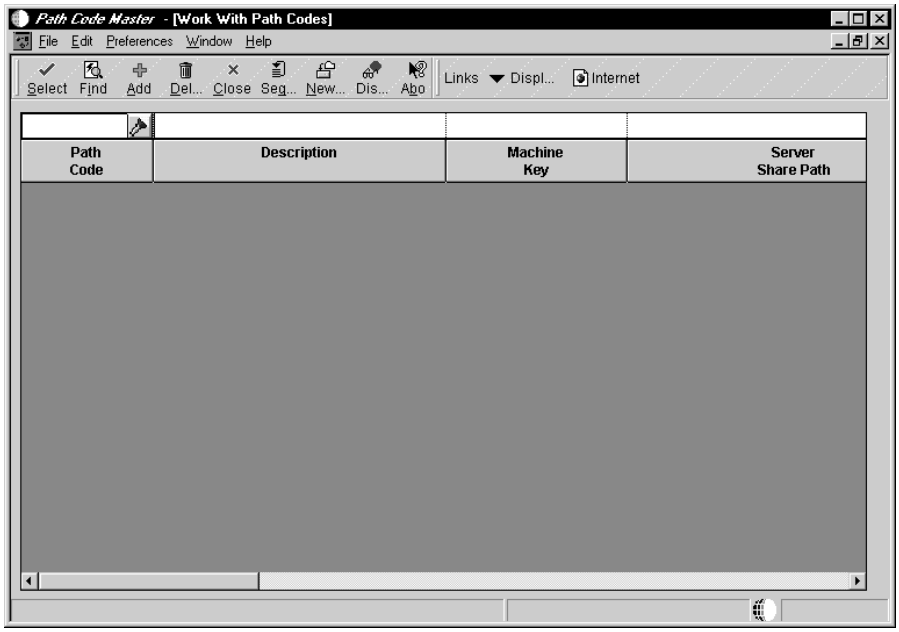
- Limited to 10 characters
- Uppercase only

Creating a Path Code Definition

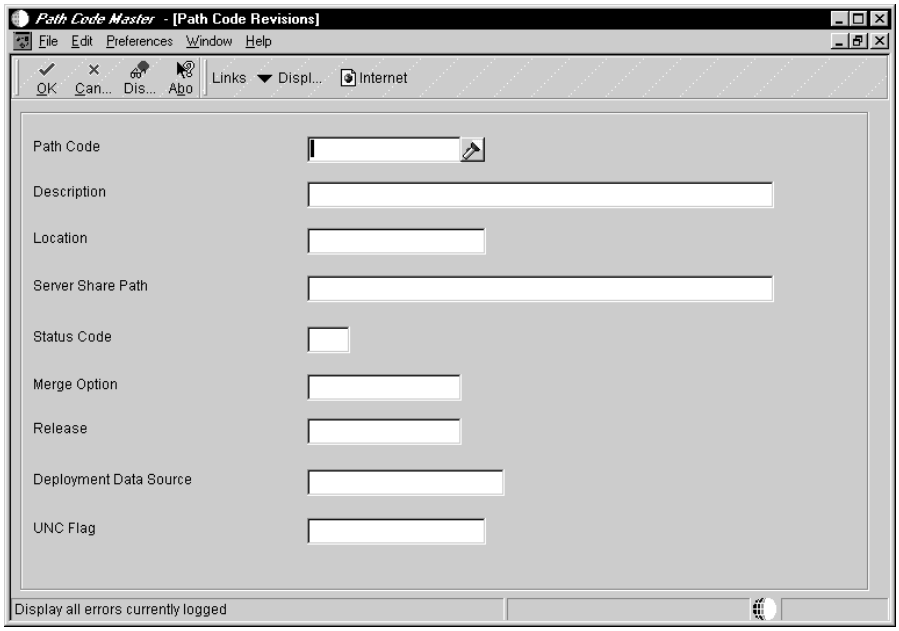
The Object Path Master table (F00942) contains all path code definitions for your configuration. This table resides in your system data source.

▶ To create a path code definition

1. From the Environments menu (GH9053), choose Path Code Master (P980042). The Work with Path Codes form appears.



2. On the Work With Path Codes form, click Add.



3. On the Path Code Revisions form, complete the following fields, and click OK.
   - Path Code
   - Description
   - Location
   - Server Share Path

- Status Code
- Merge Option
- Release
- Deployment Data Source
- UNC Flag

Field	Explanation
Path Code	The Path Code is a pointer to a set of OneWorld objects, and is used to keep track of sets of objects and their locations within OneWorld.
Description	A user defined name or remark.
Location	For World, the Location indicates the machine (server or client). For OneWorld, the Location or Machine Key indicates the name of the machine on the network (server or workstation).
Server Share Path	For World, the Server Share Path Field is used by the environment to determine the location of the current server. For OneWorld, this field indicates the shared directory for this Path Code. The objects that are stored on a file server will be found in this path.
Status Code	This code determines the status of the software in the development cycle.
Merge Option	The Merge Option denotes whether a customer's OneWorld object will be merged in with the J.D. Edwards OneWorld object. The Merge Option can be set at the Path Code level so that all objects checked into that path will carry the same Merge Option as the Path Code.
Release	For World, the release number as defined in the Software Versions Repository. For OneWorld, the release number as defined in the Release Master.

Field	Explanation
Deployment Data Source	OneWorld uses this data source if the primary data source or the data item in the primary data source cannot be located. . . . . . *Form-specific information* . . . . . . This field indicates the location (data source) of the Central Object Specifications data source that corresponds to the path code. For example, if your environment has a PRODxxxx path code, where xxxx is the current OneWorld release level, a valid data source for that path code would be Central Objects - PRODxxxx, where xxxx is the current OneWorld release level.
UNC Flag	Determines how to create the server path. Valid options are: 1 or Y Creates the path using relative paths. Enter a double slash (\\), rather than the specific drive followed by a single slash (\). 0 or N Creates the path using the actual drive letter.

Modifying Tables for the New Path Code

You must modify the Object Librarian - Status Detail table (F9861) and Versions List table (F983051) to reflect the new path code. This allows developers to perform the check-in/check-out process with valid Central Objects information.

▶ To modify the Object Librarian Status Detail table (F9861)

1. Verify that you are logged on to SQL Plus Utility for Oracle or ISQL/w for SQL Server as the user JDE.
2. Run the following SQL command to make a backup of the OBJB733.F9861 table:

   For Oracle:

   ```
   CREATE TABLE F9861SAV AS SELECT * FROM OBJB733.F9861;
   ```

   For ISQL/w:

   ```
   SELECT * INTO F9861SAV FROM OBJB733.F9861
   ```

3. Run the following SQL command to get a record count of OBJB733.F9861:

For Oracle and ISQL/w:

```
SELECT COUNT(*) FROM OBJB733.F9861;
```

4. Run the following SQL command to create a new table with a subset of the master F9861 records:

For Oracle:

```
CREATE TABLE TEMPF9861 AS SELECT * FROM OBJB733.F9861;
WHERE SIMKEY = 'DEPLOYMENTSERVERNAME'
AND SIPATHCD = 'PD7333';
COMMIT;
SELECT COUNT(*) FROM TEMPF9861;
```

For ISQL/w:

```
SELECT * INTO TEMPF9861 FROM OBJB733.F9861
WHERE SIMKEY = 'DEPLOYMENTSERVERNAME'
AND SIPATHCD = 'PD7333'
```

5. Run the following SQL command to modify the temporary F9861 records for the new path code:

For Oracle:

```
UPDATE TEMPF9861
SET SIPATHCD = 'DV7333';
COMMIT;
```

For ISQL/w:

```
UPDATE TEMPF9861
SET SIPATHCD = 'DV7333';
```

6. Run the following SQL command to add the TEMPF9861 records to OBJB733.F9861:

For Oracle:

```
INSERT INTO OBJB733.F9861
AS SELECT * FROM TEMPF9861;
COMMIT;
```

For ISQL/w:

```
INSERT INTO OBJB733.F9861
SELECT * FROM TEMPF9861
```

7. Run the following SQL command to verify the combined total is correct for steps 1 and 2:

   For Oracle and ISQL/w:

```
SELECT COUNT(*) FROM OBJB733.F9861;
```

8. Run the following SQL command to drop the table TEMPF9861:

   For Oracle and ISQL/w:

```
DROP TABLE TEMPF9861;
```

9. Run the following SQL command to drop the table F9861SAV:

   For Oracle and ISQL/w:

```
DROP TABLE F9861SAV;
```

▶ To modify the Versions List (F983051) table

1. Verify that you are logged on to SQL Plus Utility for Oracle or ISQL/w for SQL Server as the user JDE.
2. Run the following SQL command to create a backup of the OBJB733.F9861 table:

   For Oracle:

```
UPDATE DV7333.F983051
SET VRMKEY='DEPLOYMENTSERVERNAME',VRENHV='DV7333';
COMMIT;
```

   For ISQL/w:

```
UPDATE DV7333.F983051
SET VRMKEY='DEPLOYMENTSERVERNAME',VRENHV='DV7333'
```

Object Configuration Manager

OneWorld enables you to distribute data and logic in a manner that optimizes both the power of the workstation and the data integrity of the server. This provides you with:

- Flexibility in determining your own partitioning schemes. As requirements change, you can repartition the system quickly and easily to meet new needs.
- Independence in using the data and logic objects that you need. For example, if you are a salesperson on the road, you can download only the data and logic you need to quote prices and availability. Later, you can connect to your server and place your orders in a batch process.
- Growth for your enterprise systems. Your system can be as large as you need because OneWorld allows you to move objects around the system in practically endless combinations.

To keep track of where data resides and where logic processing occurs, OneWorld uses a tool called the Object Configuration Manager (OCM). OCM enables users to specify data and logic processing locations.

This section contains the following chapters:

- ❑ Understanding the Object Configuration Manager
- ❑ Working with the Object Configuration Manager
- ❑ Running Object Configuration Manager reports
- ❑ Partitioning application logic on servers
- ❑ Business function processing problems

Understanding the Object Configuration Manager

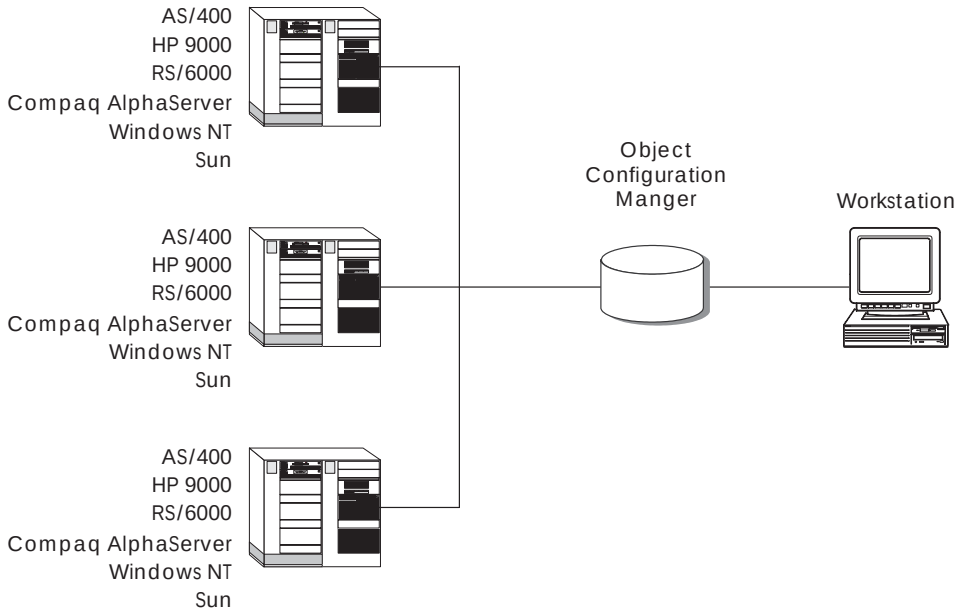
The Object Configuration Manager (P986110) tool configures distributed processing and data dynamically without any programming. Depending on the environment and the user, the Object Configuration Manager points to the correct location for the following:

- Data
- Batch processes
- Business functions

The Object Configuration Manager stores information in tables that tell OneWorld where data resides and where processing occurs. At runtime, OneWorld looks to the Object Configuration Manager to determine these data and processing locations.

It may be helpful to think of the Object Configuration Manager as a policeman directing traffic, or as an orchestra conductor who directs several members to achieve a common goal.

The following graphic illustrates the Object Configuration Manager:



In OneWorld, business data objects (tables) map to database data sources. Batch processes and business functions map to machine data sources.

You always need at least two Object Configuration Manager tables:

- One table for all workstations. Store this table in a centralized system data source. Normally, a central data server stores your system data source. If the central server is unavailable, OneWorld looks to the workstation's jde.ini file for a secondary location.
- One table for each logic server. Servers process differently than workstations. For example, the server map data source for each logic server stores separate Object Configuration Manager tables for server processing.

See Also

- *The Work with Servers Program* in the *System Administration Guide*

Examples of the Object Configuration Manager

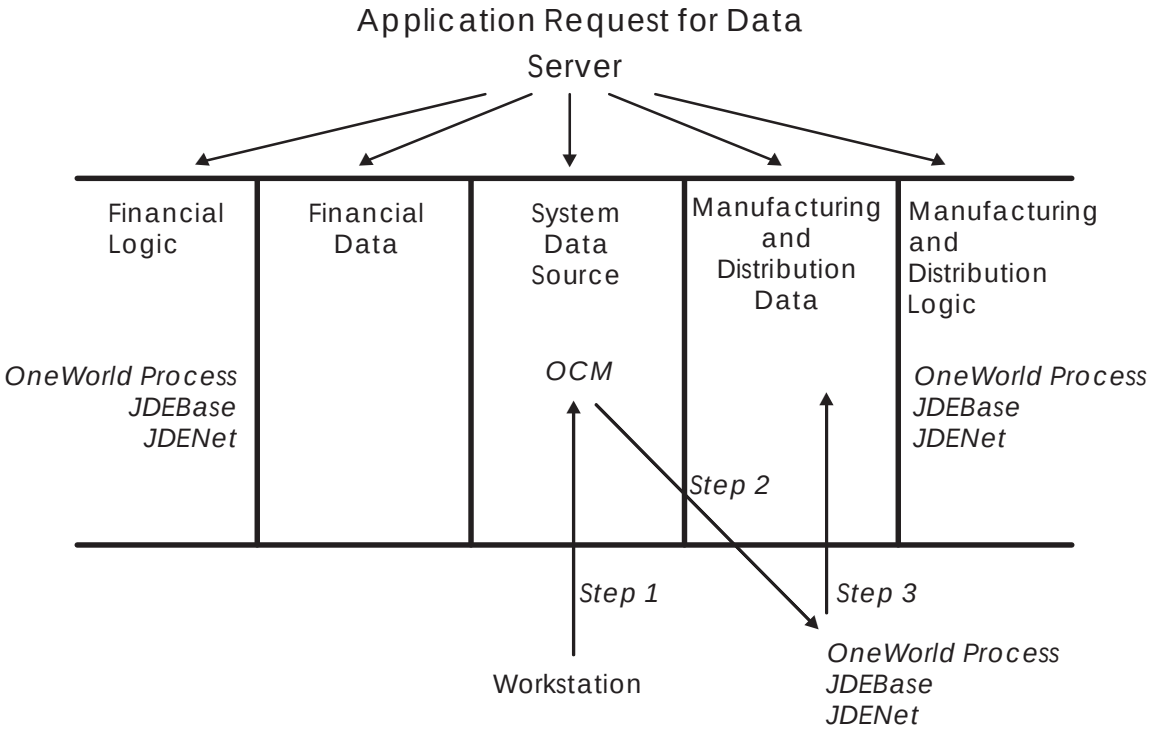
The following are examples of how the Object Configuration Manager works with:

- Application request for data
- Application request for logic

Application Request for Data

The following illustration is an example of how the Object Configuration Manager works with a general data request:

The following steps illustrate how the Object Configuration Manager works with a general data request.



Step 1: Any time an application requests data, OneWorld looks to the Object Configuration Manager table (F986101) for the data source of that table. For example, on the Work With Purchase Orders form, a user enters search criteria in the Query-by-Example line and then clicks Find. Before retrieving the information, OneWorld first has to locate the table in which the information (record) resides. To do that, OneWorld uses the Object Configuration Manager table to determine the correct data source.

The primary, unique index to the Object Configuration Manager table includes:

- Environment, such as PD7333 or DV7333
- User, which is either a specific user ID or group ID, or *PUBLIC
- Object Name, such as F0101, B401002, or R09801
- Database Path

The following table shows the sequential search hierarchy that the Object Configuration Manager uses to locate the correct data source for a data request. For this example:

- The environment is PD7333 (production)
- The status is AV (active)
- The object type is TBLE (table)

Search Sequence	Object Name	User or Group	Search Criteria
1	F0101	SI5745669 (user ID)	Is there a record for the named environment, status active, type TBLE for the named table, and the specific user?
2	F0101	OWTOOL (group ID)	Is there a record for the named environment, status active, type TBLE for the named table, and the specific group?
3	F0101	*PUBLIC	Is there a record for the named environment, status active, type TBLE for the named table, and *PUBLIC?
4	DEFAULT	SI5745669 (user ID)	Is there a record for the named environment, status active, type TBLE with no named table (default), and the specific user?
5	DEFAULT	OWTOOL (group ID)	Is there a record for the named environment, status active, type TBLE with no named table (default), and the specific group?
6	DEFAULT	*PUBLIC	Is there a record for the named environment, status active, type TBLE with no named table (default), and *PUBLIC?

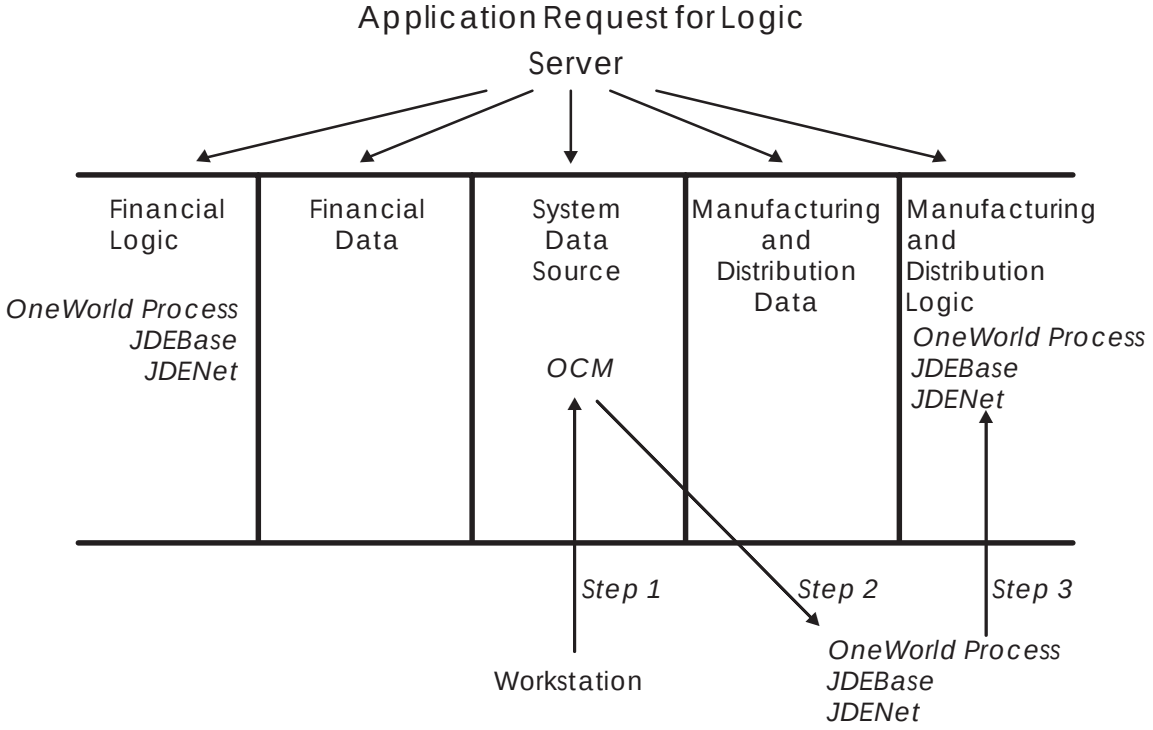
Step 2: Once the data source is determined, OneWorld passes the definition of that data source (that is, database, platform, machine name, connect string, and library) to JDEBase.

Step 3: JDEBase generates the SQL statement that is appropriate for the database being called, and works with third-party communications software to fetch the data and return it to OneWorld.

Note: For the library lists table (F0092), the search hierarchy eliminates step 2, group ID. If OneWorld searches for and cannot find a user ID, it searches *PUBLIC, not the group ID. This is because for the F0092 table the group ID is part of the same record as the user ID, which means that if OneWorld cannot find the user ID it will not find the group ID in that same record

Application Request for Logic

The following illustration is an example of how the Object Configuration Manager works with a general logic request:



Step 1: The request process for application logic is similar to data requests. The Object Configuration Manager controls where all business functions and batch processes are processed.

For example, when you add a purchase order, any event that calls a business function looks to the Object Configuration Manager to determine where to process that business function. After you click OK to complete a purchase order, the OneWorld process calls a master business function to validate all information and record the transaction. OneWorld can process these business functions locally or on the server.

The primary, unique index to the Object Configuration Manager includes:

- Environment, such as PD7333 or DV7333
- User, which is either a specific user/group ID or *PUBLIC
- Object Name, such as F0101, B401002, or R09801
- Database Path

The following table shows the sequential search hierarchy that the Object Configuration Manager uses to locate the correct data source for a logic request. For this example:

- The environment is PD7333 (production)
- The status is AV (active)
- The object type can be either a batch process (UBE) or business function (BSFN)

Search Sequence	Object Name	User or Group	Search Criteria
1	B0900049	SI5745669 (user ID)	Is there a record for the named environment, status active, type (UBE or BSFN) for the named object, and the specific user?
2	B0900049	OWTOOL (group ID)	Is there a record for the named environment, status active, type (UBE or BSFN) for the named object, and the specific group?
3	B0900049	*PUBLIC	Is there a record for the named environment, status active, type (UBE or BSFN) for the named object, and *PUBLIC?
4	DEFAULT	SI5745669 (user ID)	Is there a record for the named environment, status active, type (UBE or BSFN) with no named object (default), and the specific user?

Search Sequence	Object Name	User or Group	Search Criteria
5	DEFAULT	OWTOOL (group ID)	Is there a record for the named environment, status active, type (UBE or BSFN) with no named object (default), and the specific group?
6	DEFAULT	*PUBLIC	Is there a record for the named environment, status active, type (UBE or BSFN) with no named object (default), and *PUBLIC?
7			If there is no record for this object type, then OneWorld processes the process on the workstation.

Step 2: Once the data source is determined, OneWorld passes the definition of that data source to JDENet.

Step 3: JDENet sends a message to the server to begin processing logic. When JDENet on the server receives the message, a OneWorld process on the server responds to the message by processing the requested logic object.

Working with the Object Configuration Manager

The Object Configuration Manager provides the flexibility to map data, batch applications, and business functions to a data source. This allows you to coordinate your OneWorld distributed processing. For example, you would map table objects to database data sources and logic objects to machine data sources.

You must have at least two sets of the Object Configuration Manager and Data Source Master tables:

One for All Workstations The Object Configuration Manager and Data Source Master tables that OneWorld uses for workstation processing are stored in the centralized system data source normally kept on an enterprise server. If the system data source is not available, OneWorld looks to the workstation's jde.ini file for a secondary location.

One Per Logic Server The Object Configuration Manager and Data Source Master tables that the logic server uses are stored on that server in the server map data source. Each logic server type requires its own server map data source.

In OneWorld, business objects are used to configure distributed processing and distributed data at runtime. This configuration is defined by an Object Configuration Master table (F986101). You can work with the server object map to modify the entries in this table.

Workstation mappings are stored in a centralized system data source. The Object Configuration Master table used by the enterprise server is stored on that server in the server map data source. Each enterprise server requires a separate server map data source.

Compared to a workstation, an enterprise server processing a logic object has a different perspective of where data should be retrieved. For example, J.D. Edwards recommends that for workstations you map user defined codes to run locally. When this is the case, a workstation requesting user defined codes has its Object Configuration Master (in the system data source) pointing to the local database.

When an enterprise server is requesting user defined codes, logically it makes no sense for the enterprise server to look to a workstation for this information. Therefore, the enterprise server should have unique mappings for user defined

codes. These mappings are set up in the Object Configuration Master table (F98610) in the server map data source.

If you have changed the Object Configuration Master for the workstation, you should check the Object Configuration Master in the server maps to see if they should also be changed. For example, if you have new environments with unique mappings for the workstation, you should check to see if changes are required in the corresponding mappings for the enterprise server.

The Object Configuration Manager also provides batch processes to help with the administration of your object mappings. These processes perform such tasks as comparing, updating, copying, and deleting Object Configuration Manager records.

The Object Configuration Manager (P986110) application updates the Object Configuration Master (F986101) table.

This chapter contains the following topics:

- Mapping objects
- Object Configuration Manager processing options
- Mapping generic text

Mapping Objects

You map objects by environment. You select an environment you have already created and map that environment's objects to the data sources you want those objects to use. You can set default mappings for all instances of an object type to one data source, and you can map individual objects to data sources.

Mapping Object Types: Default Maps

To create a default map for an object type, create a mapping whose object name is the literal value: DEFAULT. Then enter an object type (such as TBLE) and a data source (such as the replicated local data source). By creating a default map for the object type TBLE, any table objects not mapped individually would point to the default data source.

Mapping Individual Objects

You can map individual objects within an environment. For example, you can map a specific table, such as the Security Workbench Table (F00950), to a data source other than the default, such as to the system data source.

If you do not explicitly map an object by name in the Object Configuration Manager, OneWorld uses the default map for that object's type.

Caution: Each environment must have a default map for TBLE (table) objects for the *PUBLIC user profile because there is no inherent default location for table objects. If table objects do not have a default map and are not explicitly mapped by name, OneWorld produces a Select/Failed error message.

Batch applications and business functions automatically run locally if there is no default map for that object type.

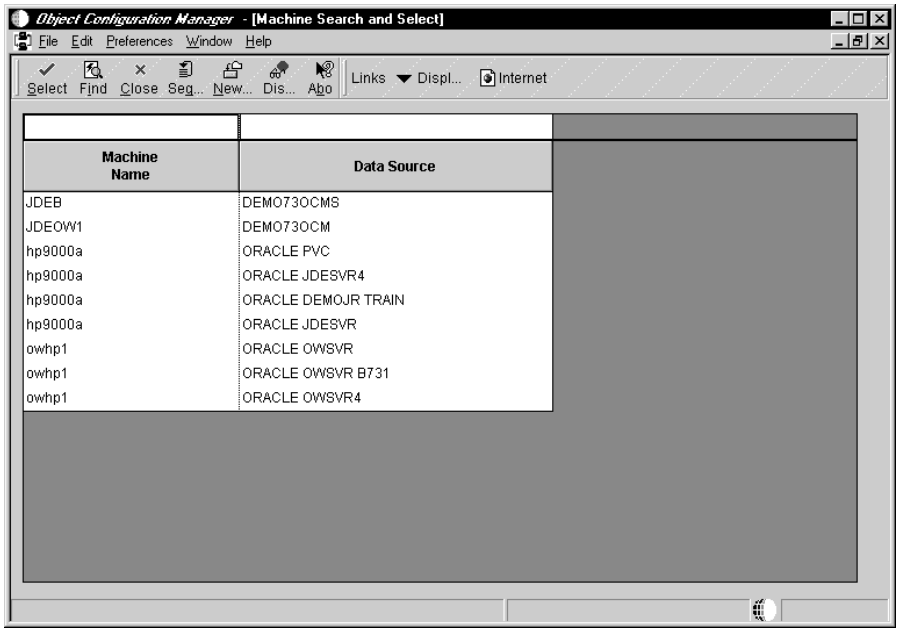
Complete the following tasks:

- Map objects
- Change the mapping for an Object Librarian table
- Update the Oracle parameters table

▶ To map objects

From the System Administration Tools menu (GH9011), choose Object Configuration Manager (P986110).

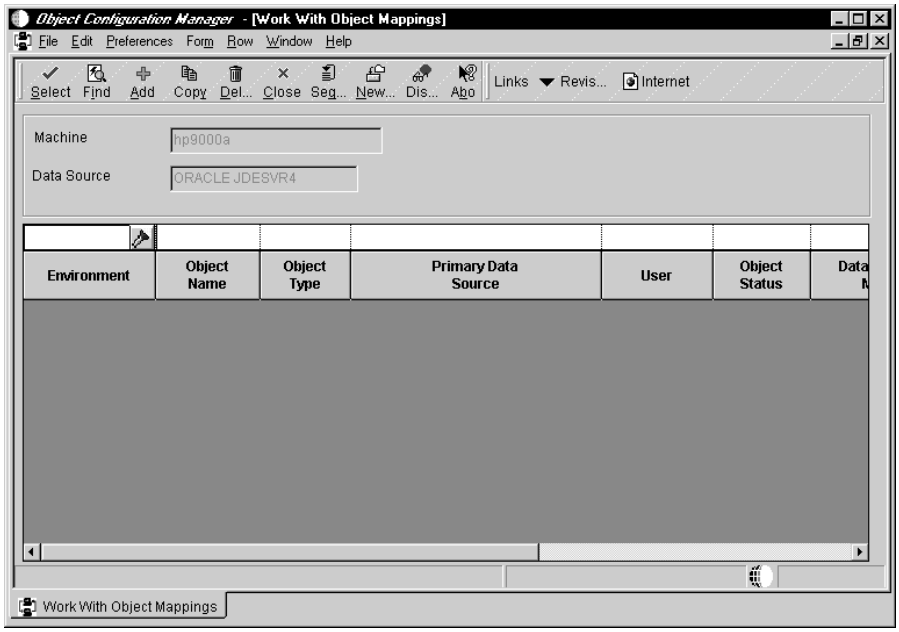
The Machine Search and Select form appears. This form displays data sources that have the OCM Data Source field checked on the Data Source Revisions form.



Choose the data source that stores the Object Configuration Manager table with which you want to work, and click Select.

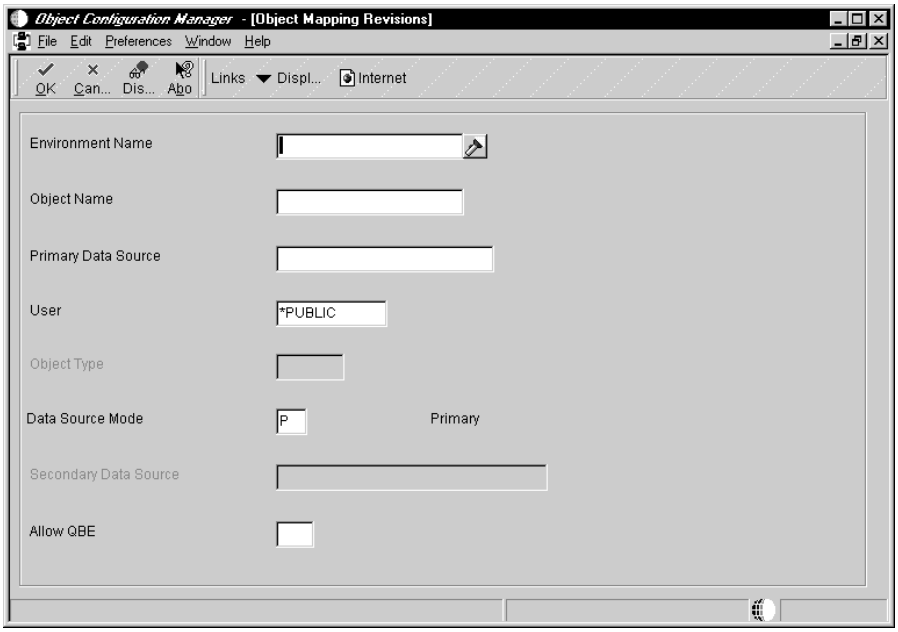
The machine name next to the data source in the grid refers to the machine where the data source resides. Workstations use the system data source for their mappings. Each enterprise server has its own server map data source.

The Work with Object Mappings form appears. On this form you can locate data sources for which to revise object mappings and access a form where you can add new object mappings for data sources.



Click Add.

The Object Mapping Revisions form appears. On this form, you specify the data source to which your objects map.



Complete the following fields:

- Environment Name
- Object Name

  You can define parameters for all objects in a data source by typing DEFAULT in this field.

- Primary Data Source

  The primary data source that you select must be valid for the type of object that you map. For example, you can only map objects of type TBLE and GT to a database data source, and you can only map objects of type BSFN to a logic data source.

- User
- Object Type

  See *Object Configuration Manager Processing Options* for information about validating the TBLE object type.

- Data Source Mode

  You should only use the secondary data source for business function (BSFN) data sources.

- Secondary Data Source
- Allow QBE

Click OK to save your object mapping.

You can continue to map objects, clicking OK after each mapping. Be sure to click OK to save your latest mapping, then click Cancel to exit the form and return to the Work with Object Mappings form.

On the Work with Object Mappings form you can verify that the Object Configuration Manager has created your map with an inactive status; the Object Status field displays NA (inactive).

To make your map active, on the Work With Object Mappings form, choose the map, and from the Row menu, choose Change Status. The Object Status field displays AV (active).

Field	Explanation
Environment Name	For World, the Environment name is also called the Plan Name and is used to uniquely identify an upgrade environment for Install/Reinstall. For OneWorld (Install Applications), the environment name is also called the Plan Name and is used to uniquely identify an upgrade environment for Install/Reinstall. For OneWorld (Environment or Version Applications), this is the path code that identifies the location of the application or version specification data.
Object Name	The OneWorld architecture is object-based. This means that discrete software objects are the building blocks for all applications, and that developers can reuse the objects in multiple applications. Each object is stored in the Object Librarian. Examples of OneWorld objects include: • Batch Applications (such as reports) • Interactive Applications • Business Views • Business Functions • Business Functions Data Structures • Event Rules • Media Object Data Structures . . . . . *Form-specific information* . . . . . . On this form, this is the OneWorld object that you want to map. To create a default map for all of an object type, enter the literal value DEFAULT into this field, then enter an object type into the Object Type field.

Field	Explanation
Data Source	The data source name. . . . . . *Form-specific information* . . . . . . Data sources are the building blocks that you use to set up a OneWorld enterprise configuration. They define all of the databases required (where your tables reside) and all of the logic machines (where OneWorld executes logic objects for your enterprise). If OneWorld cannot find your primary data source or cannot find the data item in your primary data source, it will attempt to connect to your secondary data source.
User Class/Group	A profile used to classify users into groups for security purposes. Some rules for creating a User Class/Group are as follows: • The 'Class/Group' profile must begin with * so that it does not conflict with any System profiles. • The 'User Class/Group' field must be blank when entering a new group profile. . . . . . *Form-specific information* . . . . . . On this form, you can enter an individual user, a group name, or the literal value *PUBLIC.
Object Type	The type of object with which you are working. For example, if you are working with tables the object type is TBLE, or business functions is BSFN. . . . . . *Form-specific information* . . . . . . You can map data (object type TBLE), batch applications (object type UBE), business functions (object type BSFN), or generic text (object type GT).
Data Source Mode	Indicates whether OneWorld uses the primary or secondary data source. Valid values are: P Primary. S Secondary. You should only use this for business function (BSFN) data sources.
Data Source	OneWorld uses this data source if the primary data source or the data item in the primary data source cannot be located.
Allow QBE	Use this flag to turn ON or OFF row-level record locking for the data source. You should have this flag turned ON to help prevent database integrity issues. JDEBASE middleware uses this flag to determine whether or not to use row-level record locking.

To change the mapping for an Object Librarian table

From the System Administration Tools menu (GH9011), choose Object Configuration Manager (P986110).

The Machine Search and Select form appears. This form displays data sources that have the OCM Data Source field checked on the Data Source Revisions form.

Choose the machine and data source for the Object Configuration Manager table you want to work with, and click Select.

The data source resides on the machine. Workstations use the system data source for their mappings. Each enterprise server has its own server map data source.

The Work with Object Mappings form appears. On this form you can add and revise object mappings to data sources.

From the Form menu, choose Revise OL DS.

The Revise OL Data Source form appears. On this form you define the new mapping for the Object Librarian tables.

When you map any of the Object Librarian tables, OneWorld validates your entries to ensure that all environments based on the same path code have their Object Librarian tables mapped to the same data source. OneWorld alerts you with an error message if you map your Object Librarian tables to different data sources.

Complete the following fields:

- Path Code
- OL Data Source

Click OK to save the revision to your data source mapping.

When you click OK, OneWorld automatically updates the Object Librarian data source mapping for each environment that contains the path code you entered in the Path Code field.

Field	Explanation
Path Name - Inquiry	An alphanumeric code that identifies a path, which is a sequence of formats for reviewing supplier performance information.
Data Item - Secondary	Used in the Data Dictionary Cross Reference File to use data item as it appears in the Data Dictionary (without file prefix).

To update the Oracle parameters table

You must update the Oracle parameters table if you use Oracle and you do not follow the standard naming conventions J.D. Edwards recommends, or if you add new Oracle data sources.

From the System Administration Tools menu (GH9011), choose Object Configuration Manager (P986110).

The Machine Search and Select form appears. This form displays data sources that have the OCM Data Source field checked on the Data Source Revisions form.

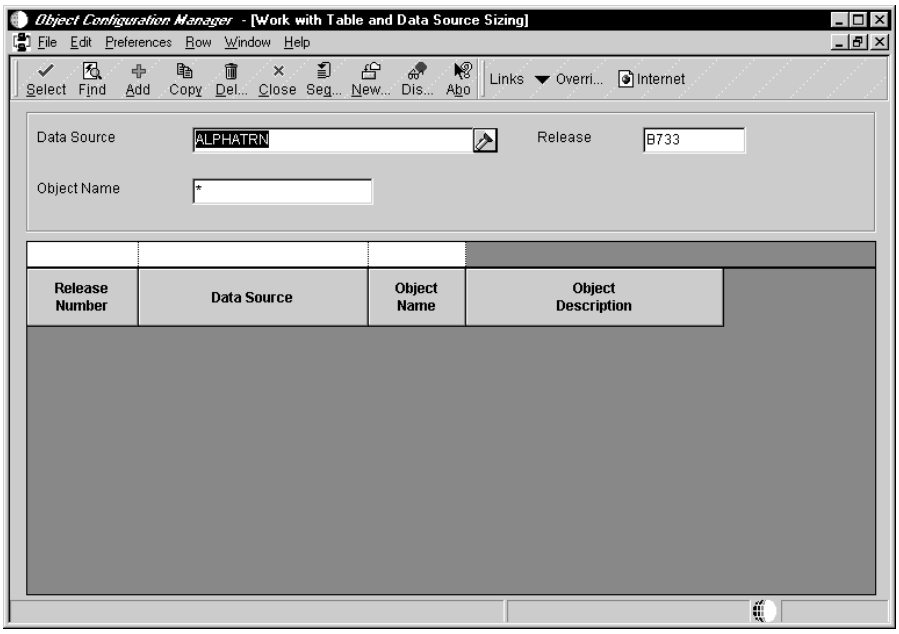
Choose the machine and data source for the Object Configuration Manager table you want to work with, and click Select.

The data source resides on the machine. Workstations use the system data source for their mappings. Each enterprise server has its own server map data source.

The Work with Object Mappings form appears. On this form you can add and revise object mappings to data sources.

From the Row menu, choose Oracle Params.

The Work with Table and Data Source Sizing form appears. On this form you can add or revise the parameters for Oracle data sources or specific tables. You can also access this application from the Data Sources application.



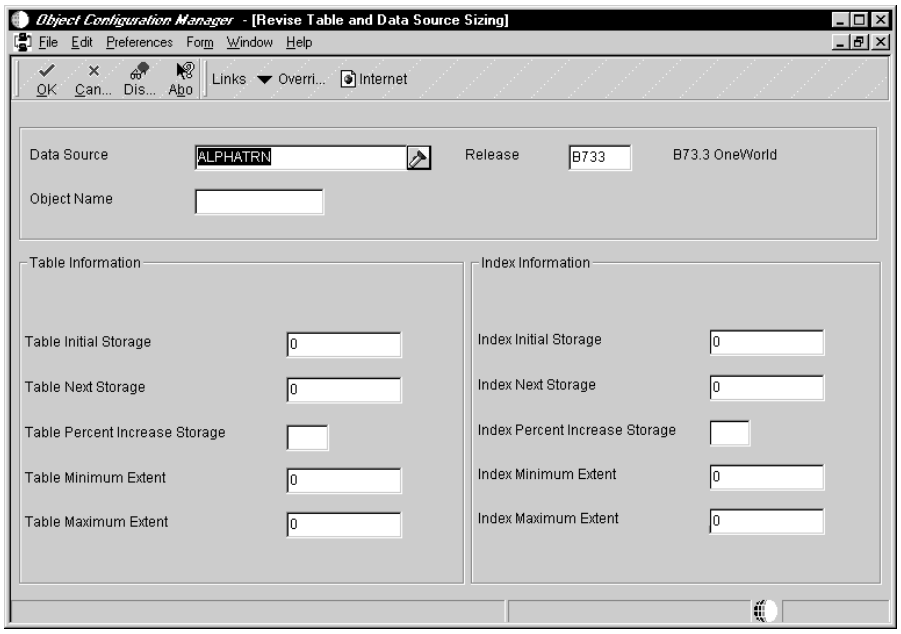
Click Find to locate information with which to work.

You can modify the fields on this form to search by release level, data source, and object name. You do not need to click Find if you are adding new parameters.

Do one of the following:

- Choose a record, and click Select.
- Click Add.

The Revise Table and Data Source Sizing form appears. On this form, you can revise and add table and index information for a data source or table.



Alternatively, you can display the Revise Table and Data Source Sizing form by choosing Default Database from the Work With Data Sources form's Row menu. When you do so, the DEFAULT value is automatically entered at the Object Name field so that default values are used for the form. If you change the object name from DEFAULT, the fields that were formerly filled with default values on the form appear.

If you are adding new parameters, complete the following fields:

- Data Source Name
- Object Name

  You can define parameters for all objects in a data source by typing DEFAULT in this field.

- Release

Depending on whether you select a data source for the DEFAULT map or for a specific object, different fields appear on the Revise Table and Data Source Sizing form. For the DEFAULT map, you can define only the name of the Oracle space where tables or indexes reside. However, for a specific object you can define parameters such as the amount of space to use for an Oracle table.

For the DEFAULT map, complete the following fields:

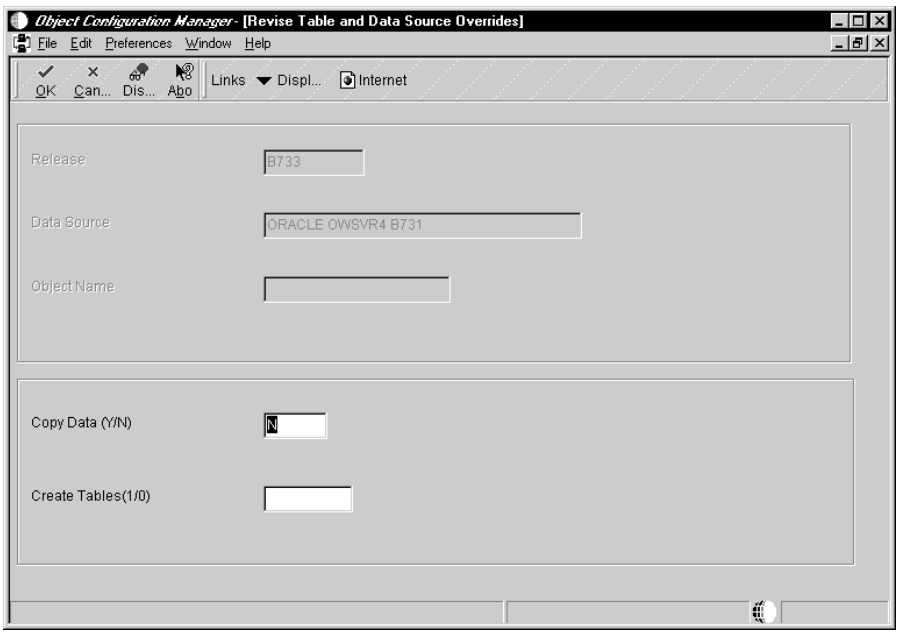
- Table Space Name
- Index Space Name

For a specific object, complete the following fields:

- Initial Storage
- Next Storage
- Percent Increase Storage
- Minimum Extent
- Maximum Extent

These values are valid for both the table information and the index information.

To enter override information, choose Overrides from the Form menu on the Revise Table and Data Source Sizing form. (This option is also available from the Work With Table and Data Source Sizing form's Row menu.) The Revise Table and Data Source Overrides form appears.



On the Revise Table and Data Source Sizing form, complete or modify the following fields:

- Copy Data (Y/N)
- Create Tables (1/0)

Click OK to return to the previous form. If necessary, click OK on all subsequent forms until you return to the System Administration Tools menu.

Click OK to save the information.

Field	Explanation
Data Source	The data source name.
Object Name	The OneWorld architecture is object-based. This means that discrete software objects are the building blocks for all applications, and that developers can reuse the objects in multiple applications. Each object is stored in the Object Librarian. Examples of OneWorld objects include: • Batch Applications (such as reports) • Interactive Applications • Business Views • Business Functions • Business Functions Data Structures • Event Rules • Media Object Data Structures
Release Number	For World, the release number as defined in the Software Versions Repository. For OneWorld, the release number as defined in the Release Master.
Table Space Name	The name of an Oracle region of space created to store tables.
Table Initial Storage	The amount of Oracle space, in bytes, required to store a table in one initial segment of space, or extent.
Table Next Storage	The amount of Oracle space, in bytes, that will be allocated to a table once the previously used segment of space has been filled.
Table Percent Increase Storage	An Oracle parameter that indicates the number of percentage points that the next extent of Oracle space will grow over the last extent of space that was allocated to a table.
Table Minimum Extent	An Oracle parameter indicating the minimum number of space segments, or extents, that Oracle will allocate to a table.
Table Maximum Extent	An Oracle parameter indicating the maximum number of space segments, or extents, that Oracle can allocate to a table.

Field	Explanation
Index Space Name	The name of an Oracle region of space created to store indexes.
Index Initial Storage	The amount of Oracle space, in bytes, required to store an index in one initial segment of space, or extent.
Index Percent Increase Storage	An Oracle parameter that indicates the number of percentage points that the next extent of Oracle space will grow over the last extent of space that was allocated to an index.
Index Minimum Extent	An Oracle parameter indicating the minimum number of space segments, or extents, that Oracle will allocate to an index.
Index Maximum Extent	An Oracle parameter indicating the maximum number of space segments, or extents, that Oracle can allocate to an index.

Object Configuration Manager Processing Options

The Object Configuration Manager application (P986110) has one processing option that controls error handling.

Process

This processing option enables you to specify whether to issue a warning or an error in the event that the TBLE object you are mapping does not exist in the data source to which you are mapping the table.

1. Non-existent table error

Use this processing option to specify whether an error or a warning should be issued when a table does not exist in the data source to which it is mapped. Valid values are:

1 Issue an error.
Blank Issue a warning.

Mapping Generic Text

The Generic Text (F00165) table has more mapping flexibility than other objects. This is because the data stored in this table has different uses. The following are the generic text categories:

- Common data that all environments use. For example, data dictionary glossaries and business function notes are the same across all environments.
- Production data specific to an environment. For example, inventory item notes and address book supplemental data could be different for a corporation running OneWorld over multiple environments such as production, test, and J.D. Edwards demo data.

▶ To map generic text

In the Object Configuration Manager program (P986110), use the GT object type to map specific generic text objects.

The following is an example of what the Object Configuration Manager table (F986101) looks like after installing OneWorld, if your business follows the typical configuration. This example shows only the PD7333 (production) and TS7333 (test) environments.

Environment	Object Name	Description	Data Source	Object Type
PD7333	F00165	Generic Text Table	Business Data - Prod	TBLE
PD7333	GT92002	Data Dictionary Glossary	Data Dictionary	GT
PD7333	GT9860A	Object Librarian	Object Librarian	GT
PD7333	GT9862A	Business Function Notes	Object Librarian	GT
PD7333	GT98DSA	Data Structure Notes-Structure	Object Librarian	GT
PD7333	GT98DSB	Data Structure Notes-Structure and Item	Object Librarian	GT
PD7333	GT98TMPL	Media Objects Templates	Object Librarian	GT
TS7333	F00165	Generic Text Table	Business Data - Test	TBLE
TS7333	GT92002	Data Dictionary Glossary	Data Dictionary	GT
TS7333	GT9860A	Object Librarian	Object Librarian	GT

Environment	Object Name	Description	Data Source	Object Type
TS7333	GT9862A	Business Function Notes	Object Librarian	GT
TS7333	GT98DSA	Data Structure Notes-Structure	Object Librarian	GT
TS7333	GT98DSB	Data Structure Notes-Structure and Item	Object Librarian	GT
TS7333	GT98TMPL	Media Objects Templates	Object Librarian	GT

Running Object Configuration Manager Reports

Running Object Configuration Manager Reports

The Object Configuration Manager provides batch processes that produce a report to help you keep track of your mappings.

Some Object Configuration Manager reports are called when you choose that report from the OneWorld menu. Other OCM reports are called from a report driver job. Although the end result is the same, there are some differences in the way you set processing options and data selection for a batch process called from a report driver.

When a report is called by a driver, processing options are set from the report for all reports that are launched from the driver.

For example, the R988611A driver calls the Mapping Comparisons, Global Update, OCM Delete, and OCM Copy reports.

Any data selection for a report must be entered for the specific report, not from the driver. When you finish entering processing options and data selection, if any, you can run the report from the driver.

This chapter contains the following topics:

- ❑ Setting processing options for OCM reports
- ❑ Processing options: OCM Mapping Comparison, Global Update, Delete, and Copy
- ❑ Running the Object Configuration Mapping Comparison report
- ❑ Running the Object Configuration Global Update report
- ❑ Running the Object Configuration Delete report
- ❑ Running the Object Configuration Copy report
- ❑ Running the Verify Object Configuration Manager report
- ❑ Processing Options: Verify Object Configuration Manager (R9861130)
- ❑ Running the Object Configuration System Table Update report

- ❑ Processing Options: System Table Update (R986101A)
- ❑ Running the Create OCM Records for Business Functions process report
- ❑ Running the Job Master Deletion By Days Old report
- ❑ Processing Options: Job Master Deletion By Days Old
- ❑ Running the OCM Category Update/Delete report

Setting Processing Options for OCM Reports

The following task describes how to use a report driver to set processing options for a data source report. The following OCM reports are called by a report driver:

- Object Configuration Mapping Comparison
- Object Configuration Global Update
- Object Configuration Delete
- Object Configuration Copy
- Job Master Deletion By Days Old

All of the reports except Job Master Deletion By Days Old are called from the R98611A report driver. Job Master Deletion By Days Old is called from the R9861101 report driver. Except as noted, the process for setting processing options for these reports is identical, regardless of which report you select.

▶ Setting processing options for OCM reports

From the Advanced Operations menu (GH9012), choose one of the OCM reports that are called by a report driver. The Work with Batch Versions - Available Versions form appears. (The same form appears regardless of which report you choose.)

Select one of the reports (it doesn't matter which one) and then choose Processing Options from the Row menu. The Processing Options form appears.

If you selected the Job Master Deletion By Days Old report, enter the fields on the Defaults and Versions tabs. For more information about entering the fields on this tab, see *Processing Options: Job Master Deletion By Days Old Report (R9861102*. When you are finished, skip to the last step.

If you selected any other OCM report besides Job Master Deletion By Days Old, on the form's Driver tab, enter information for the report driver. For

more information about entering the fields on this tab, see *Processing Options: OCM Mapping Comparison, Global Update, Delete, and Copy*.

Click the tab for the report whose processing options you want to enter. For more information about entering the fields on these tabs, see *Processing Options: OCM Mapping Comparison, Global Update, Delete, and Copy*.

Click OK to save and exit to the Work with Batch Versions - Available Versions form.

Processing Options: OCM Mapping Comparison, Global Update, Delete, and Copy

The following batch applications have processing options that are entered from the R98611A report driver using the same processing option form:

- Object Configuration Mapping Comparison (R986101)
- Object Configuration Global Update (R986110)
- Object Configuration Delete (R986120)
- Object Configuration Copy (R986121)

The processing options for OCM batch applications let you specify the data source used when comparing, updating, deleting or copying data. These processing options let you set other parameters used by the specific batch application.

Only the processing options for the report you are currently running are used. Processing option fields for other reports are ignored until you run that report.

Driver Tab

This processing option lets you specify the Object Configuration Manager report you want to run. Verify that the displayed report number corresponds to the report you want to run. You can also specify the version you want to use for the report.

For example, if you want to run the OCM Mapping Comparison report, R986101 should be displayed in the Object Configuration Manager Report Name field.

For information about a processing option, right-click the processing option field and choose What's This from the menu. Or, click the processing option field and press F1.

1. Object Configuration Manager Report Name

Use this processing option to specify the name of the report that you want to run. Valid values are:

- R986101 OCM Mapping Comparison
- R986110 OCM Global Update
- R986120 Object Configuration Delete
- R986121 Object Configuration Copy

2. Version

Use this processing option to specify the version of the report that you want to run.

R986101 (Mapping Comparison) Tab

These processing options let you specify the data sources used in the mapping comparison. You can also specify a method to use when running the comparison and how to handle report exceptions.

For information about a processing option, right-click the processing option field and choose What's This from the menu. Or, click the processing option field and press F1.

1. Data Source One and 2. Comparison Data Source Two

Use this processing option to specify the names of the data sources that you want to compare.

3. Comparison Method

Use this processing option to specify the comparison method used. The comparison method determines how OneWorld searches the data sources. When you compare in one direction, OneWorld lists records only in data source 1 that are not in data source 2. When you compare in both directions, OneWorld lists records that are in data source 1 but not in data source 2, AND records in data source 2 that are not in data source 1.

Valid values are:

1 Compare one direction only. Print only the records found in the first data source, but not in the second data source.

Blank Compare both directions. Print records in the first data source but not in the second data source, and also the records in the second data source that don't exist in the first data source.

4. Exceptions

Use this processing option to specify whether to print only the report exceptions or to print every record from the data selection, noting the differences between the data sources. Valid values are:

1 Print exceptions only
Blank Print all records

R986110 (Global Update) Tab

These processing options let you specify the data source used for the update. You can also specify whether you want to create new OCM records from existing ones or enter new values. When you elect to create new OCM records from existing ones, values you enter through the Process Control processing option will overwrite any existing values.

For information about a processing option, right-click the processing option field and choose What's This from the menu. Or, click the processing option field and press F1.

1. Process Mode

Use this processing option to specify the mode in which the report processes data. Valid values are:

1 Proof Mode. Produce a report that lets you view the records you want to delete. No records are deleted.
Blank Final Mode. Produce a report of records you want to delete. The records are then deleted.

2. Data Source name

Use this processing option to specify the name of the data source for the Object Configuration Manager table. Any updates that occur as a result of running this report will be made to the Object Configuration Manager table found in the specified data source.

3. Process Control

Use this processing option to specify whether to create new OCM records or change existing ones. Valid values are:

1 Create new OCM records from existing ones. The records created are similar to those selected, except that the new OCM values entered in the processing options will be substituted where appropriate.
Blank Change selected records with new OCM values. Use data selection to specify which records will be changed with the new OCM values entered into the processing options.

4. Enter the new OCM values

Use this processing option to enter new OCM values. If you are creating new OCM records from existing ones, any values you enter for this processing option will replace the existing ones. Valid values are:

- Environment Name
- Object Name
- Primary Data Source
- User
- Object Type
- Data Source Mode
- Secondary Data Source
- Allow QBE

R986120 (Object Configuration Delete) Tab

These processing options enable you to specify the data source from which to delete specific Object Configuration Manager records. You can also select the mode of operation and specify whether to run the Object Copy batch application (R986121), which enables you to create a copy of records before deleting them.

For information about a processing option, right-click the processing option field and choose What's This from the menu. Or, click the processing option field and press F1.

1. Process Mode

Use this processing option to specify the mode in which the report processes data. Valid values are:

1 Proof Mode. Produce a report that enables you to view the records you want to delete. No records are deleted.

Blank Final Mode. Produce a report of records you want to delete. The records are then deleted.

2. Enter Data Source Name

Use this processing option to specify the name of the data source for the Object Manager Configuration table. Any updates that occur as a result of running this report will be made to the Object Manager Configuration table found in the specified data source.

3. Process Mode

Use this processing option to specify the mode in which the report processes data. Valid values are:

1 Proof Mode. Produce a report that enables you to view the records you want to delete. No records are deleted.

Blank Final Mode. Produce a report of records you want to delete. The records are then deleted.

4. Object Copy

Use this processing option to specify whether to run the Object Configuration Copy report (R986121) along with the Object Configuration Delete report. Valid values are:

1 Run the Object Copy (R986121). This creates a copy before deleting records. When you run the Object Copy, remember to enter the processing options on the R986121 tab and to set up data selection. Data Selection should be defined on the same version of R986121 that you specified for R986120 (Object Configuration Delete).

Blank Do not run Object Copy (R986121). This deletes the records without first creating a copy.

R986121 (Object Configuration Copy) Tab

These processing options enable you to enter the target and destination data sources that are used in copying Object Configuration Manager records.

For information about a processing option, right-click the processing option field and choose What's This from the menu. Or, click the processing option field and press F1.

1. Process Mode

Use this processing option to specify the mode in which the report processes data. Valid options are:

1 Proof Mode. Produces a report that enables you to view the records you want to delete. No records are deleted.

Blank Final Mode. Produces a report of records you want to delete. The records are then deleted.

2. From Data Source

Use this processing option to specify the name of the data source with the Object Configuration Manager table from which you want to copy records.

3. Target Data Source Name

Use this processing option to specify the name of the data source with the Object Configuration Manager table to which you want to copy.

Running the Object Configuration Mapping Comparison Report

Run the Object Configuration Comparison report to compare Object Configuration Master (F986101) tables between two different data sources and display the differences between them. For example, you might compare the F986101 table in the system data source to the F986101 table for a given server map data source.

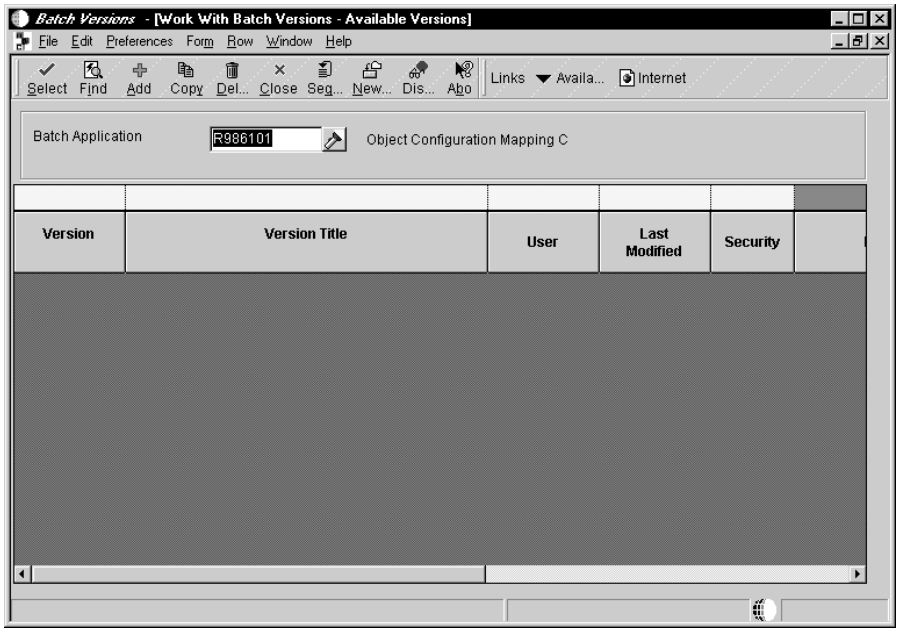
Before You Begin

- ❑ Set processing options for the report from the report driver, not from the actual report. For information about entering processing options, see *Processing Options: OCM Mapping Comparison, Global Update, Delete, and Copy*.

The following tasks describe how to set up data selection for the report and how to run the report.

▶ To set up data selection for the Mapping Comparison report

From the System Administration Tools menu (GH9011), choose Batch Versions (P98305).



Enter R986101 in the Batch Application field, and then click Find.

Double-click the version you specified on the Driver processing option tab. The Version Prompting form appears.

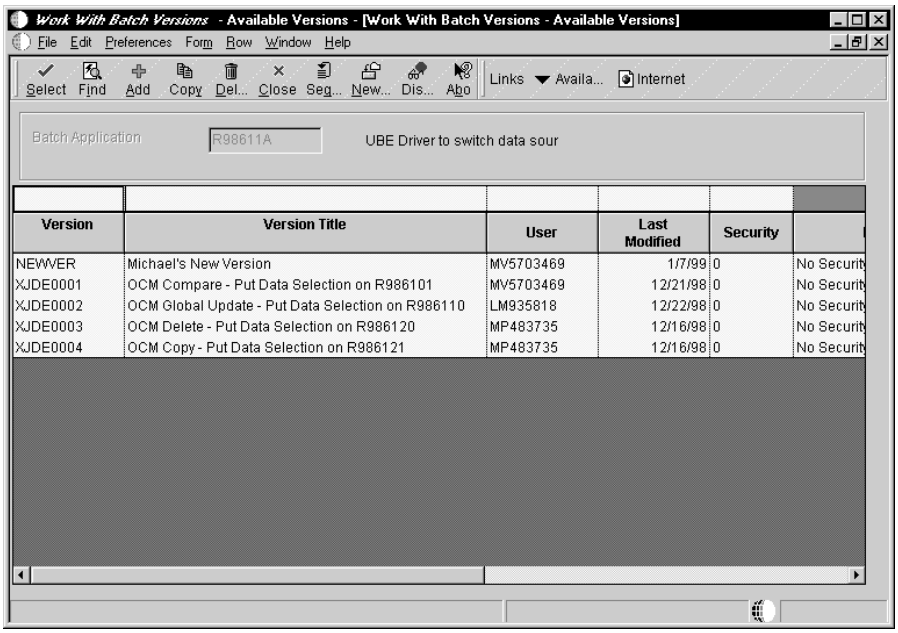
On Version Prompting, check the Data Selection option and then click Submit.

On the Data Selection form, choose from the appropriate columns to specify the exact records you want to compare.

Click OK.

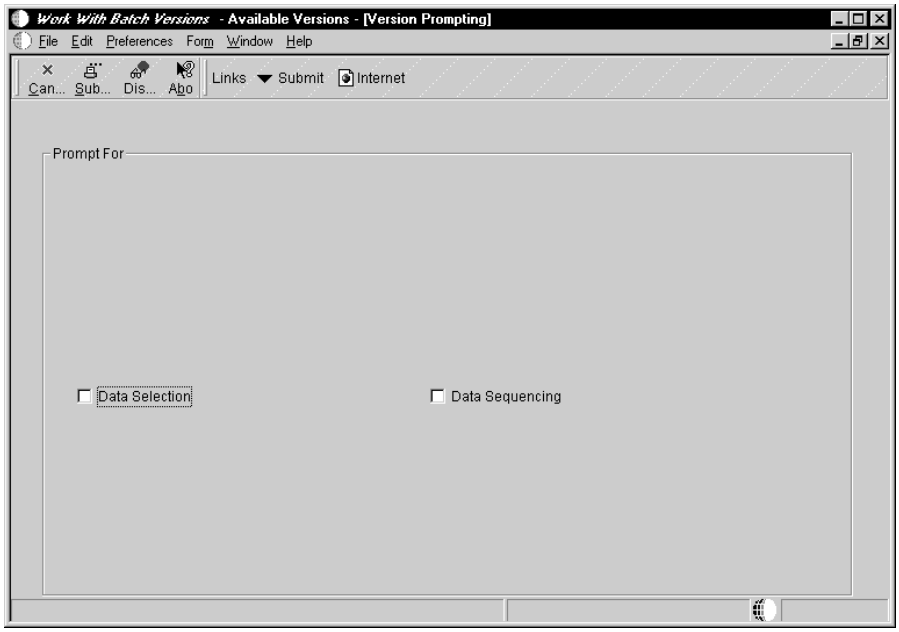
To run the Object Configuration Mapping Comparison report

From the Advanced Operations menu (GH9012), choose Object Configuration Mapping Comparison (R98611A). The Available Versions form appears.



On the Available Versions form, choose the version corresponding to the report you want to run.

Click Select. The Version Prompting form appears.



Click Submit to run the report.

Running the Object Configuration Global Update Report

This process performs global updates and global copies of object mappings in the same Object Configuration Master (F986101) table. You can use this batch application rather than the interactive application (P986110). This application is useful in updating and copying multiple records.

Before You Begin

❑ Set processing options for the report from the report driver, not from the actual report. For information about entering processing options, see *Processing Options: OCM Mapping Comparison, Global Update, Delete, and Copy*.

The following tasks describe how to set up data selection for the report and how to run the report.

▶ To set up data selection for the Global Update report

From the System Administration Tools menu (GH9011), choose Batch Versions (P98305).

Enter R986110 at the Batch Application field, and then click Find.

Double-click the version you specified on the Driver processing option tab. The Version Prompting form appears.

On Version Prompting, check the Data Selection option and then click Submit.

On the Data Selection form, choose from the appropriate columns to specify the exact records you want to update or copy.

Click OK.

▶ To run the Object Configuration Global Update report

From the Advanced Operations menu (GH9012), choose Object Configuration Copy (R98611A). The Available Versions form appears.

On the Available Versions form, choose the version corresponding to the report you want to run.

Click Select. The Version Prompting form appears.

Click Submit to run the report. The processing option form appears.

Verify processing options.

You might receive the following messages when you run this report:

- Duplicate Key - Update/Copy not done

  You cannot update a record or create a new record using the key of an existing record.

- Active DEFAULT *Public records cannot be updated

  You cannot update the DEFAULT records of the *PUBLIC group.

Running the Object Configuration Delete Report

This batch process deletes specific Object Configuration Manager records from one data source.

Before You Begin

❑ Set processing options for the report from the report driver, not from the actual report. For information about entering processing options, see *Processing Options: OCM Mapping Comparison, Global Update, Delete, and Copy*.

The following tasks describe how to set up data selection for the report and how to run the report.

▶ To set up data selection for the Object Configuration Delete report

From the System Administration Tools menu (GH9011), choose Batch Versions (P98305).

Enter R986120 at the Batch Application field, and then click Find.

Double-click the version you specified on the Driver processing option tab. The Version Prompting form appears.

On Version Prompting, click the Data Selection option and then click Submit.

On the Data Selection form, choose from the appropriate columns to specify the exact records you want to delete.

Click OK.

▶ To run the Object Configuration Delete report

From the Advanced Operations menu (GH9012) choose Object Configuration Delete (R98611A). The Available Versions form appears.

On the Available Versions form, choose the version corresponding to the report you want to run.

Click Select. The Version Prompting form appears.

Click Submit to run the report.

Running the Object Configuration Copy Report

This process copies Object Configuration Manager records from one data source to another and automatically deletes any duplicate records. For example, if you create a new environment by copying an existing one, Object Configuration Manager records are created for the new environment in the system data source.

You use the Object Configuration Copy batch process to copy those records to the appropriate server map data source. After copying the records, use the Object Configuration Global Update (R986110) process to, at the minimum, change any mappings for LOCAL to the appropriate server location.

Before You Begin

❑ Enter processing options for the report from the report driver, not from the actual report. For information about entering processing options, see *Processing Options: OCM Mapping Comparison, Global Update, Delete, and Copy*.

The following tasks describe how to set up data selection for the report and how to run the report.

▶ To set up data selection for the Object Configuration Copy report

From the System Administration Tools menu (GH9011), choose Batch Versions (P98305).

Enter R986121 at the Batch Application field, and then click Find.

Double-click the version you specified on the Driver processing option tab. The Version Prompting form appears.

On Version Prompting, check the Data Selection option and then click Submit.

On the Data Selection form, choose from the appropriate columns to specify the exact records you want to copy.

Click OK.

▶ To run the Object Configuration Copy report

From the Advanced Operations menu (GH9012), choose Object Configuration Copy (R98611A). The Available Versions form appears.

On the Available Versions form, select the version corresponding to the report you want to run.

Click Select. The Version Prompting form appears.

Click Submit to run the report. The processing option form appears.

Verify processing options.

Before copying the selected Object Configuration Manager records, this process first deletes any duplicate records.

Running the Verify Object Configuration Manager Report

The Verify Object Configuration Manager (R9861130) report is a useful troubleshooting tool that can help you verify that:

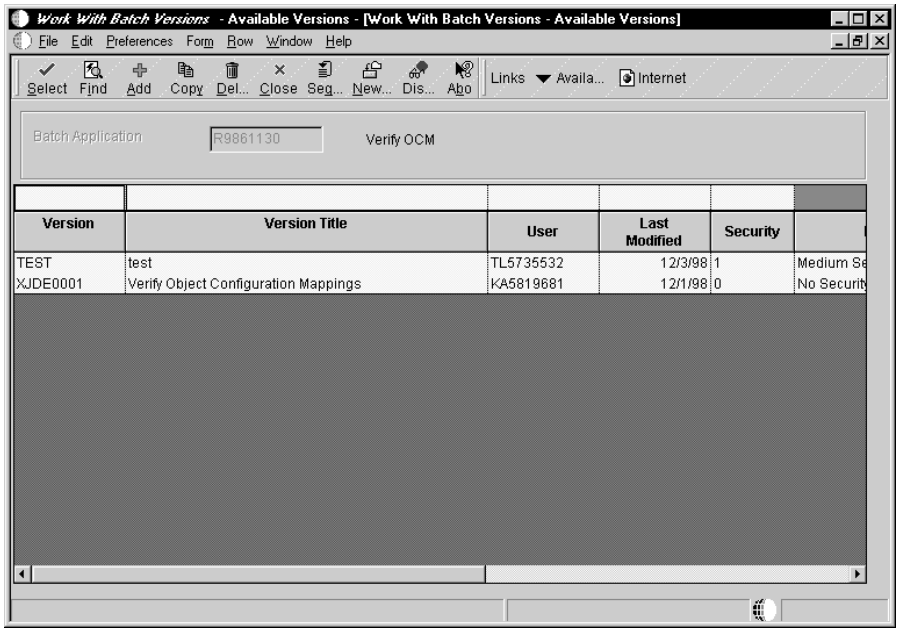
- All Data Source Master definitions have Object Configuration Manager mappings
- Object Configuration Manager mappings are consistent for a given user, that the mappings have data source definitions, and that there are no duplicate mappings
- Appropriate specification files exist on a specific server
- Business function DLLs exist on a specific server

Run this report to verify the above items for a workstation or any server. The machine on which you run this report determines which set of Object

Configuration Manager and Data Source Master tables will be verified. Consequently, you should run this report on your local machine and on each server that runs OneWorld. You can also run this report on the deployment server.

▶ To run Verify Object Configuration Manager (R9861130)

From the Advanced Operations menu (GH9012), choose Verify OCM (R9861130). The Available Versions form appears.



Choose a version, and then click Select. The Version Prompting form appears.

On the Version Prompting form, click Data Selection and then click Submit.

The Processing Options form appears.

Complete the fields on the Processing Options form. Processing options are divided into categories so that you can select the type of validation that you want. The first two options pertain to extraneous data source validation, and the remaining options pertain to object configuration mapping validations.

Leave any option blank if you do not want OneWorld to run that validation.

The fields on the Processing Options form let you do the following:

- Verify data source master definitions. Enter 1 to enable OneWorld to verify that all of your Data Source Master (F98611) definitions have Object Configuration Manager mappings.
- Enter the name of the environment you want to verify. You can also enter *ALL to verify all of your environments. This environment will be used for all of the following verifications.
- Enter the user ID of the individual or group (including *PUBLIC) for which you want to verify their object mappings. You can also enter *ALL or leave this field blank to verify all of your users. If you enter an individual's user ID, the report also prints the *PUBLIC records.
- Specify whether to verify active or nonactive mappings. Enter 1 to verify that all of the active Object Configuration Manager mappings for the user and environment that you entered have data source definitions in the Data Source Master (F98611) table. Enter 2 to verify both active and nonactive mappings.
- Check for duplicate records. Enter 1 to verify that there are no duplicate Object Configuration Manager mappings for the user and environment that you entered. This verifies duplication only for active mappings.
- Check for consolidated DLL names. Enter 1 to verify that a consolidated DLL name exists for each business function on a specific server, and that the DLL name is valid.
- Verify specification files. Enter 1 to verify that all specification files required to run OneWorld exist for the path code. This option applies only when running against a server.

The following are possible messages that can appear on this report:

The following are possible error messages that can appear on this report. Following the message are the explanation and resolution.

Message	Explanation and Resolution
Lib doesn't exist	The library that the data source references does not exist. Change the Data Source Master (F98611) to reference a valid library, or create the referenced library.
Can't open input file	The input file (TABLELIST) does not exist. On the AS/400, the input file (TABLELIST) must be in the library list. Copy the input file from the workstation (system\Bin32) to a file in the library list called TABLELIST.
Application ID is zero	The ID of the mapped object is 0. It can be 0 only if the object is equal to DEFAULT. (Some objects such as business functions and generic text do not have a nonzero ID.)

Message	Explanation and Resolution
Can't open file	Either the table does not exist in the data source, or the database is not available. If the database is available, create the table in the data source or modify the Data Source Master (F98611) record to reference a data source that contains the table.
Cannot open file	The table does not exist in the default data source. Either create a valid default data source, or add an override Object Configuration Master (F986101) record for the table.
Duplicate override	Within the same environment you have the same user and object overridden to different data sources. Delete or deactivate the duplicate override.
Data source not found	The data source does not exist as specified in the override. Do one of the following: • Create a new data source record in the Data Source Master (F98611) • Modify the Object Configuration Master (F986101) record to reference a valid data source • Deactivate the Object Configuration Master record
UBE override on server (server only)	Your enterprise server should not have any batch process overrides because the report interconnect might not work. Delete the record.
TAM of initialized Failed (spec)	The specification file does not exist, or it is corrupted. Verify the existence of the specification file and fix as needed.
Error loading (BSFN DLL)	The business function does not exist in the path code location, or the parent DLL for the business function is not valid. Verify that the parent DLL for the business function is valid.
Blank data source	The Data Source Master (F98611) has a blank data source. Delete the record.
The data source does not have overrides	The data source does not exist in the Object Configuration Master (F986101) table. Delete the data source if it is not being used. There are some cases where data sources should exist but mappings are not required. Extraneous data sources do no harm, but delete with caution.

Message	Explanation and Resolution
Blank Environment	The record is not valid because it contains a blank environment. Delete the record from the Object Configuration Master (F986101), or modify the record with a valid environment.
Duplicate record	The same record appears more than once in the Object Configuration Master (F986101) table. Delete all duplicate records.
Blank BSFN DLL	The parent DLL for the business function is blank. Specify a valid parent DLL for the business function, or delete the business function if it is not valid.

Processing Options: Verify Object Configuration Manager (R9861130)

Data Source Tab

These processing options enable you to switch on data source mapping verification, and to indicate the environments you want to verify.

For information about a processing option, right-click the processing option field and choose What's This from the menu. Or, click the processing option field and press F1.

1. Data Source Validations

Use this processing option to specify whether to perform data source validation for data source master definitions. Valid values are:

1 Verify that each data source has at least one OCM mapping defined.
Blank Do not verify data source definitions.

2. Environment Validation

Use this processing option to specify the environment(s) for which you want to run the validations. Valid values are:

*All Run the validations for all environments.
A specific environment name. (Validates only that environment.

Parent DLL Tab

This processing option enables you to verify that each business function has a valid parent DLL assigned.

For information about a processing option, right-click the processing option field and choose What's This from the menu. Or, click the processing option field and press F1.

1. Business Function Parent DLL

Use this processing option to specify whether each business function has a parent DLL assigned. Valid values are:

1 Verify that each business function has a parent DLL assigned. Also, verify that the DLL name is valid.
Blank No verification is done.

Specifications Tab

This processing option enables you to indicate whether you want to verify that all specifications required to run OneWorld exist on the server.

For information about a processing option, right-click the processing option field and choose What's This from the menu. Or, click the processing option field and press F1.

1. Validate Server Specifications

Use this processing option to verify that all specifications required to run OneWorld exist for the pathcode. This processing option applies only when running against a server. Valid values are:

1 Verify whether the required specifications exist.
Blank Do not verify specifications.

OCM Tab

These processing options enable you to enter validation parameters for User IDs, OCM mappings, and duplicate OCM mappings.

For information about a processing option, right click the processing option field and choose What's This from the menu. Or, click the processing option field and press F1.

1. User ID to Validate

Use this processing option to specify the user ID for which the OCM validations should be run. Valid values are:

Blank Run the validations for all users.
*ALL Run the validations for all users.
*PUBLIC Run the validations for only *PUBLIC.
A specific User ID. (Performs validations for only that user.)

2. OCM Mappings

Use this processing option to indicate whether to validate OCM mappings. Valid values are:

Blank Do not validate OCM mappings.
1 Validate all active OCM mappings.
2 Validate both the active and inactive OCM mappings.

3. Duplicate OCM Mappings

Use this processing option to specify whether OCM records should be checked for duplicates. Valid values are:

Blank Do not check for duplicates.
1 Verify that the active OCM mapping records have no duplicates.

Running the Object Configuration System Table Update Report

This process adds active Object Configuration Manager (OCM) records for a specified table, user ID, and data source for all environments listed in the Environment Detail - OneWorld table (F00941). You can use data selection to filter the environments to which this process adds OCM records.

▶ To run Object Configuration System Table Update (R986101A)

From the Advanced Operations menu (GH9012), choose Object Configuration System Table Update (R986101A). The Available Versions form appears.

On the Available Versions form, choose XJDE0001 and then click Select.

On the Versions Prompting form, click Data Selection and then click Submit.

On Data Selection, determine which records to update and then click OK.

The Processing Options form appears.

On the Processing Options form, enter the following information:

- Enter the table name for which to create mappings.
- Enter the data source to which the table should be mapped.
- Enter the user ID for whom to map the records. The value *PUBLIC maps the records for all users. You can also map by group.
- Specify proof or final mode. Enter 1 for final mode. If left blank, application runs in proof mode. Run this process in proof mode first to test whether the current values create a successful result.

Processing Options: System Table Update (R986101A)

Process Tab

These processing options enable you to specify the table name, data source, and user ID to use in creating an OCM mapping. You can also specify whether to run the report in proof or final mode.

For information about a processing option, right-click the processing option field and choose What's This from the menu. Or, click the processing option field and press F1.

1. Enter a specific Table Name

Use this processing option to specify the name of the table you want to map to each environment. The table name, along with the values you specify for the Data Source and User ID processing options, will is to create a whole OCM Mapping.

2. Enter a specific Data Source

Use this processing option to specify the data source you want to map to each environment. The Data source, along with the values you specify for the Table Name and User ID processing options, is used to create a whole OCM Mapping.

3. Enter a specific User ID

Use this processing option to identify the user ID that will be mapped to each environment. The user ID, along with the values you specify for the Table Name and Data Source processing options, will be used to create a whole OCM Mapping.

4. Proof/Final Mode

Use this processing option to indicate whether the batch application runs in Proof or Final mode. Running the batch application in Proof mode means that records are not changed. Final mode updates and adds records. Valid values are:

Blank Proof mode.
1 Final mode.

Running the Create OCM Records for Business Functions Report

This batch process reads the Object Librarian tables for server business functions, and then creates OCM records for those business functions in the target data source you specify through processing options. Processing options also allow you to specify the source data source and environment to use when creating these OCM records.

▶ To run Create OCM Records for Business Functions (R986140)

From the Advanced Operations menu (GH9012), choose Create OCM Records for Business Functions (R986140). The Available Versions form appears.

On the Available Versions form, choose XJDE0001 and then click Select.

On the Versions Prompting form, click Data Selection and then click Submit.

On Data Selection, specify which records to create, and then click OK. The Processing Options form appears.

On the Processing Options form, enter the following information:

- Specify proof or final mode. When you enter 1, the report runs in Final mode. This means the report will be printed and reports will be updated. When you enter 0 or leave this field blank, the report runs in Proof mode. This means the report will be printed but no updating occurs.
- Enter the name of the data source to use when creating OCM records for the business functions.
- Enter the name of the environment to use when creating OCM records for the business functions.
- Enter the name of the data source where you want to place the created OCM records for server business functions.

Running the Job Master Deletion By Days Old Report

This batch application lets you produce a report listing obsolete print jobs submitted to servers. You have the option of generating the report only, or generating the report and then deleting obsolete records from the F986110 table.

This batch application is launched by a report driver. Enter any data selection from the batch application, but enter processing options from the report driver (R9861101), not from the actual application (R9861102). For information about entering processing options, see *Processing Options: Job Master Deletion By Days Old Report (R9861102)*.

▶ To set up data selection for Job Master Deletion

From the System Administration Tools menu (GH9011), choose Batch Versions (P98305).

Enter R9861102 at the Batch Application field, and then click Find.

Double-click one of the following versions:

- XJDE0001 = Jobs With All Status
- XJDE0002 = Jobs With Done Status
- XJDE0003 = Jobs With Error Status
- XJDE0004 = Jobs With Wait Status

The Version Prompting form appears.

On Version Prompting, check the Data Selection option and then click Submit.

On the Data Selection form, choose from the appropriate columns to specify the exact records you want to compare.

Click OK.

▶ To run Job Master Deletion By Days Old (R9861101)

From the Advanced Operations menu (GH9012), choose Job Master Deletion By Days Old (R9861101). The Available Versions form appears.

On the Available Versions form, select the version corresponding to the report you want to run.

Click Select. The Version Prompting form appears.

On Version Prompting, click Submit.

On the Processing Options form, enter the following information and then click OK:

- Data Source
- Days Old
- Control Mode
- Job Control Status UBE

For information about entering processing options, see *Processing Options: Job Master Deletion By Days Old Report (R9861102)*.

Processing Options: Job Master Deletion By Days Old Report (R9861102)

The batch application Job Master Deletion By Days Old (R9861102) is launched from report driver R9861101. Enter processing options for this batch application through the report driver.

This processing option has two tabs: Defaults and Versions. Use the Defaults tab to specify the data source, the number of days old criterion, and the job control mode (Proof or Final mode). Use the Versions tab to enter the version you want to use when you run the R9861102 batch application.

Defaults Tab

These processing options let you specify the data source for the server, the number of days old a report must be in order to be included on the report (and eligible for deletion), and the control mode.

For information about a processing option, right-click the processing option field and choose What's This from the menu. Or, click the processing option field and press F1.

1. Data Source

Use this processing option to specify the data source name that will be used in the Job Master Deletion report (R9861102).

2. Days Old

Use this processing option to include in the report submitted print jobs that have been on the server longer than the specified number of days. For example, if you enter 10, the report includes print jobs that have been on the server for 10 days or more.

3. Control Mode

Use this processing option to specify the control mode for the report. Valid values are:

1: Proof Mode. Generate a report of obsolete print jobs, but not delete the jobs.
2 Final Mode. Generate a report of obsolete print jobs and delete them from the server.

If you are unsure about whether to delete records, consider running the batch application in proof mode first, and then final mode after you have verified that you want to delete the records listed.

Versions

This processing option lets you specify the version to use when you run the batch application. Depending on the version you select, different print jobs will be included on the report.

For information about a processing option, right-click the processing option field and choose What's This from the menu. Or, click the processing option field and press F1.

1. Job Control Status UBE (R9861102)

Use this processing option to specify the version of the R9861102 batch application that you want to run. Valid values are:

- XJDE0001 Jobs With All Status
- XJDE0002 Jobs With Done Status
- XJDE0003 Jobs With Error Status
- XJDE0004 Jobs With Wait Status
- Any version you have defined

Running the OCM Category Update/Delete Report

This batch application lets you add to or delete from the Object Configuration Manager table (F986101) member objects of a specified category. This means that you can add or delete OCM mappings for all members in the same category rather than adding or deleting them individually. The processing options for this batch application allow you to specify whether to add or delete mappings, as well as the appropriate path code, environment, and data source to use.

To run OCM Category Update/Delete (R986101B)

From the Advanced Operations menu (GH9012), choose OCM Category Update/Delete (R986101B). The Available Versions form appears.

On the Available Versions form, choose XJDE0001 and then click Select.

On the Versions Prompting form, click Data Selection and then click Submit.

On Data Selection, specify which records to add or delete, and then click OK. The Processing Options form appears.

On the Processing Options form, click the OCM Modes tab and enter the following information:

- Specify proof or final mode. When you enter 1, the report runs in Proof mode. The report will be printed and the Object Configuration Manager table will not be updated. When you enter 2, the report runs in Final mode. The report will be printed and the Object Configuration Manager table will be updated.
- Specify add or delete mode. Enter 1 to add OCM mappings for objects in the category, or 2 to delete mappings.
- Enter the override mapping. When you enter 1, OCM mappings for objects that already have mappings for the environment and user are deleted. When you enter 2 or leave this field blank, mappings for these conflicting OCM records are deactivated, but not deleted.

Click the OCM Settings tab and enter the following information:

- Enter the path code you want to use when adding or deleting OCM mappings. When you enter a path code, mappings will be added or deleted for all environments with the path code you enter. You don't need to enter a path code if you entered an environment name through the Environments processing option. If this field is blank, OCM mappings will be added or deleted regardless of the path code.
- Enter the environment you want to use when adding or deleting OCM mappings. If this field is blank, OCM mappings will be added or deleted regardless of the environment. If you entered a path code through the Path Code processing option, that path code is used. If both the Environments and Path Code processing option fields are blank, OCM mappings will be added or deleted for all environments in the Environment Detail (F00941) table.
- Enter the user class or group. If you are deleting OCM mappings and this field is blank, all OCM records will be deleted regardless of the user class or group. If you are adding OCM mappings, you must enter this field.

- Enter the data source. If you are deleting OCM mappings and this field is blank, all OCM records will be deleted regardless of the data source. If you are adding OCM mappings, you must enter a value in this field.

Partitioning Application Logic on Servers

The logic for OneWorld applications can be partitioned to run remotely by mapping individual or specified groups of business function components to run on an application server or enterprise server instead of on a workstation.

J.D. Edwards has found that redeploying certain business function components (including master business functions and business functions) can significantly increase the performance of a distributed OneWorld workstation while simultaneously decreasing network traffic. This redeployment involves remapping objects using OneWorld's standard Object Configuration Manager (OCM) methodology.

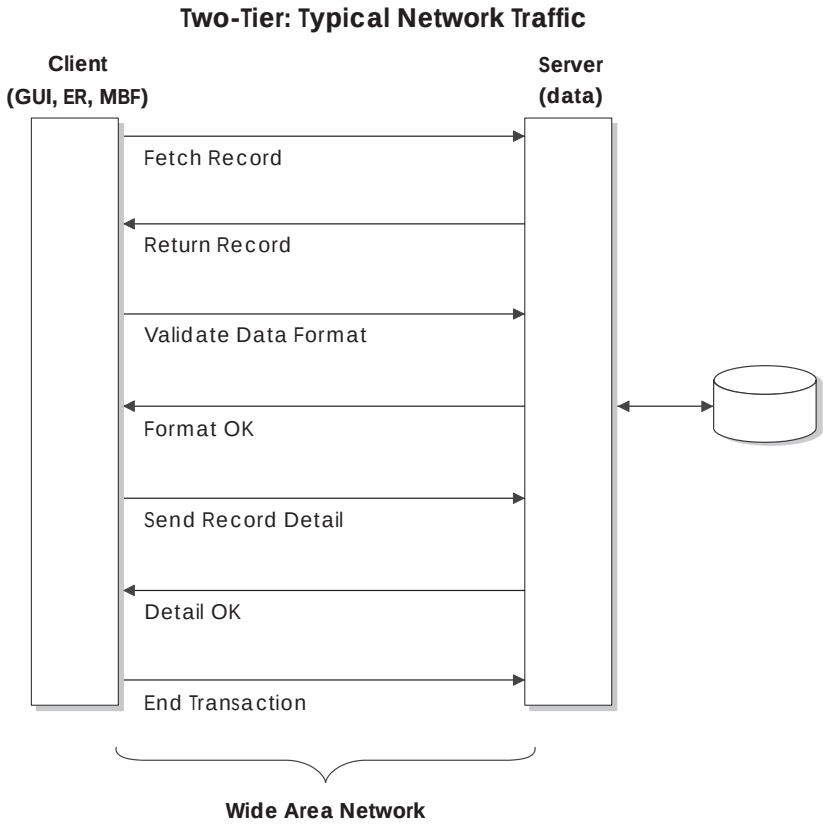
Examples of such configurations are illustrated by the Windows light client/heavy server and the Java light client/heavy server models. Both models have applications specifications on the client and business function components on the server. The main difference is that the Windows model uses JDENet communication middleware, while the Java model uses JDENet/CORBA middleware.

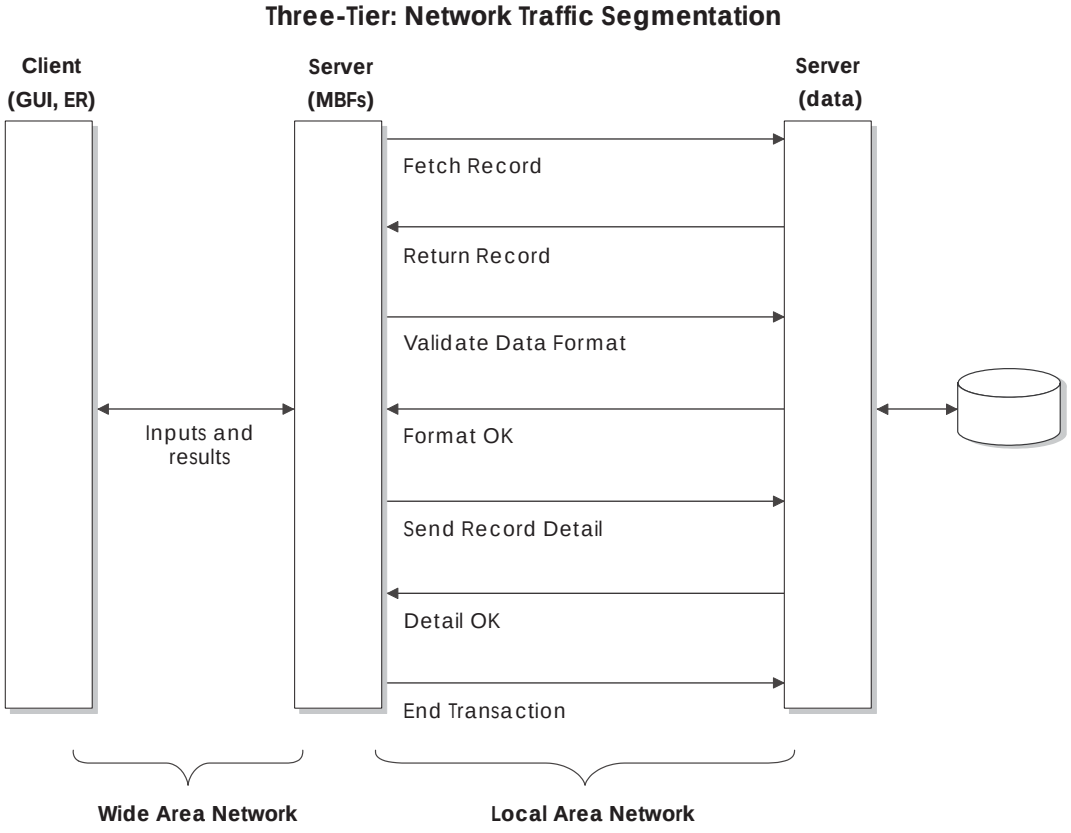
While the OneWorld design allows you to partition all OneWorld business function components, the biggest benefit is derived from partitioning Master Business Functions (MBFs).

OneWorld transaction-oriented applications are built around the concept of MBFs, which are typically responsible for transaction edits and for committing transactions to the database. Most of the I/O services for transaction-oriented applications are performed by MBFs. By localizing the majority of business logic for transactions in MBFs and partitioning the MBFs to run on application servers, network traffic can be minimized, thus dramatically improving the performance of the application in distributed and WAN environments.

In a two-tier setup where MBFs are processed on the client, there is a lot of interaction back and forth across the WAN between client and server. In a three-tier setup consisting of a client, a data server and an application or enterprise server, transaction processing can occur across a LAN between the two servers. Interaction across the WAN between the client and server is thus reduced to entering input on the client and sending back results from the server. This three-tier configuration can result in a significant reduction in traffic across the WAN.

The following figure contrasts typical network traffic for a two-tier setup where MBFs are processed on the client, versus network traffic segmentation for a three-tier setup where MBFs are processed on the server.





How a Master Business Function Operates

The following series of events demonstrates how a typical application utilizes an MBF. This example uses the Sales Order Entry (SOE) application.

End of Sales Order Line

The first event occurs when the end of a sales order line is reached, causing the OneWorld client application to call the jdeCallObject API. This command sends a message to the MBF. Included with the message is data (in the form of a data structure) for the line. The application sends asynchronously the message with its associated data. That is, once the message is sent, the client application proceeds to the next line.

MBF Receives Line Message

This event occurs when the MBF receives the JDENet message that includes the data for the line. The line data is cached in the server's shared memory.

MBF Extends and Edits the Line

This event occurs when the MBF extends and edits the sales order line. The data necessary to extend and edit the line is typically accessed locally on a LAN. The data is requested by a database-dependent SQL call and is transported by the applicable Open Database Connectivity (ODBC) or Oracle Call Level Interface (OCI) mechanisms.

MBF Sends a Return Message to the Client Application

The fourth event occurs after the MBF extends and edits the sales order line and returns the extended line as well as any error codes to the client. The return message is sent using JDENet. Events 1 through 4 are then repeated asynchronously for all of the lines associated with the sales order.

End of Sales Order (OK Button)

This event indicates that the user has completed all sales order lines. The user triggers this event by clicking OK after all edited lines have been returned to the client. When the user clicks OK an "end of transaction" message is sent to the MBF. The client is then immediately released to enter the next transaction.

MBF Processes the Full Transaction

The full transaction is processed when the MBF asynchronously reads the shared memory cache (where all transaction lines are stored) and begins the process of committing the transaction to the database.

Transaction Commitment to the Database and MBF Cleanup

The MBF commits the entire transaction to the database, typically locally through ODBC and OCI, and cleans up the shared memory cache for the completed transaction.

Mapping the MBF to run on the server causes the bulk of the database and logic interaction to occur within a single server machine (enterprise server) or between LAN-attached machines (application server and data server). This means that the transaction has been processed with a minimum of network traffic. This type of application transaction is ideally suited for performance gains in distributed and WAN environments.

Server Behavior with Partitioned Application Logic

Compared to a typical heavy-client scenario, partitioning application logic by configuring business function components (such as Master Business Functions and other business functions) to run on the server requires the server to run more processes and to manage additional user sessions. There are important jde.ini parameters that must be set to control the server's behavior under this increased workload. For details on setting these jde.ini parameters, see *Middleware* in the *System Administration Guide*.

The following summary rules explain how OneWorld manages the workload of running business function components (MBFs) on an application server or enterprise server.

Processing Rule 1 - JDENet on Client

On the client, the JDENet functions reside in a dynamic-link library (DLL) called jdenet.dll. These functions are called by the OneWorld Explorer program (Oexplore.exe). That is, they are not run as a separate process or service; they are run from within the Oexplore.exe process.

When OneWorld clients first initiate a communication session with a OneWorld server (as defined by the Object Configuration Manager), they are assigned to communicate with a specific JDENet process on that server. This assignment persists for the OneWorld session. That is, the same logical connection is maintained for as long as the user is signed on.

Processing Rule 2 - JDENet on the Server

You can configure multiple JDENet processes to run on a server. Parameters in the server's jde.ini file specify how many JDENet processes can be started on the server, as well as the total number of network connections that can occur to and from that server.

If you specify multiple JDENet processes, OneWorld starts the processes as required on a one-for-one basis with incoming session requests until the maximum number of JDENet processes are started. Then, OneWorld sequentially assigns subsequent sessions to JDENet processes. Within each JDENet process messages are queued and processed one at a time.

Example:

```
Session 1: JDENet_1
Session 2: JDENet_2
Session 3: JDENet_3
Session 4: JDENet_1
Session 5: JDENet_2
Session 6: JDENet_3
```

The preceding example is based on the characteristics listed in the following table:

Characteristic	Value	jde.ini Parameter
Number of JDENet processes	3	`[JDENet]` `maxNetProcesses=3`

Number of connections per server	*nnn*	This is a site-specific variable number. Typically the setting should be a value high enough to accommodate the practical maximum for the installation.
		
		A connection is defined as a process request by a client (such as a log-on) or a server (such as an application server connection to a data server). If this number is set too low, when the maximum number of connections is reached, no additional OneWorld processes can connect to this server.
		
		For example:
		
		`[JDENet]`
		`maxNetConnections=800`

Number of incoming sessions	6	N/A

Processing Rule 3 - Kernels on the Server

To handle different functions, the OneWorld architecture allows for different types of kernel processes to run on the server. The kernel type that processes distributed objects through the jdeCallObject API is a Type 2 kernel. A parameter

in the server's jde.ini file specifies how many individual kernels of a specific kernel type can be started.

The total number of active sessions that might connect to a kernel cannot be directly controlled. That is, OneWorld dynamically allocates sessions to applicable kernel types on an as-available basis. However, you can indirectly control the number of users per kernel by specifying a sufficient number of kernels in the desired relationship based on the number of connections (client-to-server and server-to-server). For example, you can specify enough kernels to have one user for each kernel, or two users for each kernel, and so on.

In determining the number of required kernels, remember that each kernel process consumes server memory resources. The exact amount of memory consumed is not as important a consideration as the performance aspect. Consider how many users (or MBFs) can simultaneously use a single kernel before significant performance degradation begins to occur.

Kernel processes are started in a manner similar to the JDENet network communication sessions. For each kernel type, OneWorld starts a new kernel for each new session until the maximum number of kernels allowed are started. After the maximum is reached, OneWorld sequentially assigns sessions to a specific kernel process.

Each Type 2 kernel queues and processes a single jdeCallObject API request at a time. If multiple sessions are assigned to a single kernel, when the jdeCallObject routine completes it takes the next request off the queue for that kernel type.

Care must be taken when configuring the jde.ini file for the server. Using this kernel-balancing methodology, there is no way to distribute the jobs being executed based on the nature of those jobs. As illustrated in the following example, this can result in Kernel_2 being loaded with heavy Sales Order Entry Processing while the other kernel, Kernel_1, is idling with less process-intense functions. This insight into load balancing may be used to your advantage when considering hardware tiers.

See *Middleware* in the *System Administration Guide* for information about jde.ini settings applicable to all server platforms.

Example:

```
Session 1: JDENet_1, Kernel_1
Session 2: JDENet_2, Kernel_2
Session 3: JDENet_3, Kernel_1
Session 4: JDENet_1, Kernel_2
Session 5: JDENet_2, Kernel_1
Session 6: JDENet_3, Kernel_2
```

The preceding example is based on the characteristics listed in the following table:

Characteristic	Value	Jde.ini Parameter
Number of JDENet processes	3	`[JDENet] maxNetProcesses=3`
Number of incoming sessions	6	N/A
Number of Type 2 kernels	2	`[JDENet_KERNEL_DEF2] maxNumberOfProcesses=2`
Number of sessions requesting Type 2 kernel	6	N/A

For more information about kernels, see the *System Administration Guide*.

When Not to Distribute Logic

There are several scenarios in which distributing logic is not a good idea. These scenarios are explained below:

- When a business function mapped to the server creates cache that is required by another business function invoked from the calling application, then those business functions must also be mapped to the same logic server as the business function creating the cache.
- Client-only business functions contain GUI or Windows functions that are only valid on a Windows NT workstaiton. Because of this, master business functions distributed to a logic server may not call a business function that can only execute on a client.

Note: When a master business function is mapped to a logic server, the server looks to its own server map data source to determine OCM tables for processing. For example, if a master business function is mapped to a logic server, then any logic called by that master business function will be called based on the server map data source in the OCM.

Issues with Distributing Logic

There are several issues known about distributing logic. These issues are explained below:

- Batch control functionality may cause problems with distributed logic. When exiting a transaction application, the batch control business function presents an additional form for review. There is no problem if the application calls the batch control business function directly. However, the call will fail if the mapped master business function calls the batch control

business function because GUI presentations cannot be performed from the server. For example, the Purchasing and Inventory applications cannot use batch control if logic is mapped to the server.

- When implementing localization and custom logic, you must avoid mapping logic that contains GUI or windows functionality.

Business Function Processing Problems

OneWorld's configurable network computing solution allows developers and administrators to map business functions to one or more application servers for logic processing. When a problem occurs on the server, OneWorld attempts reconnect to the application server so that the business function can run. If OneWorld can reconnect to the server and run the business function, work proceeds uninterrupted.

However, specific circumstances can complicate business function processing:

- The client workstation cannot reconnect to the application server because a server process has died.
- Business function processing creates cache, or state information, on the application server whose process has died.
- The business function causes one or more processes to die on the server.

When the client workstation cannot communicate with the server, OneWorld redirects business function processing to a secondary server. A list in the CallObject code designates the name of the original server and the name of the secondary server to which future calls should be rerouted.

> **Note:** The default configuration is that no secondary server is defined during the OneWorld installation process. Defining a server will require changes to your OCM mappings. If you do not define a secondary server and failover occurs, OneWorld remaps business function processing from the failed server to the client workstation.

When business function processing creates cache on the application server where a process has died, the client workstation reconnects to the application server, but the user must exit the application and restart it.

When a business function causes one or more processes to die on the server, the client workstation reconnects to the server. Because the business function is causing the jdenet_k process to die, OneWorld simply fails the business function call.

OneWorld's CNC solution provides a methodology that handles business function failure and allows you to continue working, even when a server has failed or a kernel process has died, ending the processing of logic on an application server. In addition, OneWorld writes a message to the jde.log whenever a failover occurs, allowing you to troubleshoot the problem.

This chapter discusses the way CNC handles the following circumstances that affect business function processing:

- ❑ Failure to connect to the server
- ❑ Business function failure
- ❑ Cache on the server when a business function failure occurs

Failure to Connect to the Server

OneWorld redirects business function processing if it cannot connect to the primary server. The following sequence of steps describes what occurs during the initial stages of an attempt to call a business function to run on an application server:

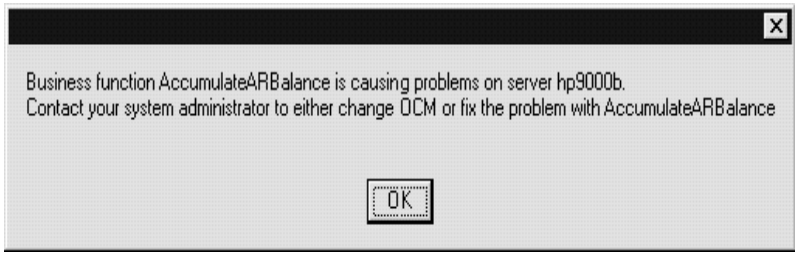
- First, the user calls a business function on a server.
- OneWorld checks to see if the server has been failed over from the primary server to a secondary server or to the client workstation.
- If processing has been directed to another server, OneWorld remaps the business function and sends the CallObject message to the secondary server or to the client workstation to run the business function.
- If the server has not been failed over, OneWorld sends the CallObject message to the original server to run the business function.

In the second phase of business function processing, OneWorld attempts to run the logic on the application server or client workstation. The following sequence of steps describes what occurs during the second stage of processing:

- If the business function runs without error, either on the original server or the failover alternative, the request has been processed.
- If the client workstation request is not successfully processed by the server, OneWorld increments a reconnect counter and attempts one reconnection.
- If the value on the reconnect counter is greater than 1, the business function fails. If the value on the reconnect counter is not greater than 1, OneWorld reconnects to the server and attempts to run the business function.
- If the client is unable to reconnect to the server, the request is redirected to a secondary server, if one is defined, or to the client workstation if one is not defined.
- If cache has been created on the server, the user must exit the application and restart it.

Business Function Failure

Finally, the business function itself might cause one or more processes to die on the server. If this is the case, OneWorld displays a dialog box indicating that the business function is causing problems. You might have to change OCM mappings or fix a bug in the business function.



Cache on the Server When a Business Function Failure Occurs

If the business function does not run the first time, OneWorld checks to see if cache was created on the server during the first failed attempt. If no cache is created and the reconnection attempt to the primary server fails, OneWorld attempts to run the business function on the secondary server or the client workstation.

However, if cache is created on the server, OneWorld instructs the user to exit the application and start over.

The creation of cache on the server is vital to the processing of business functions. OneWorld creates cache when one business function runs so that one or more subsequent functions can use the data in the cache. For example, one business function might create and initialize the cache, a second might add data to it, and a third might access the data and insert it into a database.

If a process on the server dies after the first business function creates the cache and the client workstation is unable to communicate with the process on the server that contains the cache, the subsequent business functions are not able to access the original cache. Therefore, in this scenario, OneWorld forces you to exit the application and start over.

Note: UBEs and table conversions continue to process business functions after a failure, even if they create cache on the server.

Environment Setup

OneWorld environments, which you define, are collections of pointers indicating the location of data and OneWorld objects. A OneWorld environment definition contains a path code and a set of Object Configuration Manager mappings.

This section defines concepts of environments and how to set up and maintain OneWorld environments.

This section contains the following:

- ❑ Understanding environments
- ❑ Working with an environment

Understanding Environments

A OneWorld environment consists of Object Configuration Manager mappings and a path code. The Object Configuration Manager mappings for an environment provide answers to the following questions:

- Where are my logic objects processed?
- Where do my data objects reside?

The path code associated with an environment provides an answer to another question:

- In what directory are the objects?

OneWorld has the following types of environments:

- Distributed data and distributed logic environments

  Distributed data and logic environments determine where data resides and where application processing occurs.

- Mode of processing environments

  Mode of processing environments are environments designed to support the three types of transaction processing available with OneWorld: store and forward, direct connect, and batch of one.

- Group of object environments

  Group of object environments control which set of objects are used, such as production, development, or pristine.

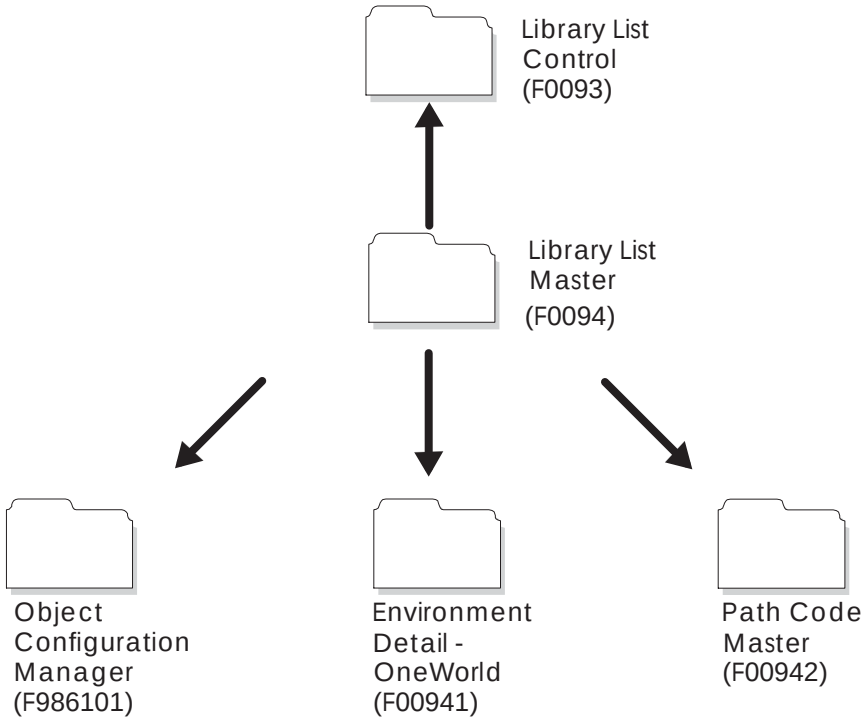
Where OneWorld Stores Environment Definitions

A OneWorld environment definition involves the following tables:

- Library List Control (F0093)
- Library List Master (F0094)
- Environment Detail - OneWorld (F00941)
- Object Configuration Manager (F986101)

- Path Code Master (F00942)

The following illustrates the relationship of the tables involved in an environment definition.



The Library List Control Table (F0093)

The Library List Control table (F0093) contains valid environments for each user. You must assign to each user at least one valid environment for signing onto OneWorld. The environments you assign to each user in the Library List Control table are validated at OneWorld startup against the OneWorld directories on the workstation. Only those environments that a user is authorized to sign onto and that are installed on the machine they sign onto are listed as available environments.

The Environment Revisions form of the User Profile application enables you to specify the users who are authorized to sign onto specific environments.

The Library List Master Table (F0094)

The Library List Master table (F0094) contains the name of the environment and the description.

If OneWorld and WorldSoftware coexist, you can share the Library List Master table between both products. By using existing WorldSoftware environment names, you do not need to duplicate existing user profile information.

The Environment Detail - OneWorld Table (F00941)

This table contains the environment name and the associated path code from the Path Code Master (F00942) table.

Every environment must have an associated path code. Environments can share the same path code. For example, two environments can use production objects and have different data location mappings. This would be the case if you had a group of users processing against data on a corporate server and a group of users processing against data on a departmental server.

The Object Configuration Master Table (F986101)

The Object Configuration Master table has data and logic object mappings for every environment. Every environment must have entries in the Object Configuration Manager table. This is true even if the mappings for the environments are identical. Two environments might have different path codes but have the same mappings for data and logic. For example, developers and testers could have different sets of objects but their data is in the same database and their logic processes locally.

The Object Path Master Table (F00942)

This table stores the location of the path code's central objects, the release associated with this path code, and other details.

Coexistence Considerations

In a coexistence environment where WorldSoftware and OneWorld require access to the same tables, you must define the shared tables in WorldSoftware. Tables that are created and generated in OneWorld cannot be accessed by WorldSoftware's RPG programs even when they reside in an AS/400 program, because the AS/400 cannot read OneWorld's Access, Oracle, or Microsoft SQL Server database format.

Therefore, when you initially set up your coexistence environment, all OneWorld tables that store environment information (F0093, F0094, F00941, and F00942) must be created on the AS/400.

Similarly, if you plan to create custom modifications that are shared by WorldSoftware, you must create new or shared tables using the AS/400 database structure. Do not create these tables in OneWorld.

If you have questions or concerns about setting up your coexistence environment, contact J.D. Edwards Worldwide Customer Support or your J.D. Edwards representative.

Working with an Environment

You can create a new environment either by adding a new environment or by copying an existing environment.

Because the Object Configuration Manager (OCM) mappings are an important element of the environment, determining these mappings would be a good first step in deciding whether to create a new environment or copy an existing environment.

When you copy an environment, that environment's OCM mappings are copied along with the environment. It makes sense to create a new environment by copying an existing environment when the OCM mappings for the environment you want to create closely match the mappings of an existing environment.

For example, a development environment and a testing environment might contain very similar mappings with only a few differences. In these cases it is easier to copy an environment and change only the mappings that are different.

On the other hand, you should add a new environment rather than copying an existing environment when you do not want to use another environment's object mappings.

For example, if you are setting up store and forward processing consider adding a new environment because the OCM mappings for the store and forward users are completely different from the OCM mappings for the direct connect users.

Keep in mind that when you add an environment without copying an existing one you must create the OCM mappings manually. For this reason, it is typically easier to create a new environment by copying an existing one, and then adjusting the OCM mappings instead of creating them all manually.

This topic contains the following:

- ❑ Adding an environment
- ❑ Copying an environment
- ❑ Deleting an environment
- ❑ Processing Options: Environment Master (P0094)

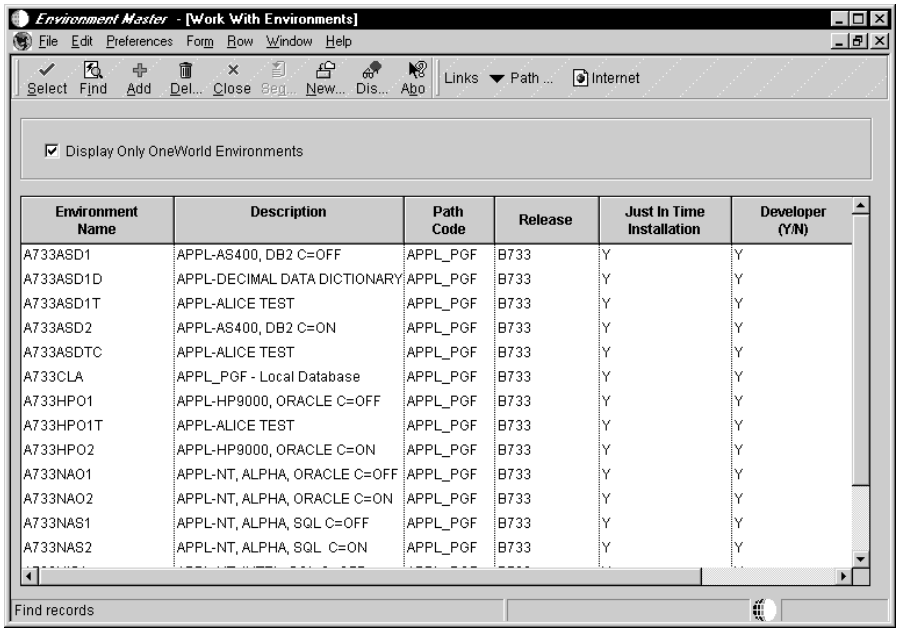
Adding an Environment

Add a new environment rather than copying an existing environment when you do not want to use another environment's OCM mappings.

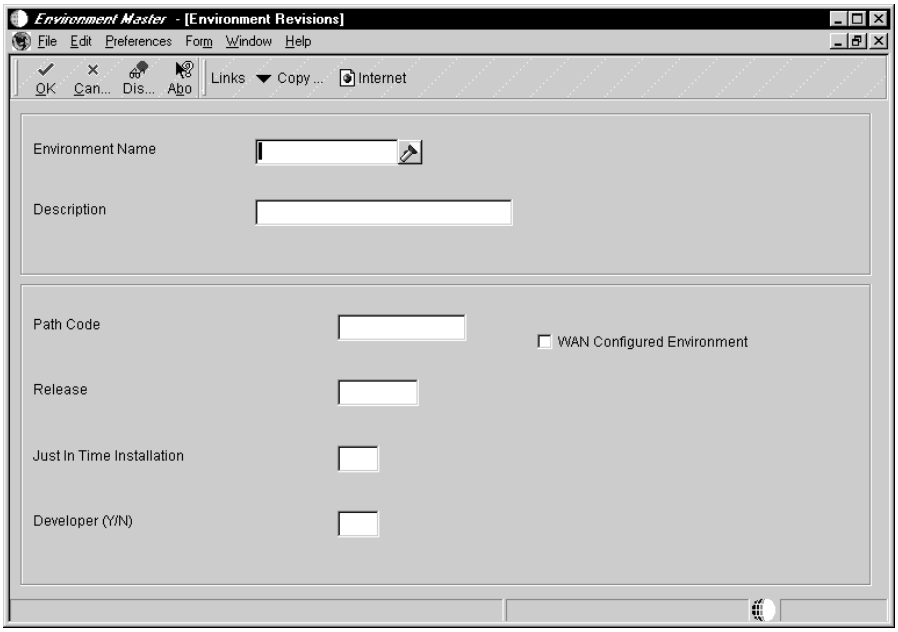
▶ To add an environment

1. On the Environments menu (GH9053), choose Environment Master (P0094). The Work With Environments form appears.

   If the Display Only OneWorld Environments option is checked, this form displays only those environments that have an entry in the Environment Detail - OneWorld table (F00941). If the option is not checked and you operate a coexistence enterprise, you will also see your AS/400 libraries.



2. Click Add. The Environment Revisions form appears.



3. On the Environment Revisions form, complete the following fields, and then click OK:

   - Environment Name
   - Description
   - Path Code
   - Release
   - Just In Time Installation
   - Developer (Y/N)
   - WAN Configured Environment

   At this point there will be a new environment in the Environment Master table (F0094). For that new environment, a record is created in the Environment Path Code Tag table (F00941).

4. If the path code you enter on the Environment Revisions form does not exist on any other record in the Environment Master File (F0094), the system displays the Data Source Selection form.

   When this form appears, complete the OL Data Source field.

   The system uses the data source you enter on this form to create Object Librarian OCM mappings for the new environment. The system writes the required records to the Object Configuration Master Table (F986101).

   If the path code you enter on the Environment Revisions form already exists in another environment record, the system determines the data source that is associated with that environment's Object Librarian OCM

record. The system uses this value to create a new record in the Object Librarian Object Configuration Master Table (F986101).

5. Add other Object Configuration Manager mappings. For more information, see *Working with the Object Configuration Manager*.

Field	Explanation
Environment Name	The name associated with a specific list of libraries. The J98INITA initial program uses these library list names to control environments that a user can sign on to. These configurations of library lists are maintained in the Library List Master table (F0094). For OneWorld, this field represents a valid environment that can be used to run OneWorld. The environment encompasses both a path code (objects) and a data source (data). When put together, users have a valid workplace within OneWorld.
Description	A user defined name or remark.
Path Code	The Path Code is a pointer to a set of OneWorld objects, and is used to keep track of sets of objects and their locations within OneWorld.
Release	For World, the release number as defined in the Software Versions Repository. For OneWorld, the release number as defined in the Release Master.
Just In Time Installation	Use this field to turn ON/OFF just-in-time installation for anyone signed onto this environment. Consider turning just-in-time installation OFF before you transfer modified applications into the production path code. Once you have fully tested the applications and are ready for production users to receive the changes, you can turn just-in-time installation back ON.
Developer (Y/N)	A one byte field for the Install Group. Future use.
OL Data Source	The data source name.
WAN Configured Environment	This code allows the user to indicate that this environment is an environment that is configured for the WAN. 1 WAN Configured Environment Blank Not Configured for WAN

Copying an Environment

When you create a new environment by copying an existing environment, OneWorld also copies the Object Configuration Manager mappings associated with the existing environment. You can set up a processing option if you want to specify additional Object Configuration Manager mappings to copy at the time that you copy the environment.

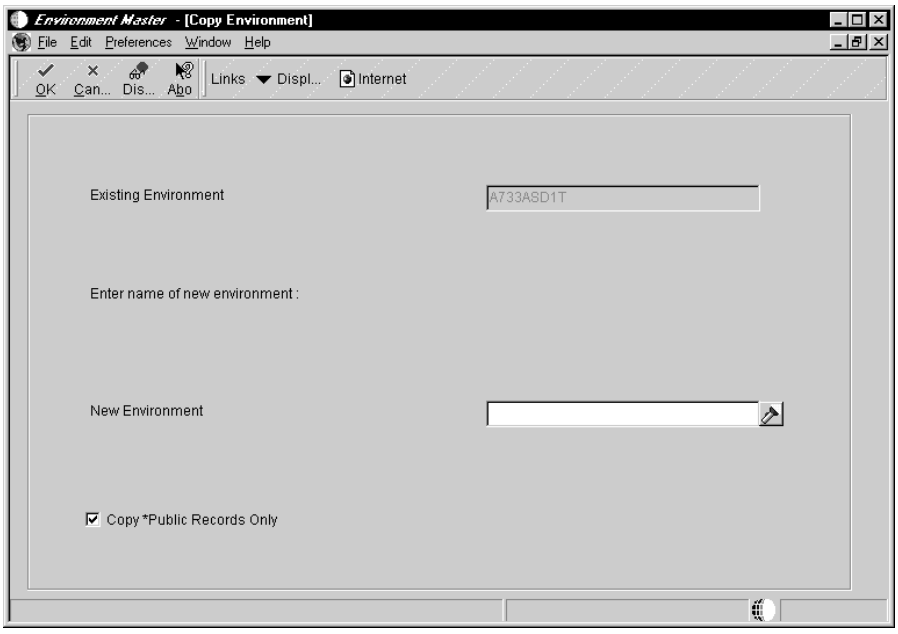
You can copy environments from your own existing environments, or you can use the Installation Planner, which provides sample production environments you can use as templates. These templates contain suggested mappings for all J.D. Edwards tables.

▶ To copy an environment

1. On the Environments menu (GH9053), choose Environment Master (P0094). The Work With Environments form appears.

   If the Display Only OneWorld Environments option is checked, this form displays only those environments that have an entry in the OneWorld-only environment detail file (F00941). If the option is not checked and you operate a coexistence enterprise, you will also see your AS/400 libraries.

2. Choose an environment, and choose Copy Environment from the Row menu.



3. On the Copy Environment form, type an environment name in the New Environment field.
4. To copy only the *PUBLIC Object Configuration Manager mappings of an environment, ensure that the Copy *PUBLIC Records Only option is turned on. Leave this option turned off to copy mappings for the environment, for individual objects, and for *PUBLIC.
5. Click OK.

   At this point, there will be a new environment in the Environment Master table (F0094). For that new environment, a record in the Environment Path Code Tag table (F00941) is created, as well as a set of mappings in the Object Configuration Manager table (F986101) in the system data source. Depending on your processing options, there might also be a set of mappings in server map data sources.

6. To change the path code for the newly created environment, select the environment from the Work With Environments form and change the path code.
7. Modify other Object Configuration Manager mappings if needed. For more information, see *Mapping Objects*.

Field	Explanation
New Environment	The name associated with a specific list of libraries. The J98INITA initial program uses these library list names to control environments that a user can sign on to. These configurations of library lists are maintained in the Library List Master table (F0094). For OneWorld, this field represents a valid environment that can be used to run OneWorld. The environment encompasses both a path code (objects) and a data source (data). When put together, users have a valid workplace within OneWorld.

Copying a New Environment to a New Path Code

The following task describes the typical scenario of how to use the copy feature to create a new test environment with demo data loading in a new path code.

This task is divided into the following procedures:

- Create test batch files
- Copy a new environment to a new path code
- Verify data sources and OCM mappings for your new path code
- Configuring ODBC drivers
- Update the AS/400 server map with the new environment

- Add the new path code on the AS/400

To create test batch files

1. Remark out all lines in the LOADALL and JDESET files and format a test set in the same format as the prod set.
2. Copy the LOADPROD to a LOADTEST.BAT file, and change the appropriate parameters to the ones you created in the JDESET.BAT file.
3. Run the LOADALL.BAT batch application.
4. Add the data sources you need in your current plan's planner environment. For logical data sources, you will probably want Business Data Test, Central Objects Test, OneWorld Local Test, and Control Tables Test.

   Make sure the deployment server source matches the Central Objects Data source and the LOADALL specifications you entered.

To copy a new environment to a new path code

1. Create your new path code as described in *Working with Path Codes*.
2. From the Environments menu (GH9053), choose Environment Master (P0094). The Work with Environments form appears.
3. Choose the environment that most closely matches the one you want, and choose Copy Environment from the Row menu. Make sure the option for *Public Records Only is set in the manner you want.
4. Click Close to go back to the Work with Environments form and click Find. Choose your new environment and change the path code to the new one you created in step 1. Click OK when you are finished.
5. Verify that the path code was changed by exiting to the Environments menu (GH9053) and choosing Environment Master (P0094) again.

To verify data sources and OCM mappings for your new path code

1. On the System Installation Tools menu (GH961), choose Custom Installation Plan (P98240).
2. Choose your plan and then choose Expand from the Row menu.
3. Highligh Data Sources: and click Select. The Data Source Planner form appears.
4. On Data Source Planner, double-click a data source name. The Data Source Revisions form appears.
5. On Data Source Revisions, choose Database Params from the Form menu. The Work With Table and Data Sizing form appears.

6. On Work with Tables and Data Source Sizing, choose Overrides from the Row menu. The Revise Table and Data Source Sizing form appears.
7. On Revise Table and Data Source Sizing, if you created the data sources recommended in the first task, set Create Tables to 1 and Detail Status to 10. Click Save, and then click Cancel.
8. Choose Environment from the Row menu. For the new test environment, set Load Demo Data to 2. Click Save.
9. Choose the new environment and choose Data Load from the Row menu. Verify that you are loading demo data, data source OneWorld Local Test. Click OK.
10. Return to the previous form and click Save.
11. Choose Environment from the Form menu and choose an environment.
12. Choose OCM from the Row menu. Make the appropriate OCM modifications, and then click Save. Click Cancel to exit the form.

▶ To configure ODBC drivers

1. On the System Installation Tools menu (GH961), choose Installation Workbench (P9841).
2. Find and select the plan.
3. Click Next until the Data Source Workbench form is displayed.
4. Choose Configure ODBC from the Row menu. Configure the ODBC for each of the four data sources.
5. Choose Configure from the Form menu and then press Next for the Environment Workbench. Choose the Configure option.
6. Click Next to continue through the rest of the Installation Workbench options.
7. Assign the new package to the user, and deploy the package to the workstation.

▶ To update the AS/400 server map with the new environment

1. On the System Installation Tools menu (GH961), choose Custom Installation Plan (P98240).
2. On the Work with Plans form, choose your plan and then choose Host from the Row menu.
3. Choose the host and choose Revise/Host from the Row menu.
4. Choose Environment from the Form menu, and then add the new environment to the grid. Save the environment you added.
5. When you are at the Host Planner form, choose Revise Host from the Row menu.

6. Choose Generate Svr. Map from the Form menu.

▶ To add the new path code on the AS/400

1. Modify the LIBRARY file in B733/hosts/as400/*machinename* so that it contains B7331SYS as the first entry. Add the name of each path code you would like to add, such as PD7333, PY7333, and so on.
2. Use the ENDNET and CLRIPC commands on the AS/400 to end B7331 services.
3. Sign on to the AS/400 as QSECOFR.
4. Save or rename the B7331SYS, B7331NET, and B7331CTL libraries so that you can restore them after the J98OW10 job runs.
5. Use the CLRLIB JDEOW command to clear the JDEOW library on the AS/400.
6. Use the CHGCURLIB JDEOW to change the current library on the AS/400 to JDEOW.
7. Use the FTP *deploymentservername* command to transfer files from your deployment server.
8. Sign on as JDE and enter the following commands:
   - `cd x:\jdedwardsoneworld\B733\hosts\as400 \`*`enterpriseservername`*
   - `get ftpinput (replace)`
   - `cd..`
   - `bin`
   - `get jdesav1 (replace)`
   - `quit`
9. Enter the following commands on the AS/400:
   - `RSTOBJ OBJ(J98OW10 J98OW20) SAVLIB (JDEOW) DEV(*SAVF) SAVF(JDEOW/JDESAV1) MBROPT(*ALL) ALWOBJDIF(*ALL) FRCOBJCVN(*YES *ALL) RSTLIB(JDEOW)`
   - `SBMJOB CMD(CALL PGM(J98OW10) PARM('`*`deploymentservername`*`' 'JDEOW'))`
10. After the J98OW10 job completes, replace the B7331SYS, B7331NET, and B7331CTL libraries with the ones you saved in step 4.
11. Start the OneWorld services with the STRNET command.
12. Run PORTTEST over each environment to verify that they are working.

Deleting an Environment

When you delete an environment, the environment definition is removed and the Object Configuration Manager records associated with the environment are deleted. You can set a processing option if you want to specify additional Object Configuration Manager mappings to delete when you delete the environment.

▶ To delete an environment

1. On the Environments menu (GH9053), choose Environment Master (P0094).
2. On the Work With Environments form, choose an environment, and click Delete.

Processing Options: Environment Master (P0094)

Process Tab

These processing options give you greater flexibility when copying or deleting by enabling you to specify whether users have the ability to copy or delete OCM mappings for this environment in other data sources.

Regardless of what you enter for these processing options, OneWorld automatically copies or deletes OCM mappings for the environment you are using. The processing option values you enter determine whether you can copy or delete mappings in other data sources.

1. Delete OCM Mappings

Use this processing option to indicate whether users can delete Object Configuration Manager mappings for this environment in other data sources. Valid values are:

Blank Users cannot delete OCM mappings.
1 Users can delete OCM mappings.

2. Copy OCM Mappings

Use this processing option to indicate whether users can copy Object Configuration Manager records for this environment in other server map data sources. Valid values are:

Blank Users can copy OCM mappings.
1 Users cannot copy OCM mappings.

Data Sources

OneWorld data sources are the building blocks that you use to set up a OneWorld enterprise configuration. Data sources define to OneWorld all the databases and logic machines required by the OneWorld configuration.

This section contains the following:

- ❑ Understanding data sources
- ❑ Planning OneWorld data sources
- ❑ Working with data sources
- ❑ Running data source reports

Understanding Data Sources

OneWorld data sources define where your database tables reside and where OneWorld runs logic objects for your enterprise. Data sources can point to:

- A database in a specific location (for example, a Microsoft Access database, such as JDEB7.MDB located in \B7\DATA, or an AS/400 data library, such as PRODDATA)
- A specific machine in the enterprise that processes logic

Data source definitions are stored in the Data Source Master table (F98611). Workstations use a common Data Source Master table, which generally resides in the system data source on the enterprise server. OneWorld servers that process logic and request data require their own unique definitions for OneWorld data sources; therefore, they have their own Data Source Master in the server map data source.

There are at least two sets of the Data Source Master (F98611) tables. They reside in a centralized system data source normally kept on an enterprise server and accessed by workstations, and in a server map data source, which each logic server requires.

This chapter describes the following:

- ❑ Understanding database data sources
- ❑ Understanding logic data sources
- ❑ Understanding the required data source types

Understanding Database Data Sources

A database is a grouping of tables in a database management system. You must identify databases to applications that access them. You can distribute databases across a network and involve various servers and database management systems. A database data source identifies to OneWorld the database information OneWorld needs to identify and connect to a database.

The following identifiers are important when defining a database data source:

- Data source name
- Database name
- Table owner
- Network machine name
- JDEBase program

Data Source Names

You define names to identify the data source. You should use a meaningful name for your data sources. For example, to indicate that you are storing business data for production users, your data source name could be: Business Data - Prod, or something similar.

J.D. Edwards provides demonstration data source names at installation, which you can use for your own data sources. See the *OneWorld Installation Guide* and *OneWorld Installation Reference Guide* for more information.

Database Names

The data source definition must contain information about the database and the server in which it is located. Different database management systems identify the databases in different ways. For example, you must identify Oracle databases by the Oracle SQL*Net V.2 connect string. You must identify databases that you access through ODBC by the ODBC data source name.

The Table Owner (Object Owner ID)

Oracle and SQL Server database management systems have unique table owner IDs for each group of tables. For example, the database that contains the system tables might have an owner such as B733SYS. You must identify the table owner ID for Oracle and SQL Server database management systems.

The Network Machine Name (Server Name)

Database management systems reside on a machine. You must identify this machine to the network so that other computers can access its resources. You must provide to OneWorld (in the data source definition) the machine name for the server that hosts the database management system in which the database resides.

JDEBase (DLL Name)

J.D. Edwards buffers the business solution from the technology. The JDEBase middleware product provides a set of APIs that resolve SQL differences across multi-vendor database management systems. You must associate the proper JDEBase program with the data source definition. For example, you must use JDBODBC.DLL with Microsoft SQL Server, DB2/400, and Microsoft Access. These programs translate the generic Data Manipulation Language (DML) requests into appropriate SQL statements and work with the third-party communication software to communicate requests across the network.

Understanding Logic Data Sources

A logic machine is the machine on which batch applications and master business functions run. You must identify these logic machines to OneWorld using a data source definition. The data source definition must include the network information about the machine, such as a server name like HP9000.

When mapping logic objects for distributed processing, OneWorld uses the machine data source (distributed processing data source) as the target location for processing logic objects.

Understanding the Required Data Source Types

You must set up a minimum number of data sources for OneWorld to run. Two of the required data sources define machines that process logic in the enterprise. The other data sources define various databases used in the enterprise.

You should understand the following required data sources:

- Distributed Processing (logic)
- Local (logic)
- Business Data (database)
- Control Tables (database)
- Replicated Local Data (database)
- Data Dictionary (database)
- System (database)
- Object Librarian (database)
- Central Objects (database)
- Server Map (database)
- Versions (database)
- AS/400 data sources (databases)

The OneWorld installation software provides samples of these required data sources to build your system configuration. See the *OneWorld Installation Guide* for details.

Distributed Processing Data Source

This data source definition contains information that OneWorld uses to identify the logic machine in the network. You need to define each logic machine as a data source.

Local Data Source

This data source defines the OneWorld workstation. Use this data source to override the process location of a batch application that you mapped in the Object Configuration Manager to run on the server.

Business Data Data Source

This data source is used when you divide your business data into multiple owners or libraries, which can reside on the same enterprise server or on different ones. Each group of data requires a separate data source.

Some examples of business data include:

- Production data (nontechnical data, such as financial and manufacturing data)
- Test data
- Demo data (demonstration or training data)
- Conference Room Pilot (CRP) data

The OneWorld installation procedure provides demonstration data that you can copy to supported host databases. The data source name is Business Data - JDE.

Control Table Data Sources

This data source consists of user defined codes, menus, and next numbers.

Replicated Local Data Data Source

This data source points to the Microsoft Access database you use for edits made on the workstation. The amount of replicated local data depends on the transaction model you choose (for example, store and forward rather than direct connect). The user defined codes and menu tables should reside in a Microsoft Access database file on the workstation. Replicated local data is by path code and stored on the workstation in the following path: b7/*path code*/data.

Data Dictionary by Release Data Source

This data source allows you to store OneWorld data dictionary master tables in a central location to allow easier administration of changes. Group these master tables together to form a data dictionary database. You should share one data dictionary between your production (such as PD7333) and development (such as DV7333) path codes. OneWorld allows one data dictionary per path code, but multiple data dictionaries are not recommended or supported. The Data Dictionary data source is named by base release number, for example, Data Dictionary - B733 or Data Dictionary - B732.

System Data Source

This data source consists of the technical tables you use to run all OneWorld applications. You must set up one system data source per release. When running applications, the system tables provide the following:

- Environment detail (You should map these to the AS/400 for coexistence with WorldSoftware.)
- User profiles and preferences (You should map these to the AS/400 for coexistence with WorldSoftware.)
- Object mappings (location of tables, batch processes, and business functions)
- Data source definitions
- OneWorld security
- Next IDs (used for development only)

All workstations use a central set of system tables usually stored on your enterprise server, but not on your deployment server. Each logic server requires its own subset of system tables. These server system tables are stored in the server map data source. See *The Server Map Data Source*.

Understanding System Table Caching

When a user firsts signs on, OneWorld uses the user ID and environment to retrieve information from the system tables for that user and environment. OneWorld caches this information in memory on the workstation. Any time a change is made to the central system tables, dynamic caching of the system information occurs for those workstations with an active OneWorld session.

How OneWorld Connects to the System Data Source

When OneWorld starts on a workstation, OneWorld attempts to connect to the base data source found in the workstation jde.ini file. If this data source is unavailable, OneWorld attempts to connect to a secondary data source for system information. It is important to have processes for ensuring that the alternate system data source location contains current information. You can

maintain an alternate data source's information using table conversion or data replication.

The jde.ini file should look like the following for the primary system data source connection:

```
[DB SYSTEM SETTINGS]

.
.
Default Env=DEMOB7A
Default PathCode=DEMO
Base Datasource=System B733
Database=System B733
.
.
.
Secondary System Data Source connection

[DB SYSTEM SETTINGS - SECONDARY]
Base Datasource=Access32
Object Owner=
Server=
Database=Access32
Load Library=JDBODBC.DLL
```

During installation, the system data source is tied to a release with the Release Master application. Configuring the release updates the setup.inf file used during the workstation install to create the jde.ini file.

See Also

- *Storing Object Librarian and Central Objects*
- *Major OneWorld Technical Tables* in the *OneWorld Installation Reference Guide*

Object Librarian Data Source

This data source points to the Object Librarian tables you use for custom development. You should have only one set of Object Librarian tables per software release, regardless of how many path codes (sets of central objects) you maintain. This data source can reside on any J.D. Edwards-supported platform. The Object Librarian data source is named by base release number, for example, Object Librarian - B733.

Central Objects Data Source

This data source points to the OneWorld source objects (central object specifications), as well as the User Overrides table (F98950). Central Object data sources are databases.

If you have multiple path codes, each must have a separate Central Objects data source. Developers check objects out of a Central Objects data source for modification. When the developer checks in the objects, the system copies the objects from the developer's workstation to the relational database tables in the Central Objects data source. You must set up one Central Objects data source for every path code needed in your configuration, for example, Central Objects - PD7333 or Central Objects - DV7333.

For example, you must have a Central Objects data source for:

- Pristine objects
- Production objects
- Development objects

You connect each Central Objects data source to a path code used by the environments that you created for your configuration. This data source must reside in a SQL server or Oracle database, and can reside on any J.D. Edwards-supported platform that can run those databases. For performance and table space reasons you should keep your central objects in Oracle UNIX. If your configuration does not include a UNIX enterprise server, you should use Microsoft SQL Server or Oracle on the deployment server.

Storing Object Librarian and Central Objects

The Object Librarian and the Central Object data sources can reside on either Oracle Server or Microsoft SQL Server. The Object Librarian can also reside in DB2/400 and can be on a machine separate from the Central Objects data source (which resides on Oracle or Microsoft SQL Server).

J.D. Edwards recommends:

- For businesses with an Oracle/UNIX platform, store the Object Librarian and the Central Object data sources in Oracle. Otherwise, store them on your deployment server's Oracle Server or Microsoft SQL Server database.
- Regardless of which database is storing the relational database (RDB) components of the central objects, you must store the C programming language components on the deployment server that is accessible to all workstations for deployment and redeployment. Therefore, J.D. Edwards recommends storing C components the deployment server.

Server Map Data Source

This data source enables you to create for each logic server its own subset of system tables called server map tables. Server map tables are required for each logic server. You must maintain these tables to ensure integrity with the workstation's system tables.

Use Server Map data sources to establish unique object mappings for logic servers. When batch jobs and business functions running on the server request data, they look to the Object Configuration Manager and the Data Source Master tables in the server map data source. This is necessary because the mappings are different.

For example, suppose a user signs on to an environment that maps static local data on the workstation, dynamic transaction data to the server, and the master business functions and batch processes to the server. The user enters a sales order. The user edits user defined codes, tax rules, and other static data against the workstation's replicated data. The user clicks OK to enter the order, which runs the Sales Order Entry master business function on the server. The master business function also needs transaction data and some of the replicated data. It does not make sense for the master business function to go back to the workstation to retrieve user defined codes and tax information; therefore, the server map Object Configuration Manager table maps all data to the appropriate server data source. More than likely, the server user defined codes, tax rules, and so on, are the published tables from which the workstation's tables are replicated. See *Data Replication* in the *System Administration Guide* for information about replicating tables.

The following tables in the Server Map database are unique to a server's perspective of processing:

Data Source Master (F98611)

To enable servers to request data from other servers, you must assign the proper JDEBase program to the data source definition in the Data Source Master file in the Server Map data source.

For example, when a workstation requests data from an Oracle database, the data source definition for the Oracle database uses JDBOCI32.DLL. The third-party product called Oracle Call Level Interface (OCI) performs the communications between workstation and server. When an AS/400 requests data from an Oracle database, the data source definition for that database should use JDBNet, which manages communications.

Object Configuration Manager (F986101)	Logic objects processing on a server request data and perhaps other logic objects. When these requests are made to OneWorld running on a server, Object Configuration Manager must be accessed to find the correct mappings for the data and logic objects. Servers might have different mapping requirements than workstations. For example, you should map all user defined codes locally to the workstation for performance during interactive processing. Server processing would require you to map these files locally to a server database to enhance performance of server processing. See *Working with the Object Configuration Manager* for more information.
Job Control Status Master (F986110)	Records information about batch jobs launched on a server.
Job Number Master (F986111)	Records next numbers for batch jobs launched on a server.
Data Replication Change Log (F98DRLOG)	Stores one record per data change and stores the actual changed data. Each server that has publisher tables must have this table in its Server Map data source.
Data Replication Pending Change Notifications (F98DRPCN)	Stores one record per data change per subscriber. This log records unprocessed change deliveries so that OneWorld can know which subscribers have not yet received the change deliveries. Each server that has publisher tables must have this table in its Server Map data source.

Versions Data Source

This data source corresponds to the path code, as in Versions - PD7333. It stores versions and processing option information. It includes the following tables:

- Versions List (F983051)
- Processing Option Text (F98306)

AS/400 Data Sources

Additional data sources are required for users with coexistence configurations who have an AS/400 enterprise server. You must identify these data sources as Do Not Translate (DNT) because they contain OneWorld tables with binary large objects (BLOBs). BLOBs are used by OneWorld for security and performance purposes. They cannot be translated into a form directly readable by a database other than OneWorld.

The data source master definition for DNT data sources must have the AS/400 BLOB Data Source field checked. The following are the data sources:

- System - DNT. This data source stores the following nontranslated tables:
  - OneWorld Security table (F98OWSEC) for sign on security
  - Data Replication Change Log (F98DRLOG)
  - Data Replication Pending Change Notifications log (F98DRPCN)
- Versions - DNT. This data source stores, by path code, the Versions List table (F983051).
- Business Data - DNT. This data source corresponds to the environment, and stores the Workflow Activity Specifications table (F98811).

Planning OneWorld Data Sources

Because OneWorld data sources are the building blocks of your OneWorld configuration, proper identification of all required OneWorld data sources is critical to having a correct configuration to support your business needs.

The following data source types require just one data source definition for each release:

- Object Librarian
- System
- Replicated Local Data (except for store and forward processing, in which you have a separate Microsoft Access database for those store and forward users who need additional replicated data)
- Data Dictionary
- Local

Other data source types might require you to define multiple data sources for each type to support your business requirements:

- Business Data (production, test, CRP, and JDE demo data)
- Distributed processing (one per logic server)
- Server Map (one per logic server)
- Central Objects (one per path code)
- Control Tables

To plan for OneWorld data sources, do the following:

1. Evaluate the location of your system data.
2. Identify all the enterprise servers in your configuration that will host logic:
    - Set up one distributed processing data source for each enterprise logic server
    - Set up one Server Map data source for each distributed processing data source
3. Evaluate how many groups of OneWorld objects you require to support your business environment, such as production, pristine, and development. You need to set up one Central Objects data source for each group of objects.

4. Evaluate the various locations of your business data:

   - Set up one Business Data data source for each database and library that contains Business Data tables that OneWorld applications use
   - Set up a Control Table data source for production and one for testing
   - Set up a Replicated Local Data data source for at least menu tables. Other business decisions will require you to replicate data to the local workstation database.

Working with Data Sources

The database data sources and logical data sources applications let you add or modify a database data source or logic data source, respectively. After you add a new data source you must update the Release/Data Source Map table (F00948) by using the Release/Data Source Map application.

This topic contains the following:

- ❑ Data source naming conventions
- ❑ Adding or modifying a data source
- ❑ Modifying the Release/Data Source Map table
- ❑ Processing Options: Database Data Sources and Logical Data Sources (P986115)

Data Source Naming Conventions

When you add data sources, be sure to observe the following naming conventions:

- Limited to 30 characters
- Case sensitive
- Space sensitive

Specific exceptions for the Client Access data source are as follows:

Data Source Name:

- Limited to 32 characters
- Must begin with an alphabetic character
- You cannot use the following characters: { } [ ] ( ) ? * = ! @ ;

Note: You must type the data source name before you can use the Client Access ODBC driver to access AS/400 data.

Data Source Description:

- Limited to 80 characters

Adding or Modifying a Data Source

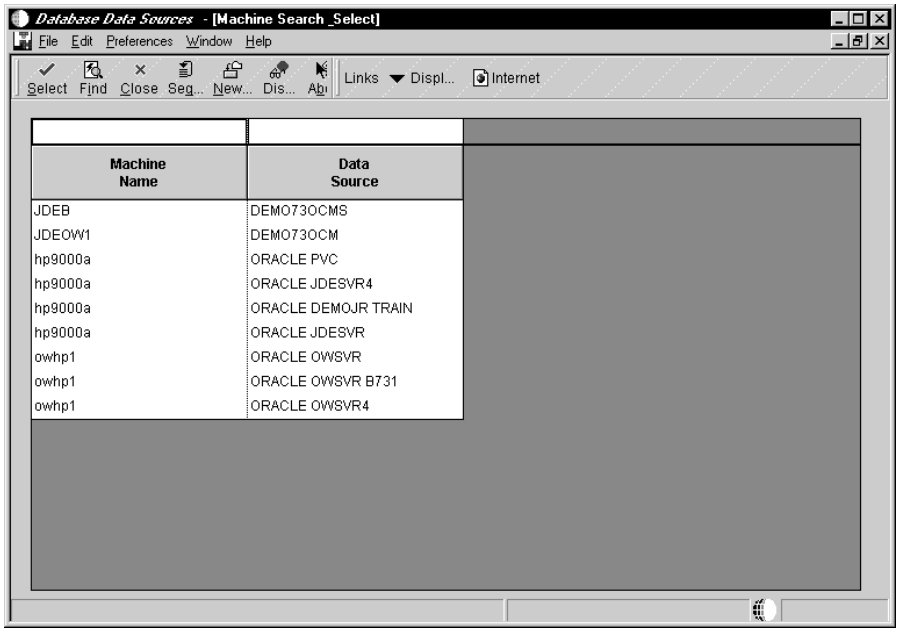
This task explains how to add a database or logic data source to a Data Source Master table (F98611) in the system data source, or to modify an existing data source. You use this application to modify or to add a data source definition after initial OneWorld installation.

As part of the add or modify process, you can also enter or modify table and data source sizing and override information.

▶ To add or modify a data source

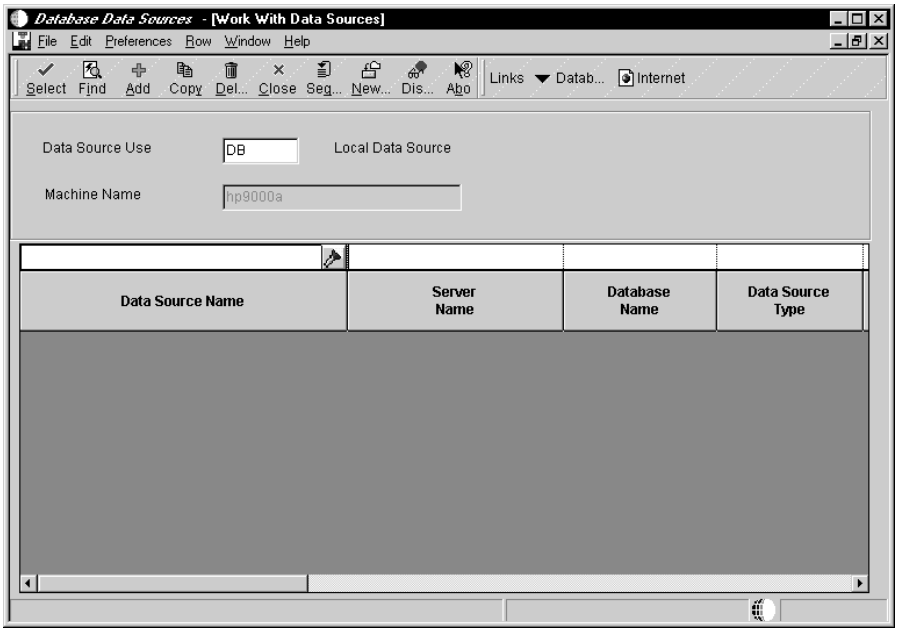
On the System Administration Tools menu (GH9011), choose Database Data Sources (P986115) to add or modify a database data source, or choose Logical Data Sources to add or modify a logic data source.

The Machine Search & Select form appears. It displays those data sources that have the OCM Data Source field checked on the Data Source Revisions form.



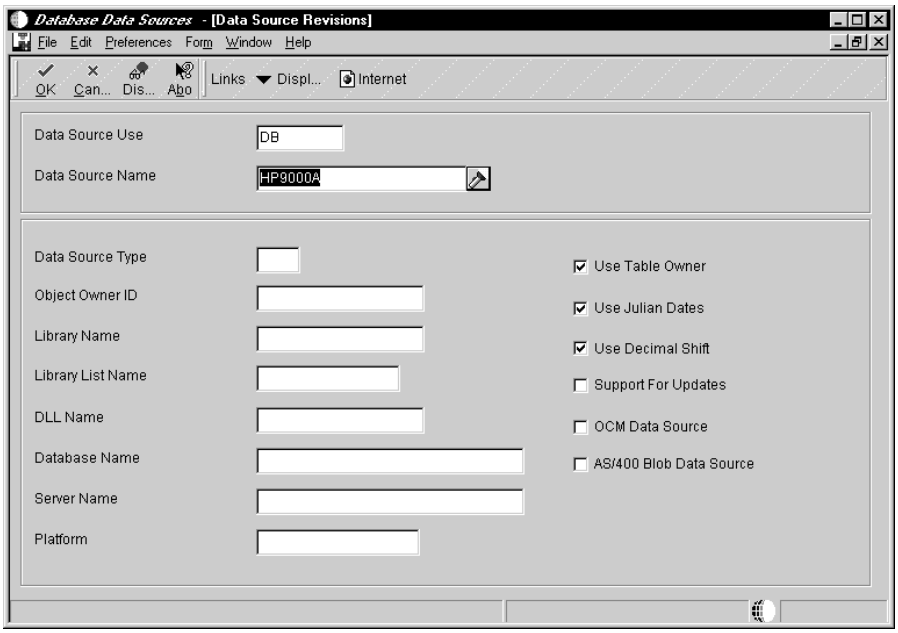
On the Machine Search & Select form, choose the data source you want to work with, and click Select. Your selection should be the machine that contains the data source you want to view. Workstations use the system

data source for their mappings. Each enterprise server has its own server map data source.



If you want to add a data source, click Add on the Work With Data Sources form. The Data Source Revisions form appears.

If you want to modify an existing data source, click Find on the Work With Data Sources form to display a list of data sources. Locate the data source you want to modify and click Select. The Data Source Revisions form appears with information for the data source.



Caution: If you want to only view your data sources without adding or changing one, make sure you exit from the Data Source Revisions form by clicking Cancel instead of clicking OK. When you click OK OneWorld assumes you have added or changed a data source, and your existing ODBC drivers may not work correctly. You will have to modify them using the ODBC Data Source Administrator applet accessible from the Control Panel.

On the Data Source Revisions form, complete or modify the following fields, and then click OK:

- Data Source Name
- Data Source Type
- Object Owner ID
- Library Name
- Library List Name
- DLL Name

  For a list of DLL names, refer to the online help or see *Database Middleware Drivers*.

- Database Name
- Server Name
- Platform

If you are creating a logic data source, be sure to enter the Database Name, Object Owner ID, DLL Name, and Data Source Type fields. These fields provide information about the associated server map and are all required for accessing a logic server through an interactive application.

Choose any of the following options:

- Use Table Owner
- Use Julian Dates
- Use Decimal Shift
- Support For Updates
- OCM Data Source
- AS/400 Blob Data Source

After you click OK, the ODBC Data Source Administrator applet that allows you to enter or change ODBC drivers opens automatically. Add or update ODBC drivers to reflect the data source you just added or changed.

Field	Explanation
Data Source	The data source name.

Field	Explanation
Data Source Use	Indicates how the data source is configured, Servers (SVR) to run UBE's and Business Functions, or a Database (DB) to access table data.
Data Source Type	The type of database.
Object Owner ID	The database table prefix or owner.
Library Name	These are the names of the libraries that contain J.D. Edwards files and programs. Each file should exist in only one library in the environment other than the 'JDF' library (usually JDFDATA). You may, however, enter multiple 'COM' or 'DTA' libraries. COM Common Files (for example, DREAMWriter and Data Dictionary) CSO Custom executable programs CSS Custom source library DTA Data files (for example, F0411 and F0911) JDF J.D. Edwards data (for example, JDFDATA) KBG World Case files OBJ JDE executable programs (for example, JDFOBJ) SEC Security files (for example, F0092, F0094) SRC JDE source code (for example, JDFSRC)
Library List Name	The name associated with a specific list of libraries. The J98INITA initial program uses these library list names to control environments that a user can sign on to. These configurations of library lists are maintained in the Library List Master table (F0094). For OneWorld, this field represents a valid environment that can be used to run OneWorld. The environment encompasses both a path code (objects) and a data source (data). When put together, users have a valid workplace within OneWorld.

Field	Explanation
DLL Name	Specifies the name of the database driver file. This file is specified in the [DB SYSTEM SETTINGS] section of the enterprise server jde.ini file. The file you specify depends upon the platform and the database. Values for specific machines and databases are: AS/400 to DB2/400 = DBDR AS/400 to any other server DBMS = JDBNET HP9000 to DB2/400 = libjdbnet.sl HP9000 to Microsoft SQL Server = libjdbnet.sl HP9000 to Oracle (Version 8.0) UNIX = libora80.sl RS6000 to DB2/400 = libjdbnet.so RS6000 to Microsoft SQL Server = libjdbnet.so RS6000 to Oracle (Version 8.0) UNIX = libora80.so Intel to AS/400 = jdbodbc.dll Intel to Oracle (Version 8.0) NT = jdboci80.dll Intel to SQL Server NT = jdbodbc.dll Digital Alpha to AS/400 = jdbnet.dll Digital Alpha to Oracle (Version 8.0) NT = jdboci80.dll Digital Alpha to SQL Server NT = jdbodbc.dll
Database Name	The name assigned to the database during installation, such as HPDEVORAP or HP9000.
Server Name	The name of the network server where data resides or where objects can be executed.
Platform	The type of physical hardware the database resides on.
Use Table Owner	This activates use of the Owner ID field.
Use Julian Dates	Check this to store dates in a Julian format. Otherwise, dates are stored as defined in the Microsoft Windows Control Panel. OneWorld automatically turns this flag OFF for non-OneWorld tables.
Use Decimal Shift	Check this to automatically shift decimals when retrieving or updating data based on specifications in the data dictionary. This field is for OneWorld tables only. Non-Oneworld tables should be in a separate data source with decimal shift turned OFF. If you bring in a OneWorld table as non-OneWorld table, OneWorld will not recognizethe fields in this table that have been decimal shifted. This is not checked in Table Conversion.
Support For Updates	Use this flag to turn ON or OFF row-level record locking for the data source. You should have this flag turned ON to help prevent database integrity issues. JDEBASE middleware uses this flag to determine whether or not to use row-level record locking.

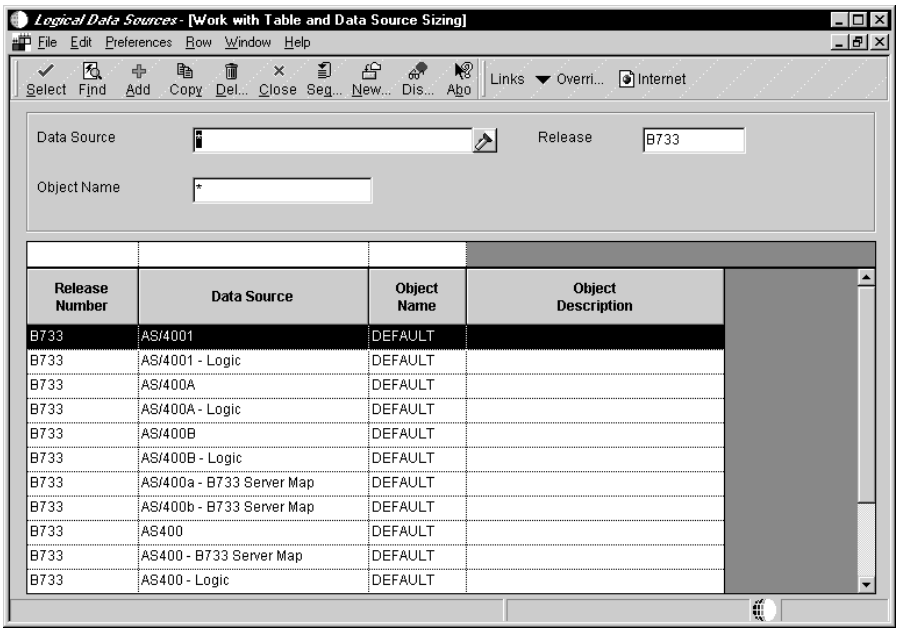
Field	Explanation
OCM Data Source	Use this flag to specify whether the data source will contain an OCM table (F986101). If you enable this flag, OneWorld displays a 1 in the OCM Data Source field on theWork With Data Sources form. You should only set this flag for the system data source and any server map data sources. That is, you should not enable this setting for any database data source or log data sources that are not system or server map data sources.
AS/400 Blob Data Source	The setting for the AS400 Binary Object (BLOB) Datasource indicates whether a datasource may contain OneWorld BLOB Tables.

▶ To enter or modify table and data source sizing or override information

On the System Administration Tools menu (GH9011), choose Database Data Sources or Logical Data Sources (P986115).

On the Machine Search & Select form, find the data source you want to work with, and click Select.

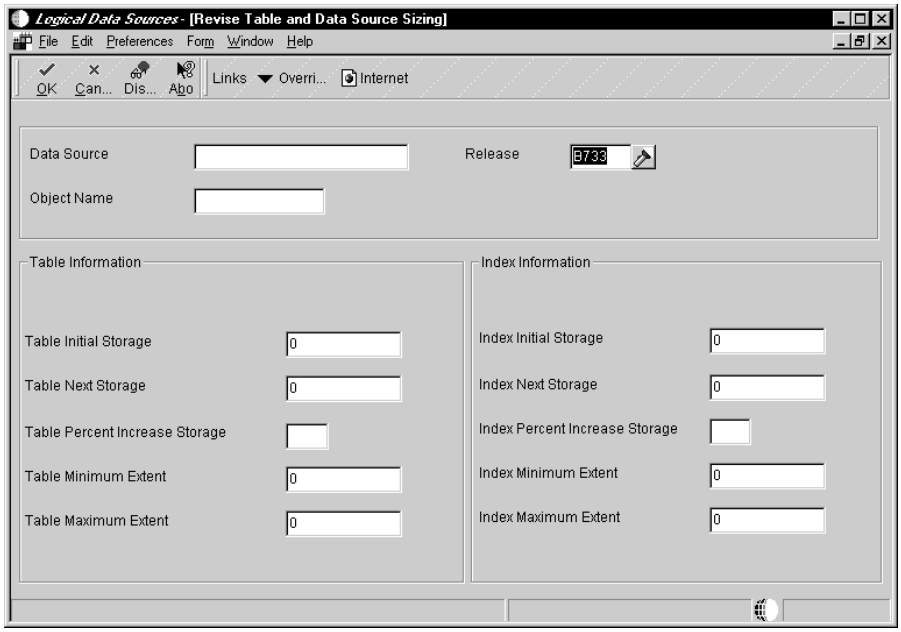
On the Work With Data Sources form, choose Database Params from the Row menu. The Work With Table and Data Source Sizing form appears.



Note: You can also access this application from the Object Configuration Manager application.

To add table and data source sizing information, click Add on the Work With Table and Data Source Sizing form. The Revise Table and Data Source Sizing form appears. On this form you can revise or add table and index information for a data source or table.

To modify existing table and data source sizing information, on the Work With Table and Data Source Sizing form, find and select the desired data source. The Revise Table and Data Source Sizing form appears with previously entered information.



Note: Alternatively, you can display the Revise Table and Data Source Sizing form by choosing Default Database from the Work With Data Sources form's Row menu. When you do so, the DEFAULT value is automatically entered at the Object Name field so that default values are used for the form. If you change the object name from DEFAULT, the fields that were formerly filled with default values on the form appear.

If you are adding new parameters rather than modifying existing ones, complete the following fields:

- Data Source
- Object Name

Note: You can define parameters for all objects in a data source by entering DEFAULT at the Object Name field.

On the Revise Table and Data Source Sizing form, complete or modify the following fields:

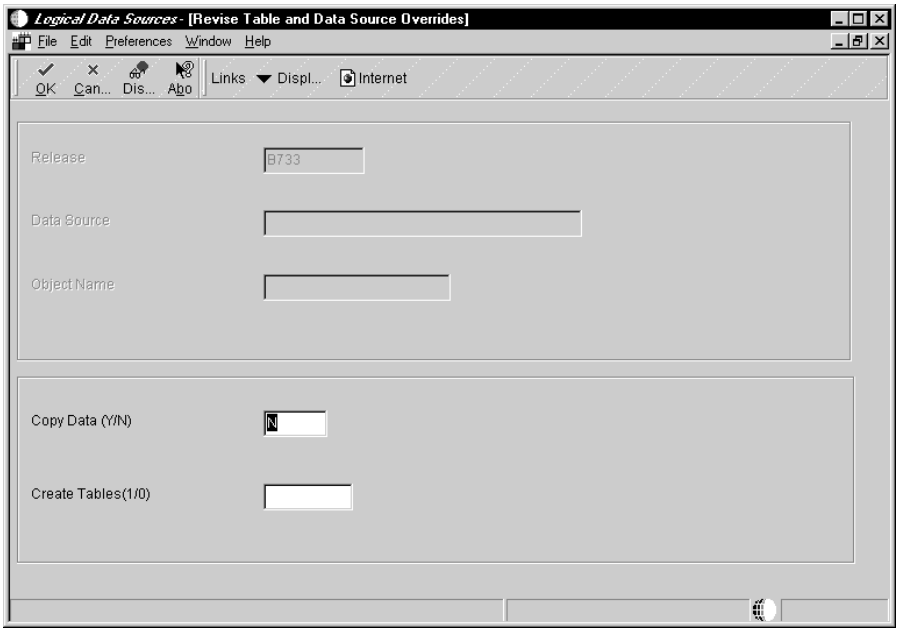
- Initial Storage
- Next Storage

- Percent Increase Storage
- Minimum Extent
- Maximum Extent

These values are valid for both the table information and the index information.

Note: If the object name is DEFAULT, the only fields you can enter or modify on the Revise Table and Data Source Sizing form are Table Space Name and Index Space Name.

To enter override information, choose Overrides from the Revise Table and Data Source Sizing form's Form menu. This option is also available from the Work With Table and Data Source Sizing form's Row menu. The Revise Table and Data Source Overrides form appears.



On the Revise Table and Data Source Overrides form, complete or modify the following fields:

- Copy Data (Y/N)
- Create Tables (1/0)

Click OK to return to the previous form. If necessary, click OK on all subsequent forms until you return to the System Administration Tools menu.

Database Middleware Drivers

The following table lists the database middleware drivers supplied with OneWorld. When adding a data source, choose the driver appropriate for your

platform. For more information about how to add or modify a data source, see *Adding or Modifying a Data Source*.

HP 9000 drivers	HP9000 to DB2/400 = libjdbnet.sl
	HP9000 to Microsoft SQL Server = libjdbnet.sl
	HP9000 to Oracle (Version 8.0) UNIX = libora80.sl
RS 6000 drivers	RS6000 to DB2/400 = libjdbnet.so
	RS6000 to Microsoft SQL Server = libjdbnet.so
	RS6000 to Oracle (Version 8.0) UNIX = libora80.so
Intel NT drivers	Intel to AS/400 = jdbodbc.dll
	Intel to Oracle (Version 8.0) NT = jdboci80.dll
	Intel to SQL Server NT = jdbodbc.dll
Digital Alpha NT drivers	Digital Alpha to AS/400 = jdbnet.dll
	Digital Alpha to Oracle (Version 8.0) NT = jdboci80.dll
	Digital Alpha to SQL Server NT = jdbodbc.dll
AS/400 drivers	AS/400 to DB2/400 = DBDR
	AS/400 to any other server DBMS = JDBNET
Sun Solaris drivers	Sun Solaris to DB2/400 = libjdbnet.so Sun Solaris to Microsoft SQL Server = libjdbnet.so Sun Solaris to Oracle (Version 8.x) UNIX = libora80.so

Processing Options: Database Data Sources and Logical Data Sources (P986115)

Defaults Tab

These processing options enable you to do the following:

- Specify a local (database) data source or a server data source

- Indicate whether you want to create an ODBC data source when the Workstation Installation program runs

1. Data Source Type

Indicates how the data source is configured, Servers (SVR) to run UBE's and Business Functions, or a Database (DB) to access table data.

2. ODBC Data Sources

Use this processing option to specify whether to create an ODBC data source during the workstation installation process. Valid values are:

- Blank: Do not create ODBC data sources.
- 1: Create Client Access, SQL Server and DB/2 data sources.

Modifying the Release/Data Source Map Table

When you add a new data source, you need to update the Release/Data Source Map table (F00948). This table contains entries for the data dictionary, Object Librarian, and versions list and organizes the entries according to release. For example, you should have separate data dictionary entries for releases B73.1, B73.2, and B73.3.

When you copy or transfer objects between data sources, OneWorld checks the release of each data source and then determines whether the source data source and the target data source reside in compatible releases. OneWorld does not allow you to move or copy objects between data sources with incompatible releases.

To modify the Release/Data Source Map table, complete the following tasks:

- View a record in the Release/Data Source Map table
- Add a record to the Release/Data Source Map table
- Delete a record from the Release/Data Source Map table

▶ To view a record in the Release/Data Source Map table

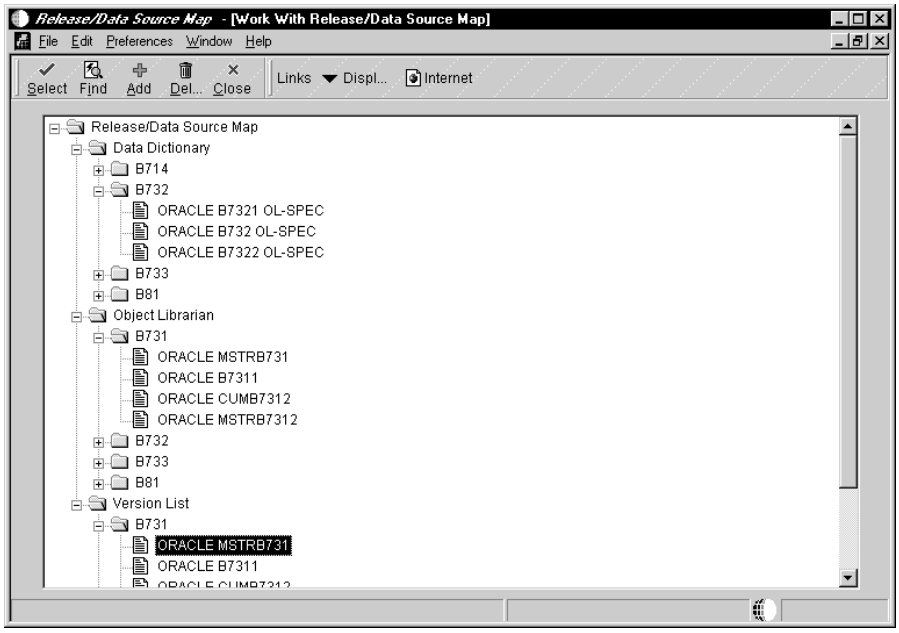
From the Environments menu (GH9053), choose Release/Data Source Map (P00948).

The Work with Release/Data Source Map form appears. On this form, you can view a list of data sources organized by data source type (data dictionary, object librarian, versions list) and release (for example, B73.3).

You can also add and delete data sources. Although you can view a record for an existing data source, you cannot modify the record.

Click Find to view the contents of the Release/Data Source Map table (F00948).

The Release/Data Source Map application organizes data sources into a tree structure first by type and then by release.



Open the appropriate nodes of the tree structure and double-click a record to view the information for that record.

▶ To add a record to the Release/Data Source Map table

On the Environments menu (GH9053), choose Release/Data Source Map (P00948).

The Work with Release/Data Source Map form appears. On this form you can view a list of data sources organized by data source type (data dictionary, object librarian, versions list) and release (for example, B73.3). You can also add and delete data sources. Although you can view a record for an existing data source, you cannot modify the record.

Click Add.

The Release/Data Source Map Revisions form appears. On this form you define the data source and the release.

Enter the following information and click OK:

- Data Source Type
- Release/Data Source

  Enter the name of the data source, such as Data Dictionary - B7333**.**

- Release

Click Cancel to return to the Work with Release/Data Source Map form.

Note: The new record does not appear in the tree structure on Work with Release/Data Source Map until you click Find.

▶ To delete a record in the Release/Data Source Map table

From the Environments menu (GH9053), choose Release/Data Source Map (P00948).

The Work with Release/Data Source Map form appears. On this form you can view a list of data sources organized by data source type (data dictionary, object librarian, versions list) and release (for example, B73.3). You can also add and delete data sources. Although you can view a record for an existing data source, you cannot modify the record.

Click Find to view the contents of the Release Data Source Map table (F00948).

The Release/Data Source Map application organizes data sources into a tree structure by type then by release.

Select the record you want to remove from the table, and then click Delete.

On Confirm Delete, click OK.

Running Data Source Reports

The Data Source Master batch processes enable you to create a report for a data source or a comparison report for two data sources.

The Data Source Master batch processes are called from a report driver (R98611B), whereas other OneWorld reports do not require a report driver.

You use the R98611B report driver to set processing options for each report. Any data selection for a report must be entered for the specific report, not from the driver. When you finish entering processing options and data selection, if any, you run the report from the driver.

This chapter contains the following:

- Setting processing options for data source reports
- Processing options: Data Source Master (R98611) and Data Source Master Comparison (R986112)
- Running the Data Source Master Comparison Report
- Running the Data Source Master Report

Setting Processing Options for Data Source Reports

The following task describes how to set processing options for a data source report through the report driver.

▶ Setting processing options for data source reports

From the Advanced Operations menu (GH9012), choose Data Source Master Report or Data Source Master Comparison. The Work with Batch Versions - Available Versions form appears. (The same form appears regardless of which report you choose.)

Select one of the reports (it doesn't matter which one) and then choose Processing Options from the Row menu. The Processing Options form appears.

On the form's UBE Driver tab, enter information for the report driver. For more information about entering the fields on this tab, see *Processing options: Data Source Master (R98611) and Data Source Master Comparison (R986112)*

Click the tab for the report whose processing options you want to enter. For more information about entering the fields on these tabs, see *Processing options: Data Source Master (R98611) and Data Source Master Comparison (R986112)*.

Click OK to save and exit to the Work with Batch Versions - Available Versions form.

Processing Options: Data Source Master (R98611) and Data Source Master Comparison (R986112)

The processing options that are set from the report driver enable you to enter parameters for the Data Source Master Report (R98611) and the Data Source Master Comparison Report (R986112).

The processing options for the report you are running must be completed. Because the XJDE0001 version of the report runs, if data selection is necessary it must be done on the XJDE0001 version of the report.

UBE Driver Tab

This processing option lets you specify the Data Source Master report you want to run. Verify that the displayed report number corresponds to the report that you want to run. You can also specify the version you want to use for that report.

1. Data Source Master Report Name

Use this processing option to specify the name of the report you want to run. Valid values are:

R98611 Data Source Master Report
R986112 Data Source Master Compare

2. Version

Use this processing option to specify the version that you want to use for the report.

R986112 Tab

This processing option lets you specify the two data sources to use in the comparison, the comparison method, and how to handle exception printing.

1. Data Source One and 2. Comparison Data Source Two

Use this processing option to specify the names of the data sources that you want to compare.

3. Comparison Method

Use this processing option to specify the comparison method used. The comparison method determines how OneWorld searches the data sources. When you compare in one direction, OneWorld lists records only in data source 1 that are not in data source 2. When you compare in both directions, OneWorld lists records that are in data source 1 but not in data source 2, AND records in data source 2 that are not in data source 1.

Valid values are:

1 Compare one direction only. Print only the records found in the first data source, but not in the second data source.

Blank Compare both directions. Print records in the first data source and also the records in the second data source that don't exist in the first data source.

4. Exceptions Only

Use this processing option to specify whether to print only the report exceptions or to print all records on the comparison report. Valid values are:

1 Print only the records that are different between the first and second data sources.

Blank Print all records from both data sources.

R98611 Tab

This processing option lets you specify the data source to use for the report.

1. Data Source Name

Use this processing option to specify the data source that you want to use to produce the report.

Running the Data Source Master Comparison Report

Run the Data Source Master Comparison (R986112) report to compare Data Source Master (F98611) tables between two data sources. For example, you might compare the F98611 table in the system data source to the F98611 table for a given server map data source. Data sources should always be defined differently on each machine.

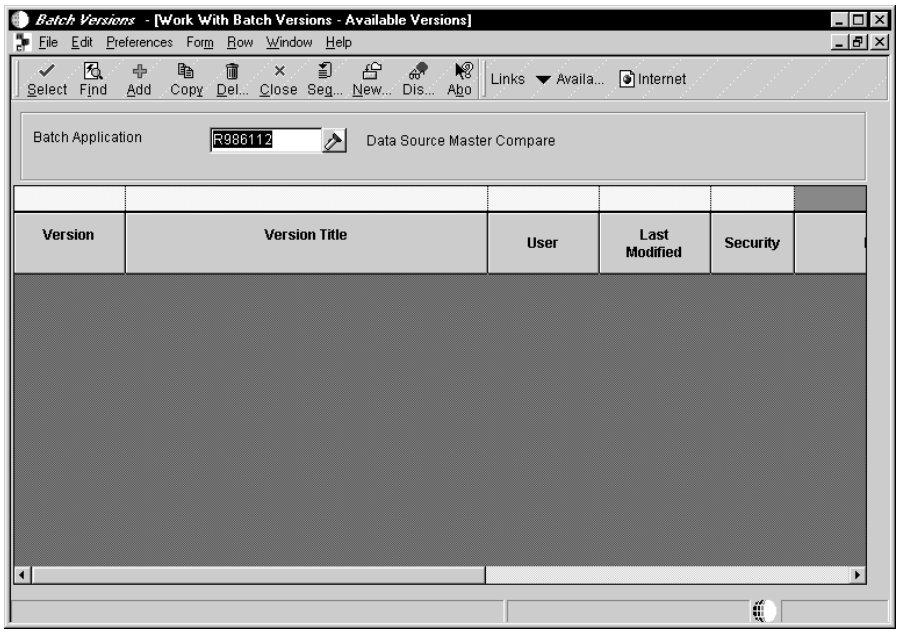
Before You Begin

- ❑ Use the report driver to enter processing options for the report. For information about entering processing options, see *Setting Processing Options for Data Source Reports* and *Processing options: Data Source Master (R98611) and Data Source Master Comparison (R986112)*.

The following tasks describe how to set up data selection for the report and how to run the report.

▶ To set up data selection for Data Source Master Comparison

From the System Administration Tools menu (GH9011), choose Batch Versions (P98305).



Enter R986112 at the Batch Application field, and then click Find.

Double-click version XJDE0001. The Version Prompting form appears.

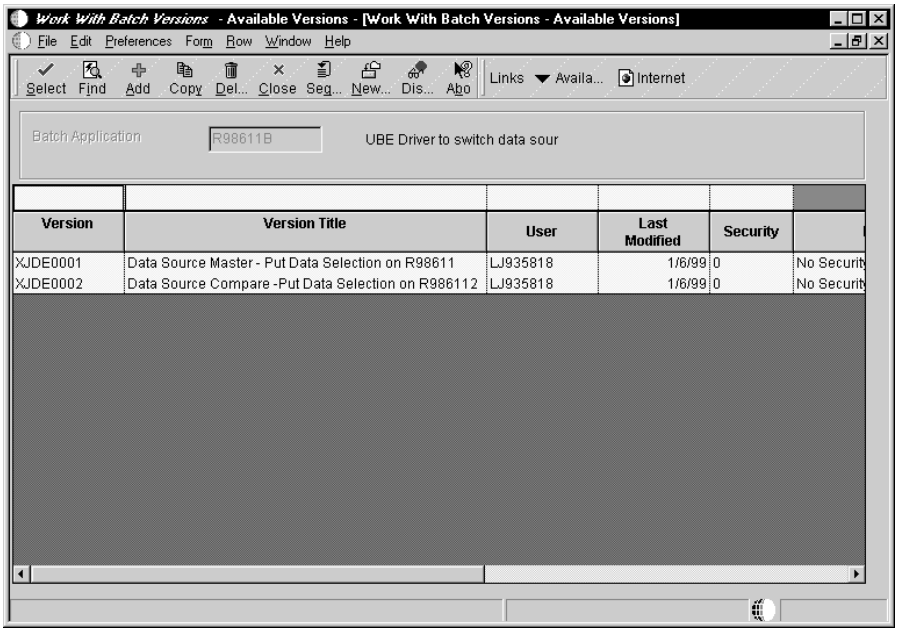
On Version Prompting, check the Data Selection option and then click Submit.

On the Data Selection form, choose from the appropriate columns to specify the exact records you want to compare.

Click OK.

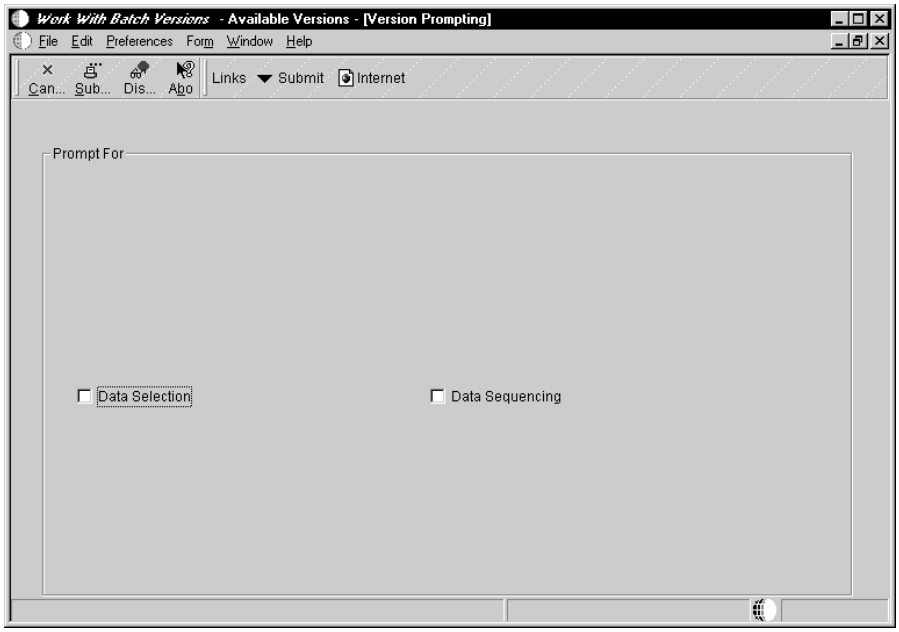
To run Data Source Master Comparison

From the Advanced Operations menu (GH9012), choose Data Source Master Comparison (R98611B). The Available Versions form appears.



On the Available Versions form, choose the version corresponding to the report you want to run.

Click Select. The Version Prompting form appears.



Click Submit to run the report.

Running the Data Source Master Report

Run the Data Source Master Report (R98611) to print a report of the Data Source Master table (F98611) for a specific data source.

The following tasks describe how to set up data selection for the report and how to run the report.

▶ To set up data selection for the Data Source Master Report

From the System Administration Tools menu (GH9011), choose Batch Versions (P98305).

Enter R98611 at the Batch Application field, and then click Find.

Select version XJDE0001, and then choose Data Selection from the Row menu.

The Processing Options form appears.

Enter the Processing Options and click OK.

The Report Output Destination form appears.

On Report Output Destination, choose On Screen or To Printer and click OK.

On the Data Selection form, choose from the appropriate columns to specify exact records.

Click OK.

To run the Data Source Master Report

From the Advanced Operations menu (GH9012), choose Data Source Master Report (R98611B). The Available Versions form appears.

On the Available Versions form, select the version corresponding to the report you want to run.

Click Select. The Version Prompting form appears.

Click Submit to run the report.

Middleware

In a client/server environment, applications must communicate across different platforms. These platforms can have different communications protocols, database management systems, and hardware operating systems. In order for clients to communicate with servers and servers to communicate with other servers, there must be a mechanism that can bridge multiprotocol and multivendor issues. This mechanism is a layer of software called middleware, which resides between the operating system and the business applications. It is important to have an application architecture that is based on a single, consistent middleware strategy.

J.D. Edwards provides the following types of middleware:

JDENet Communication Middleware	Performs the connections from client to server and server to server, and sends messages for distributed requests. It is a peer-to-peer, message-based, socket-based, multiprocess communication middleware solution.
JDEBase Database Middleware	Provides platform-independent application program interfaces (APIs) for multidatabase access. These APIs are used in two ways: • By OneWorld applications that dynamically generate platform-specific Structured Query Language (SQL), depending on the data source request. • As open APIs for writing advanced business functions in the C programming language. OneWorld uses these APIs to dynamically generate platform-specific SQL statements. JDEBase also provides client-to-server and server-to-server database access. To accomplish this, OneWorld is integrated with a variety of third-party database drivers, such as the IBM Client Access/400 database software, and the Microsoft Open Database Connectivity (ODBC) programming interface.

This section describes the following:

- ❑ Understanding JDENet communication middleware
- ❑ Understanding JDEBase database middleware

Understanding JDENet Communication Middleware

To communicate with each other across a network, the two computers must share a communications protocol (or set of protocols). A communications protocol is a formal set of rules that specifies the format and relationship for exchanging data among different devices. The following are the protocol layers with which the communication middleware is concerned:

Network layer	The network layer handles addressing and routing information, error checking, and retransmission requests.
Transport layer	The transport layer provides connection-oriented data-delivery services across networks. This layer provides end-to-end data exchanges in which systems maintain a session or connection with each other for the reliable, sequenced exchange of data. J.D. Edwards supports the TCP/IP protocol suite.
Application Layer	The application layer provides application-to-application interaction and data exchange. JDENet is the application layer communication middleware.

JDENet Communication Middleware

JDENet is the J.D. Edwards proprietary communication middleware package that provides for server-to-server and client-to-server communication. Features include:

- ❑ Socket-based communication
- ❑ Message-based communication
- ❑ OneWorld process-based design
- ❑ OneWorld dedicated processes

JDENet is a peer-to-peer middleware solution. Think of a client as a network conversation initiator and a server as a network conversation responder. A client always initiates the conversation by asking for something from another machine. A machine acts as a server when it responds to a network request, such as when

it gives something asked for by the client. In this peer-to-peer middleware solution, the distinction between client and server is determined by which machine starts the conversation. Any machine, running on any platform, can act as a client or a server at a given time.

With JDENet, communication between client and server occurs through messages. JDENet messages contain requests for processing such as requests for business functions, batch jobs, data replication, or OneWorld signon security. JDENet messages can originate from the client or the server. JDENet handles database requests only if multiple servers are in use and if they are different server types.

Application requests (messages) can be synchronous or asynchronous. A synchronous message, such as calling a business function, requires the client to wait for the server to complete the request. An asynchronous request, such as a batch process, enables the client to continue with another task while OneWorld processes the request. In some circumstances, business functions can also be called asynchronously.

Socket-Based Communication

A socket is a communications end point through which an application sends or receives packets of data across a network. Sockets provide a duplex communication channel between one or more systems. JDENet uses stream sockets to provide end-to-end communications. Sockets guarantee that the data arrives intact.

Message-Based Communication

Message-based communication means that applications send service requests for logic or data in the form of messages that are received and stored in a queue for processing. The middleware handles message transmission, which enables the client application to process other tasks. Without messaging services to handle these jobs, the application must wait until the request is handled and the results returned.

Messaging is most appropriate for applications that are event driven. It is the opposite of remote procedure calls (RPC), which are synchronous. The message packaging and handshaking of JDENet ensures that the message transmission is complete.

OneWorld Process-Based Design

Although OneWorld client workstations can have more than one copy of OneWorld loaded, only one OneWorld Windows-executable application can be running at any one time. OneWorld uses an internal network process (also referred to as a net process) called JDENet to communicate a request to the OneWorld server.

Servers also have a net process called JDENet. This process communicates with the client workstations and routes request messages to appropriate dedicated OneWorld processes. In turn, the dedicated processes route work to the appropriate platform-specific logic processes, such as DLLs, shared libraries, and job queues. A server can have multiple OneWorld main processes, multiple OneWorld dedicated processes, multiple DLLs, shared libraries, and job queues.

The advantage of this architecture is that multiple workstations can make requests to the same server at one time. You can control the number of workstations that can make and maintain a session connection to a main server process. You can also define the total number of dedicated processes (and the number of each type) that OneWorld uses to process specific types of workstation requests.

- Network processes
- Kernel processes

Network Processes

There is a relationship between the network processes, the dedicated processes, and the logic processes. This relationship is specifically defined by the jde.ini file on the enterprise server. Every enterprise server must have at least one OneWorld network process, referred to as a JDENET_n job. This job handles network connections and traffic for OneWorld.

As defined in the jde.ini file for each server, there can be multiple JDENET_n processes. Regardless of the number of JDENET_n processes, the initial JDENET_n process serves as the “master listener.”

If multiple JDENET_n jobs are specified, OneWorld starts the jobs as required, allocating a job to each request. When the maximum number of JDENET_n processes is started, OneWorld automatically alternates between the currently running JDENET_n jobs until the maximum number of connections is reached. This provides a degree of load balancing between OneWorld network processes. If, on a given server, the maximum number of connections for the JDENET_n job is met, a client or server cannot initiate an additional OneWorld session on that server until an existing session connection is ended. This is because, by design, all connections to JDENET_n persist for the duration of a OneWorld session.

For example, suppose that the jde.ini file on the server specifies that four JDENET_n processes are allowed. The first JDENET_n request is routed to the “master listener,” which is the initial JDENET_n process that is run at server start-up. When a second request to JDENET_n is received, the “master listener” receives the request and assigns it to a second JDENET_n process, which it then starts. This assignment persists for the duration of the session between the requesting device and this server. The same process occurs for the third and fourth JDENET_n requests. When the fifth request is received, that request is assigned to the first JDENET_n process, and the cycle continues.

Kernel Processes

The responsibility of the JDENET_n process is to handle the network layer of communication. If the JDENET_n job determines that the incoming message is a request for logic processing, it routes the request to an appropriate JDENET_k job. OneWorld determines an appropriate JDENET_k job based on message identifiers. The JDENET_k job is the OneWorld process that provides the link between the JDENET_n job and the appropriate platform-specific processing job. The JDENET_k process is applicable only to servers.

The JDENET_k job handles the two-way routing to and from the various logic processes, and the JDENET_n job handles the return delivery to the appropriate machine. There are 11 dedicated kernel types, each responsible for a specific type of OneWorld process.

Examples of logic processes include dynamic link libraries (.dll) for Windows NT platforms, shared libraries (.sl) for UNIX platforms, and JDENet processes for AS/400 platforms.

For more information about JDENet middleware, see *Middleware* in the *System Administration Guide*.

Understanding JDEBase Database Middleware

Different database management systems (DBMS) have their own version of Structured Query Language (SQL). For example, the following illustrates how Microsoft SQL Server, Oracle, and DB2/400 handle the same SQL statement:

Microsoft SQL Server	SELECT * FROM PRODDTA.F0101
Oracle	SELECT * FROM PRODDTA.F0101;
DB2/400	SELECT * FROM PRODDTA/F0101

The purpose of a database middleware layer is to provide a common interface to interpret the various versions of SQL. J.D. Edwards has a database middleware product called JDEBase, which is a common set of application programming interfaces (APIs) that programmers can call to request data and perform data manipulation logic. JDEBase interprets the generic APIs and converts the SQL into the appropriate statements for OneWorld to access the database.

Multiple databases in a distributed environment require a monitoring program to ensure database integrity. This monitoring program is referred to as a transaction monitor. The JDEBase database middleware has an embedded transaction monitor.

JDEBase provides:

- The ability to insulate developers from platform-specific SQL coding
- Rapid development of native drivers
- Server-to-server communication
- Transaction processing

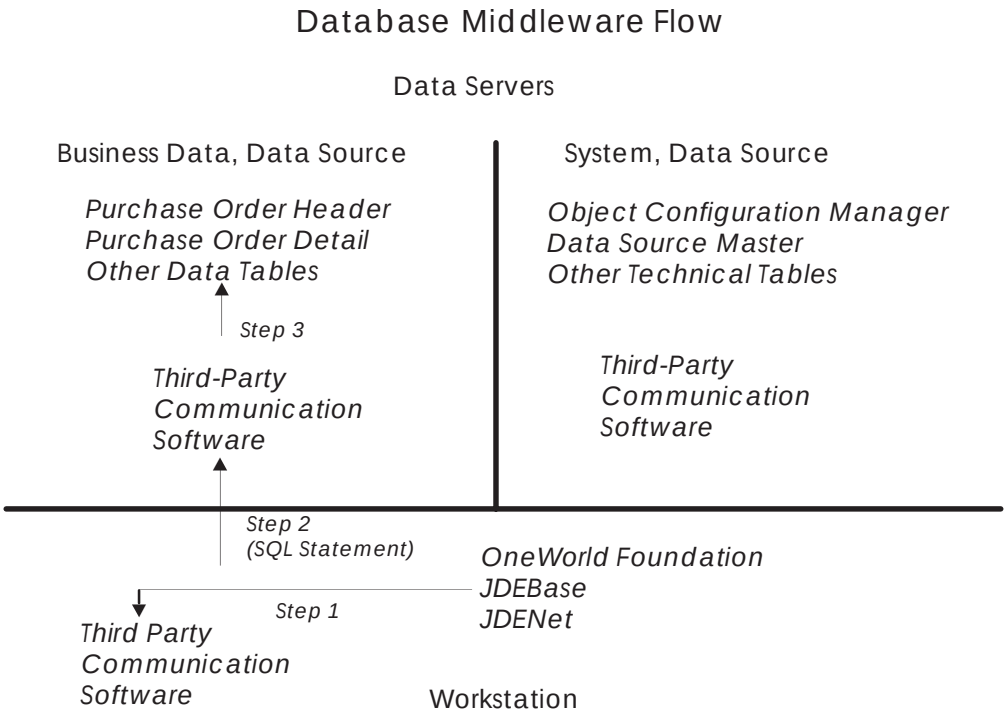
JDEBase provides a set of APIs to the developer and a set of translation programs to OneWorld. The translation programs are embedded in the data source definitions.

For example, suppose a data request for Address Master is made. The Object Configuration Manager (OCM) determines which data source contains the requested table. The Data Source Master table (F98611) provides the database

information. This includes the required driver, which is loaded to translate the request into the appropriate SQL statements.

Database Middleware Flow

The following example illustrates the data-request process. For this example, the user enters search criteria for a Purchase Order and clicks Find.



OneWorld accommodates any number of data servers. This illustration shows only two.

1. OneWorld sends a data request to JDEBase on the workstation. The request includes the form's data structure and any values needed to locate the record.
2. JDEBase does the following:
   - Builds data structures from the application structures, creates the actual SQL statement or equivalent commands, and passes it to the third-party communication software, which resides on the workstation. Examples of third-party communication software are IBM Client Access/400, Microsoft Open Database Connectivity (ODBC), or Oracle SQLNET.
   - Manages the physical connections to the database.
   - Manages optimal fetch algorithms.
   - Performs all binding.

- Passes result codes back to the application. Logs errors in the log files (jde.log).

3. The third-party communication software on the workstation passes data to the third-party communication software on the server. The server then accesses the table and returns the data back to the third-party communication software.
4. Finally, the third-party communication software passes data back to the JDEBase and the OneWorld Foundation processes.

Modes of Processing

OneWorld provides many ways for you to run interactive applications. The method you select depends on your needs and your system configuration. All methods allow you to distribute data and logic in a manner that optimizes both the power of the workstation and the data integrity of the server.

This section contains the following topics:

- ❑ Understanding the modes of processing
- ❑ Working with direct-connect processing
- ❑ Working with store-and-forward processing
- ❑ Working with detached mode processing

Understanding the Modes of Processing

OneWorld has four operation modes for interactive applications:

- ❑ Direct-connect processing
- ❑ Store-and-forward processing
- ❑ Batch-of-one processing
- ❑ Zero-client processing

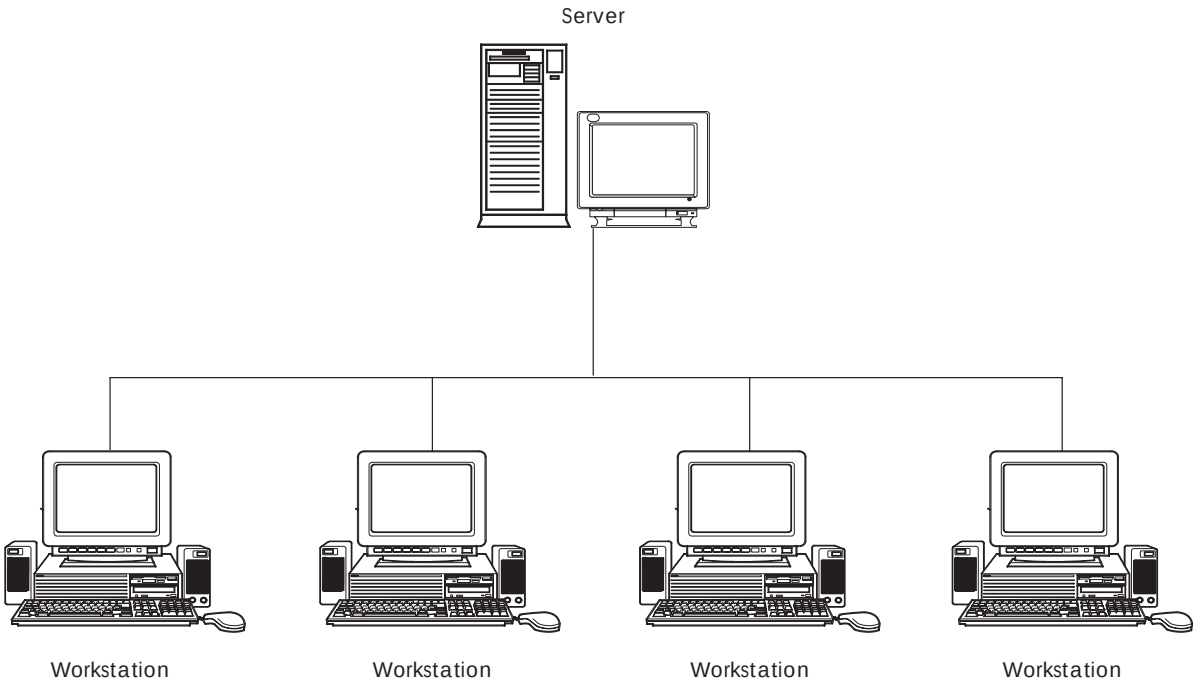
Understanding Direct-Connect Processing

With direct-connect processing, when you run an application you are connected to a server that contains your major business data. The direct-connect model allows you to distribute data and logic. For performance reasons, consider distributing your data and logic in a manner that reduces network traffic and unnecessary input/output on the server.

While J.D. Edwards allows batch applications to run locally, there is rarely a business reason to do this. Therefore, all batch applications should be set up for server processing (distributed).

The following illustration shows a simple direct-connect configuration.

Direct–Connect Processing



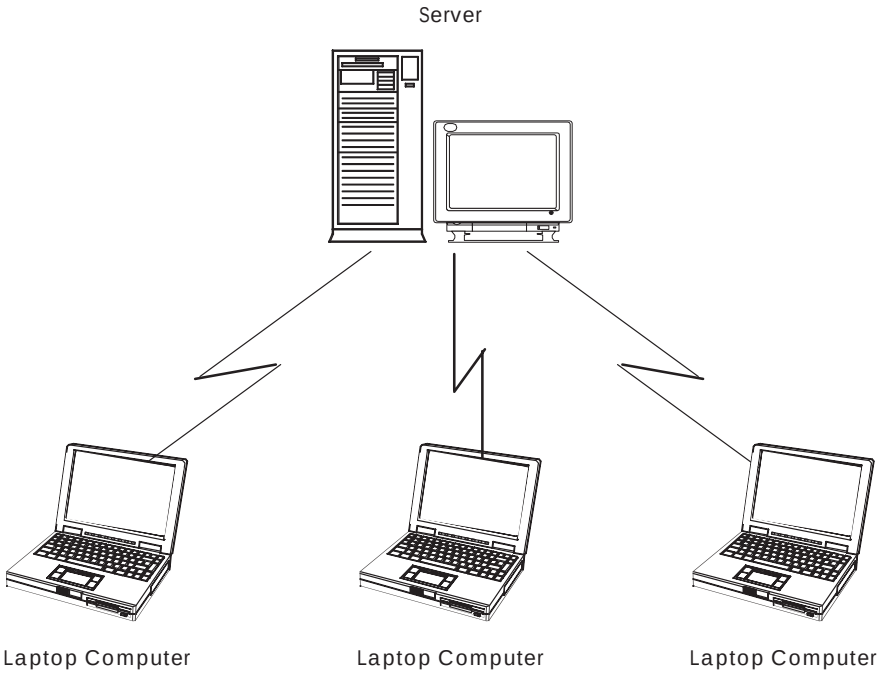
See *Working with Direct-Connect Processing* for more information.

Understanding Store-and-Forward Processing

J.D. Edwards designed the store-and-forward model for users who need to enter transactions on a workstation that is disconnected from a server. For example, a sales representative working away from the office can use a laptop to enter an order. With store-and-forward processing, that user can hook up to the network at a later time and transfer the data.

The following illustration shows a simple store-and-forward configuration.

Store-and-Forward Processing



OneWorld applications designed for store and forward are separate applications. These applications perform edits on static data and other critical information that must be valid to process an order. After the initial edits are complete, OneWorld stores the transaction in work tables, called Z files, on the workstation. When a network connection is established, Z files are uploaded to the enterprise server. The transactions are edited again by a master business function that then updates the records into transaction files.

The upload process, which is the batch process the user initiates, transfers data from your workstation and performs the following:

- Copies order information from the work tables on the workstation to work tables on the server
- Updates the Transaction Control (F0041Z1) table on the server with one transaction per record and provides one of the following statuses: uploaded, processing, complete with errors, or complete without errors.
- Updates the Transaction Control table on the workstation to indicate that OneWorld uploaded the transactions. This prevents OneWorld from uploading the same transactions more than once.
- Invokes a master business function on the server to repeat all local editing and additional editing that was not deemed critical during order entry, and writes records to the permanent transaction tables.

- Sends success or error messages to the Work Center, where users can review and clear them.
- Updates the Transaction Control table on the server and workstation with the status of each transaction processed in the upload.

The Work Center sends to the user who executed the process a message for each completed job (with or without errors). To clear an error, drill down to see the details. If the error requires a user action, drill down again to bring up the exact transaction where the error occurred.

See *Working with Store-and-Forward Processing* for more information.

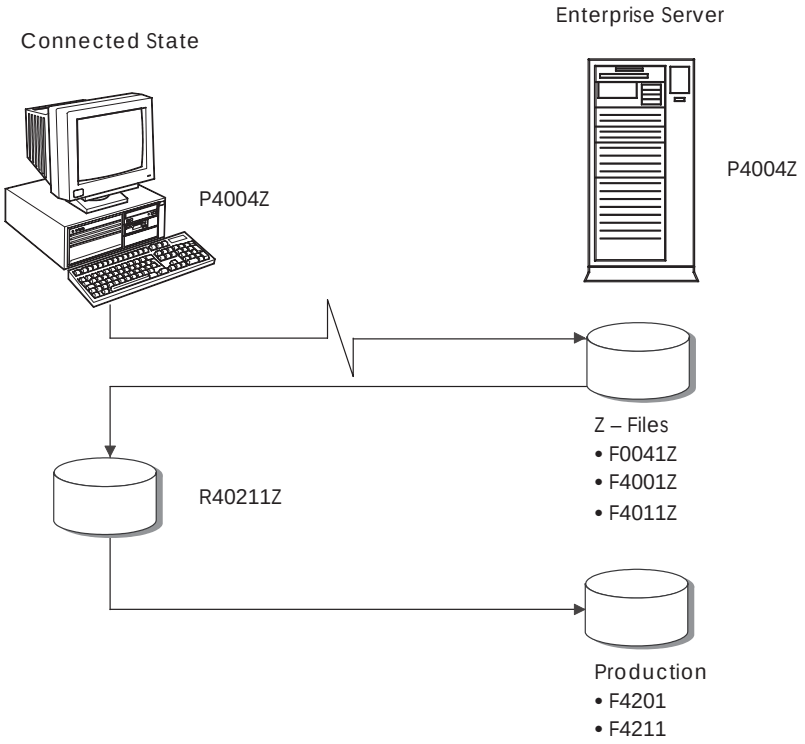
Understanding Batch-of-One Processing

Batch-of-one processing combines the performance benefits of the store-and-forward model with the real-time updates of transaction data found in direct-connect processing. Batch-of-one processing uses the store and forward application (P4004Z) with the standard production environment, for example PD7333, which means the local workstation has a physical connection to the network. The P4004Z application still uses the Z-files but these tables are located on the enterprise server. Each sales order is written directly to the enterprise server.

Batch-of-one processing greatly reduces network traffic because the P4004Z application performs fewer field-by-field edits. For this reason is ideal for sales order entry over wide area networks (WANs).

One of the trade-offs to batch-of-one processing is that the user gives up direct access to the F4201 and F4211 files, which contain the most current and complete data on all sales orders. Since the P4004Z application does not directly access those files, it can only review, update, and delete current orders in the Z files. Once those orders are moved to the production files (F4201 and F4211), the user would have to use the P4210 application to review, update, or delete a sales order.

The following graphic explains how batch-of-one processing works for sales order entry processing:



Batch-of-one processing has two modes of processing available:

- Subsystem
- Online mode

To set up either type of processing, change the processing option in the Recurring Orders application (P4004Z).

Subsystem

The subsystem type of batch-of-one processing has a version of the R40211Z batch application running in the background. The subsystem batch application periodically checks the Z-files on the enterprise server, performs the batch edit function when data is present and ready to be processed, and then transfers the data from the Z-files on the enterprise server into production tables F4201 and F4211.

Note: Version ZJDE0002 of the R40211Z batch application is shipped with a period of 30 milliseconds. This means that this batch application is running at a frequency of 33.3 times per second, which is not the best use of CPU resources on the enterprise server. Should batch-of-one subsystem mode be utilized in a production environment, a more reasonable figure for this value should be 15,000 milliseconds or once every 15 seconds.

Online Mode

The online option presents the batch application printer selection screen following completion of the sales order. After selecting the printer, a version of the R40211Z batch application performs the batch edit function, transferring the data from the Z-files on the enterprise server into production tables F4201 and F4211.

OneWorld sends all errors to the Work Center. Users can either set an option to have OneWorld prompt them when a new message is sent, or they can wait and view all messages.

Understanding Zero-Client Processing

Zero-client processing enables you to connect to OneWorld applications and data from your workstation using only an Internet browser. This mode of processing is similar to direct-connect processing because the data and the logic are stored in a central location. This configuration allows you to easily maintain storage requirements on your workstation. For more information about zero-client processing, see the *Web-Based Solutions Guide*.

Working with Direct-Connect Processing

In direct-connect processing, workstations are connected directly to servers that can store data and process logic. For performance reasons, you will want to distribute your data and logic in a manner that reduces network traffic and unnecessary input and output on the server.

The following topics provide suggestions for distributing objects to support various direct-connect processing configurations:

- ❑ Recommendations for data and logic distribution
- ❑ Setting up direct-connect processing
- ❑ Understanding object mappings for direct-connect environments

Recommendations for Data and Logic Distribution

To achieve the best performance with direct-connect processing, J.D. Edwards recommends the following location for data and logic processing:

- Map your transaction data to a data server.
- Map user defined codes and menus to the workstation. You can also map other static files locally. If the maintenance costs are more than your performance returns in mapping these tables locally, you can map them to a data server.
- Map all batch applications to the enterprise logic server.

Setting Up Direct-Connect Processing

The following is an overview of the process involved in setting up direct-connect processing.

1. Create a production environment and verify that the new environment uses a production path code.

   See *Adding an Environment* for information.

2. If applicable, determine the name of the master business functions you should map to the server.

See *Locating Master Business Functions* for information.

3. Modify Object Configuration Manager mappings for the new environment.

   See *Working with the Object Configuration Manager* for information.

Understanding Object Mappings for Direct-Connect Environments

Depending upon your configuration, you might require multiple direct-connect environments. For example, to support multitiered configurations, you might need an environment that maps all application processing to one server and data to a corporate server. Examples of other direct-connect environments include:

- Master data administration environment
- Strategic reporting environment

Master Data Administration Environment

The data administrator uses the Master Data Administration environment to maintain the published tables in the central location. To set up this environment, do the following:

- Map all table objects to a business data data source on your server
- Map user defined codes and menus to a control table data source on your server. The control data contains the published set of user defined codes
- Map system tables to the system data source
- Map Object Librarian tables to the Object Librarian data source
- Map data dictionary tables to the data dictionary data source
- Map batch applications to the server

Strategic Reporting Environment

The Strategic Reporting environment supports configurations that have operational data replicated to another data source for strategic reporting purposes. To set up this environment, do the following:

- Map all table objects to a business data data source on your server (this data source should contain the replicated data for reporting)
- Map user defined codes and menus to the workstation
- Map system tables to the system data source
- Map Object Librarian to the Object Librarian data source
- Map data dictionary to the data-dictionary data source
- Map batch applications to the server

Locating Master Business Functions

In a direct-connect environment, you need to identify the master business functions so that you can map them to a server.

▶ To locate master business functions

From Object Librarian, type 1 in the Query By Example column for Function Type.

Click Find. A list of Master Business Functions appears.

Working with Store-and-Forward Processing

Store-and-forward is a mode of processing that enables users who are disconnected from a server to enter transactions, and then later connect to the server to upload those transactions. For example, sales people on the road who use laptops can process transactions and then upload those transactions to the server at a later date. Store-and-forward can also help departments that have store-and-forward applications by allowing them to enter transactions in a disconnected state (which improves performance because the applications perform all edits locally), and then upload those transactions to the server during off-peak hours.

OneWorld performs minimal editing of the transactions on the store-and-forward machine, which allows for fast entry. A master business function performs the full editing when the user uploads the store-and-forward transactions to the enterprise server.

Consider the following factors that must be in place for store-and-forward processing:

- A process for replicating business data to the local workstation
- A process for replicating the system data source (technical data tables) to the local workstation
- A store-and-forward environment for disconnected transaction entry
- A direct-connect environment for uploading transactions to the enterprise server

Important: To maintain data integrity, you need to ensure that unique next numbers or document types exist for each store and forward user. Depending on your business requirements, J.D. Edwards recommends the following solutions:

- If your enterprise has a small number of store-and-forward users, you can assign a separate document type for each user. For example, user 1 might have the document type S1, user 2 might have the document type S2, and so on. When user 1 uploads an order number 1009, the order would be 1009S1. User 2 would forward record 1009S2. This method avoids data corruption by creating data unique to a specific sales person and maintaining separate business data during the upload process.

  Note: Do not use all your document types for store and forward users because you might need document types for other areas in your enterprise.

Document types range from A0-Z9. If you think you might use all your document types for store-and-forward users, you should consider the following solution to maintain data integrity in a store-and-forward environment.

- If your enterprise has a large number of store-and-forward users, assign each user a unique range of next numbers to assign to records. For example, user 1 assigns only numbers between 0001-9999, user 2 assigns only numbers between 10000-19999, and so on. Multiple records do not share numbers, so your enterprise maintains data integrity. This method can be difficult to manage, so you should use this method only if other methods will not work for your enterprise.

The following tasks explain how to set up store-and-forward processing:

- ❑ Setting up store-and-forward processing
- ❑ Administering store and forward processing
- ❑ Performing end-user tasks

This topic contains the following tables:

- Technical data tables needed for store-and-forward
- Business data tables needed for store-and-forward

Setting Up Store-and-Forward Processing

Following are the first steps you should perform when setting up store-and-forward processing. This is a one-time process you do after each workstation installation.

▶ To set up store-and-forward processing

Create a new environment for your store-and-forward processing. This environment should use your production path code. You can call this environment SFPD7333 or something similar.

Do not create the environment by copying an existing environment, because the copy will include that environment's Object Configuration Manager (OCM) mappings. Also, make sure that the Just-in-time Installation flag is set to N to disable just-in-time installation for the store-and-forward environment.

See *Adding an Environment* for more information.

Using the Object Configuration Manager, create a default map for table (TBLE) objects for the store-and-forward environment that points to the local Microsoft Access data source called OneWorld Local - PD7333.

See *Working with the Object Configuration Manager* for more information.

After you have created the store-and-forward environment, add it to your store-and-forward users' environment list.

See *Assigning Environments to User and Group Profiles* in the *System Administration Guide* for information.

Copy the store-and-forward Microsoft Access database (StoreFwd.mdb) that J.D. Edwards provides. The database is located in the *deployment server machine name*\b733\planner\data directory on the deployment server. Copy the database to the b7\pd7333\data directory on the administrator's local workstation. You now have two Microsoft Access databases under the b7\pd7333\data directory on the administrator's workstation.

From the b7\pd7333\data directory on the administrator's workstation, export to the StoreFwd.mdb database the following tables from any JDEb7.mdb database that has your most current data:

F0004	User Defined Code Types
F0005	User Defined Codes
F0082	Menu Master File
F00821	Menu Selections File
F0083	Menu Text Override File
F0084	Menu Path File

Ensure that the administrator's workstation is set up as a subscriber to the menu and user defined code tables so that future releases have the latest tables. See *Working with the Data Replication Application* in the *System Administration Guide* for information about subscribers.

From the b7\pd7333\data directory on the administrator's workstation, delete the JDEb7.mdb database and rename StoreFwd.mdb to JDEb7.mdb.

Using Microsoft Access, ensure that all of the technical and business data tables are in the JDEb7.mdb on the administrator's workstation and that all of the tables (except user defined codes and menus) are empty.

If any tables are missing, import them from the JDEb7.mdb database located on the deployment server in the following path: *deployment server machine name*\b733\planner\data. Make sure you import only the structure and not the data. That is, import only empty tables (except for user defined codes and menus).

Copy all data dictionary tables (F92*) or a full data dictionary TAM file to the store-and-forward client machine.

See *Technical Tables Needed for Store-and-Forward* and *Business Data Tables Needed for Store-and-Forward* at the end of this chapter for a list of all of the tables needed for store-and-forward processing.

This completes your initial setup of store-and-forward processing. You must perform the administration and end-user tasks explained in this chapter at least once to enable store-and-forward processing.

Administering Store-and-Forward Processing

The following tasks create the JDEb7.mdb Microsoft Access database on the administrator's workstation. Your end-users need the JDEb7.mdb database to perform store-and-forward processing.

Important: You must perform these tasks at least once. The administrator should perform these tasks on a regular schedule to ensure accurate and current data for your store and forward users. These tasks refresh your server store and forward Microsoft Access database.

Complete the following tasks, in order, to perform routine administration:

- ❑ Downloading technical data
- ❑ Downloading business data
- ❑ Making the database accessible to end-users
- ❑ Verifying the correct store-and-forward settings in the jde.ini file

Downloading Technical Data

For store-and-forward processing, you must copy the technical data tables from the production environment to the administrator's workstation. Use the Populate System Tables for Store & Forward batch application (R98403) to perform this process, which uses a Copy Table script. This enables your users to run the technical data locally after they install the store-and-forward package. (Users do not have access to the server during store-and-forward transaction processing.)

The administrator's workstation must have the latest production package installed on it. You need to perform this task only if your technical data has changed.

▶ **To download technical data**

From the System Administration Tools menu (GH9011), choose Batch Versions (P98305).

On the Work With Batch Versions form, type R98403 in the Batch Application field and then click Find.

Choose the version titled Populate System Tables for Store-and-Forward, and then click Select.

On the Version Prompting form, choose the Data Selection option and then click Submit.

On the Data Selection form, ensure that all of the technical data tables are listed in the data selection area of this form.

See *Technical Data Tables Needed for Store-and-Forward* for a list of these tables.

Click OK.

The Processing Options form appears.

Enter the following values for the processing options, and click OK:

Enter the Environment for the database to be created for. (If this report is called from another process, the Environment will be passed in.)	Enter the name of the store-and-forward environment that you created, such as SFPD7333.
or Enter the Data Source for the database to be created for.	If you do not enter the previous option, enter the name of the store-and-forward data source.
Enter a '1' to load Production Data or a '2' to load Demonstration Data. The default is to load production data. (If this report is called from another process, this flag will be passed in.)	You should always enter 2 for demonstration data because this batch process looks at the Copy Data Y/N flag in the Object Librarian record to determine whether data should be copied for this table. Because some of these technical tables will be specified with a NO for the Copy Data Y/N flag, a value of 2 for this processing option means OneWorld will ignore that flag and copy all of your normal production data for those tables.
Enter the source Data Source for Loading of Data. (If this report is called from another process, the source Data Source will be passed in.)	Leave blank if you are copying from many data sources, and use the next processing option instead. If you use this option, enter a data source name such as OneWorld Local.

or Enter the Source Environment for the database to be copied from. (If this report is called from another process, the Source Environment will be passed in.)	Instead of entering a data source name, enter the name of the source environment. Use this option instead of the previous option if you have multiple data sources from which to copy that all reside within one environment.
Enter a '1' for Proof Mode or a '2' to create the Environment Database.	Enter 2 to generate a report and copy the technical tables to the local machine.
Enter a 'A' to Re-create existing tables in data sources that allow automatic table creation. The default is not to re-create tables.	Enter A to clear files and re-create the data.
Enter a '1' to only copy tables that exist in the target data source.	Leave blank.
Enter a 'Y' to add records without cleaning the target table. This is used for language tables.	Leave blank.
Enter a '1' to only print exceptions. The default is to print all lines.	Enter 1 to print only the exception lines of the report. Leave this option blank to print the report. This report describes any errors encountered during the download, including a list of any technical tables missing from the JDEb7.mdb Microsoft Access database on the administrator's workstation.
Enter a 'Y' to create all tables or a 'N' to create licensed tables only.	Enter Y to create all tables.
Enter a 'Y' to print all tables in the report or a 'N' to print licensed tables only.	Enter Y to print all tables.

After you enter processing options, Copy Table runs and a Copy Table message box appears providing a status of the download.

Downloading Business Data

Because the business data is shared by multiple store-and-forward users, run this batch process to download the records for the master tables that pertain to all of the store-and-forward applications from the server to the JDEb7.mdb Microsoft Access database.

Before You Begin

- ❑ Use Microsoft Access to ensure that all business data table definitions exist within the JDEb7.mdb Microsoft Access database on the administrators workstation.

▶ To download business data

From the System Administration Tools menu (GH9011), choose Batch Versions (P98305).

On the Work With Batch Versions form, type P00140 in the Batch Application field and then click Find.

Note: For all non-populated business data tables, you must run the Environment Database Creation batch application R98403.

All of the versions that J.D. Edwards ships appear. These versions are the batch processes that download the business data table.

You must run each version of the business data tables that you need copied to the administrator's local machine.

See *Business Data Tables Needed for Store-and-Forward* for a list of the business data tables you need to download.

Choose the business data table version, and then choose Run Version from the Row menu.

On the Version Prompting form, click Submit.

On the Report Output Destination form, choose a destination and then click OK.

On the Environment Overrides form, complete the following fields with the name of the source environment and then click OK:

- Source Environment Override (ensure that this is your normal production environment, which points to all of your publisher tables that you have set up)
- Target Environment Override (this should be your store-and-forward environment, which should have all of its OCM mappings pointing to OneWorld Local)

If you click a Browse button, the Environment Search form appears. Choose the appropriate environment, and click OK.

Based on your store-and-forward application-specific needs, run the appropriate version for the Store-and-Forward Download (P00140) batch application. Each version populates one master table on the JDEb7.mdb Microsoft Access database. Ensure that the batch process runs locally (rather than on a server).

Users can run this batch process from their workstation after they have copied the Microsoft Access database from the server to their workstation. This would mean the administrator could skip this step, leaving the business data tables empty.

Making the Database Accessible to End-Users

This task explains how to deploy the JDEb7.mdb Microsoft Access database created with the technical and business data downloading processes. You must make this database accessible to your store-and-forward users because this is the database that they copy to their workstations.

▶ To make the database accessible to end-users

Copy the JDEb7.mdb Microsoft Access database from the administrator's workstation to a shared folder on the deployment server.

Refresh this database as needed to keep your business data synchronized.

Verifying the Correct Store-and-Forward Settings in the jde.ini File

This task explains how a store-and-forward user's jde.ini [DB SYSTEM SETTINGS] should be set. The jde.ini file must be set this way for store-and-forward processing to work on an end-user's workstation.

▶ **To verify the correct store-and-forward settings in the jde.ini file**

In your store-and-forward end-user's jde.ini file, ensure that under the [DB SYSTEM SETTINGS] section, the Default Env is set to SFPD7333. Also, verify that under the [DB SYSTEM SETTINGS - SECONDARY] section the default is set to OneWorld Local - PD7333. This default environment accesses the local Access database after the primary network database setting times out.

You can also change your setup.inf file to have the default environment set to SFPD7333 for your production packages so that your store-and-forward users do not need to update their jde.ini file after each workstation installation.

The settings should look like the following:

```
[DB SYSTEM SETTINGS]
Default Env=SFPD7333
Default Pathcode=PD7333
```

```
[DB SYSTEM SETTINGS - SECONDARY]
Default Env=OneWorld Local - PD7333
Default Pathcode=PD7333
```

Performing End-User Tasks

The following tasks explain what your store-and-forward end-users must perform to run store-and-forward processing.

Complete the following tasks, in order, to perform end-user tasks:

- ❑ Updating the end-user database for store-and-forward
- ❑ Entering store-and-forward transactions
- ❑ Uploading store-and-forward transactions

Updating the End-user Database for Store-and-Forward

Your store-and-forward users must perform this task at least once to enable store-and-forward processing on their workstations.

Caution: Your users should perform this task on a regular schedule to ensure accurate and current data. You should establish procedures for how often store and forward users need to refresh their Microsoft Access database from the Microsoft Access database on the server.

▶ To update the end-user database for store-and-forward

Ensure that your store-and-forward users have the current production package installed on their workstation.

With their workstation connected to the enterprise, have your store-and-forward users copy the JDEb7.mdb Microsoft Access database from the shared folder on the server to the b7/data directory on their workstations, which will replace the JDEb7.mdb database already on their workstations.

Entering Store-and-Forward Transactions

This task explains how to prepare your users' workstations for store-and-forward processing, and then how to log onto OneWorld and enter store-and-forward transactions.

▶ To enter store-and-forward transactions

After the latest production package and the latest JDEb7.mdb Microsoft Access database has been installed on their workstation, have your store-and-forward users disconnect their workstations from the enterprise network and reboot their workstation.

With their workstation disconnected from the enterprise, have your users log onto OneWorld.

OneWorld first attempts to connect to the System data source.

The System data source password screen appears.

Click Cancel.

OneWorld switches to the secondary base data source, which is pointing locally (OneWorld Local - PD7333).

Your store-and-forward users can now enter their store-and-forward transactions. They should consult the application-specific documentation for store-and-forward processing of journal entry, voucher entry, and sales order entry transactions.

Uploading Store-and-Forward Transactions

This process describes how to upload to the enterprise server the transactions entered by store-and-forward users. A batch application uploads transactions from the work files on the local machine to the work files on the server.

▶ To upload store-and-forward transactions

After your users have completed their transactions and are ready to upload their store-and-forward transactions:

Have them sign off OneWorld.

Have them reconnect their workstation to the enterprise network and reboot their workstation.

Have them sign onto their normal production environment.

On the System Administration Tools menu (GH9011), choose Batch Versions (P98305).

On the Work With Batch Versions form, type one of the following applications in the Batch Application field and then click Find:

- R0911Z1 for Journal Entry Upload
- R0411Z1 for Voucher Upload
- R42101Z for Sales Order Transaction Upload

You must run each of these versions separately to upload each type of transaction that you processed with store-and-forward.

Choose the upload version and then click Select.

Caution: To properly process the data, you must run this batch application locally on the workstation.

This process copies the following store-and-forward work files from your workstation to your server:

- Transaction Control File (F0041Z1)
- Journal Entry Transactions - Batch File (F0911Z1)
- Voucher Transactions - Batch File (F0411Z1)
- Batch Receiver File - Order Headings (F4001Z)
- Batch Receiver File - Order Details (F4011Z)

After you run each upload batch process, OneWorld creates and displays a transmission upload report for all of the transactions that you upload to the server.

Use this report to verify that OneWorld uploaded the transactions correctly.

From the server, run OneWorld. On the Work With Batch Versions form, type one of the following applications into the Batch Application field and click Find:

- R09110Z for Journal Entry Batch Processor
- R04110Z2 for Voucher Batch Processor
- R40211Z for Sales Order Batch Transaction Editor

Running the batch application creates records in the transaction files on the server using the data from the work files.

Store-and-forward processing is now complete.

Technical Data Tables Needed for Store-and-Forward

The following technical data tables must reside on the user's local machine that the tables use for store-and-forward processing.

See *Downloading Technical Data* in this chapter for information on how to copy these tables to a local machine.

Table	Description
F0002	Next Numbers - Automatic
F00021	Next Numbers by Company/Fiscal Year
F0092	Library List User
F00921	User Display Preferences
F00922	User Display Preferences Tag File
F00924	User Install Packages
F00925	User Access Definition
F0093	Library List Control

Table	Description
F0094	Library List Master
F00941	Environmental Detail - OneWorld
F00942	Object Path Master
F00945	Release Master
F00950	Security Table
F98101	Imaging Constants
F983051	Versions List
F98306	Processing Option Text
F9860	Object Librarian Master Table
F986101	Object Configuration Manager Mappings
F98611	Data Source Master
F98613	Business View Environmental Server
F9865	Form Information File
F98825	Package Deployment Scheduling
F98950	User Overrides Table
F98980	Font Override by Language

In addition to the technical data tables, make sure you have the following tables on the local machine. These tables should already be present because they are files that you should have replicated to the local machine as part of your data replication process:

Table	Description
F0004	User Defined Code Types
F0004D	User Defined Codes - Alternate Language Descriptions
F0005	User Defined Codes
F0005D	User Defined Codes - Alternate Language Descriptions
F0082	Menu Master File
F00821	Menu Selections File
F0083	Menu Text Override File
F0084	Menu Path File

The following tables are the data dictionary tables that must reside locally on the store-and-forward client:

Table	Description
F9200	Data Item Master
F9201	Data Field Specifications
F9202	Data Field Display Text
F9203	Data Item Alpha Description
F9204	Data Item Aliases
F9205	Data Dictionary - Error Message Program ID
F9207	Data Dictionary - Error Message Information
F9210	Data Field Specifications (OneWorld)
F9211	Data Dictionary - Smart Fields

The following table conversions are available if you want to refresh data for tables and do not need the absolute latest data. You can find these table conversions on the Periodic Replication of Technical Tables menu (GH911).

Replicated Table	Description
R910002	Replication of Next Numbers - Automatic
R9100021	Replication of Next Numbers by Company
R9100165	Replication of Media Objects
R9100825	Replication of Menu Word Search
R910092	Replication of Library List User
R9100921	Replication of User Display Preferences
R9100922	Replication of User Display Prefs Tag File
R9100924	Replication of User Install Packages
R9100925	Replication of User Access Definition
R910093	Replication of Library List Control
R910094	Replication of Library List Master
R9100941	Replication of Environmental Detail - OW
R9100942	Replication of Object Path Master
R9100945	Replication of Release Master
R9100950	Replication of Security Tables
R919200	Replication of Data Dictionary Tables
R9198101	Replication of Imaging Constants
R91983051	Replication of Versions List
R9198306	Replication of Processing Option Text
R91986101	Replication of OCM Mappings
R9198611	Replication of Data Source Master
R91986115	Replication of Table and Data Source Sizing
R9198613	Replication of BSVW Environmental Server
R91986150	Replication of Server Transfer Package Cont
R91986151	Replication of Server Package Transfer File
R91986152	Replication of Server Package Master
R9198616	Replication of Printer Definition
R91986161	Replication of Default Printer Table
R91986162	Replication of Paper Definition
R91986163	Replication of Printer Capability
R91986164	Replication of Output Conversions
R91986165	Replication of Printer Security
R919865	Replication of Form Information File
R9198701	Replication of Next ID Master
R9198825	Replication of Package Deployment Scheduling
R919885	Replication of Installation Package
R919886	Replication of Installation Package Detail
R919887	Replication of Installation Package Build
R9198950	Replication of User Overrides
R9198980	Replication of Font Override by Language
R9198CONST	Replication of Table of Constants
R9198DRENV	Replication of Data Replication Environment Map
R9198LOG	Replication of Data Replication Change Log
R9198DRPCN	Replication of Pending Change Notification
R9198DRPUB	Replication of Data Replication Publishers

Replicated Table	Description
R9198DRSUB	Replication of Data Replication Subscribers
R9198EVDTL	Replication of Event Detail File
R9198EVHDR	Replication of Event Header File
R9198MOQUE	Replication of Media Object Queues
R9198TMPL	Replication of Templates
R9198VAR	Replication of Table of Variables
R9198OWSEC	Replication of Security Table
R9198SYSDS	Replication of System Data Source
R9198403	Replication of Environment DB Creation

Business Data Tables Needed for Store-and-Forward

The following business data tables must reside on the local machine used for store-and-forward processing. To copy these tables to the local machine, you must copy each table separately.

See *Downloading Business Data* in this chapter for information about how to copy these tables to a local machine.

Before Downloading the User Defined Code Tables

Before end-users download the User Defined Code (F0004 and F0005) tables for any of the verticals (such as Journal Entry, Voucher Entry, or Sales Order Entry), they should use the following criteria to make sure they need these tables:

- The end-user needs to perform data selection on these tables
- The system administrator has not populated these tables as part of the store-and-forward JDEb7.mdb Microsoft Access database provided for end-users

Voucher Entry (P0411Z1)

Table	Description
F0004	User Defined Code Types
F0005	User Defined Codes
F0006	Cost Center Master
F0008	Date Fiscal Patterns
F0008B	Date Fiscal Patterns - 52 Period Accounting
F0009	General Constants
F0010	Company Constants
F0012	AAIs
F0013	Currency Codes
F0014	Payment Terms
F0022	Tax Rules
F0041Z1	Transaction Control File
F0101	Address Book Master
F0111	Address Book - Who's Who
F0150	Address Organization Structure Master File - OSTP, PA8, AN8
F0116	Address by Date
F0401	Supplier Master
F0411Z1	Voucher Transactions - Batch File
F0901	Account Master
F0907	Chart of Accounts Format
F0911Z1	Journal Entry Transactions - Batch File
F4008	Tax Areas
F4801	Work Order Master File

Note: Additional tables might be necessary.

Journal Entry (P0911Z1)

Table	Description
F0004	User Defined Code Types
F0005	User Defined Codes
F0006	Cost Center Master
F0008	Date Fiscal Patterns
F0008B	Date Fiscal Patterns - 52 Period Accounting
F0009	General Constants
F0010	Company Constants
F0012	AAIs
F0013	Currency Codes
F0014	Payment Terms
F0022	Tax Rules
F0041Z1	Transaction Control File
F0101	Address Book Master
F0150	Address Organization Structure Master File - OSTP, PA8, AN8
F0111	Address Book - Who's Who
F0116	Address by Date
F0301	Customer Master
F0901	Account Master
F0907	Chart of Accounts Format
F0911Z1	Journal Entry Transactions - Batch File
F4008	Tax Areas
F4801	Work Order Master File

Note: Additional tables might be necessary.

Sales Order Entry (P4004Z)

Table	Description
F0004	User Defined Code Types
F0005	User Defined Codes
F0006	Cost Center Master
F0006D	Business Unit Alternate Description Master
F0008	Date Fiscal Patterns
F0008B	Date Fiscal Patterns - 52 Period Accounting
F0009	General Constants
F0010	Company Constants
F0012	Automatic Accounting Instruction Master
F0013	Currency Codes
F0014	Payment Terms
F0022	Tax Rules
F0041Z1	Transaction Control File
F0101	Address Book Master
F0111	Address Book - Who's Who
F0116	Address By Date
F0150	Address Organization Structure Master File - OSTP, PA8, AN8
F0301	Customer Master
F0401	Supplier Master
F0901	Account Master
F0907	Chart of Accounts Format
F4001Z	Order Headings
F40073	Preferences Hierarchy File
F4008	Tax Areas
F4009	M&D Constants
F40095	Default Locations
F4013	Order Processing Cross Reference File
F4011Z	Order Details
F40205	Line Type Constants
F4070	Price Adjustment Schedule (only if Advanced Pricing)
F4071	Price Adjustment Type (only if Advanced Pricing)
F4072	Price Adjustment Detail (only if Advanced Pricing)
F4075	Price Variable Table (only if Advanced Pricing)
F4092	Group Code Key Definition Table
F4094	Item/Customer Key ID Master File
F41001	Branch Constants
F41002	Item Units of Measure Conversion Factors
F41003	Unit of Measure standard conversion
F4101	Item Master
F4102	Item Branch File
F4106	Base Price
F4201	Sales Order Header File
F4207	Inventory Pricing Rules
F4208	Customer Pricing Rules
F4801	Work Order Master File

Typical Customer Configuration

There are many ways to set up your OneWorld configuration. This section provides a few examples and recommendations.

You should follow J.D. Edwards typical setup and naming standards wherever possible, unless there is a strong business case that supports the need to change. Following the typical setup and naming standards enhances the likelihood of success and minimizes confusion when communicating with individuals outside the core project team who are not aware of your specific configuration setup.

If you want to customize your configuration, you should change only the descriptions with the typical setup, not the names. Upgrades will be easier with fewer manual steps if you use the naming standards J.D. Edwards recommends.

An environment description is important because the description appears on the environment list of the OneWorld signon screen where the user chooses the environment. The environment description should indicate the following:

- Path code
- Data type (such as production, test, or CRP)
- Data location
- Location that batch applications will execute

Understanding the Typical OneWorld Configuration

This chapter helps you understand a typical OneWorld configuration from the workstation's perspective. That is, the mappings presented in this chapter are for workstation to enterprise server mappings, and not server to server mappings (for example, mappings for the server map are not listed). Although J.D. Edwards ships OneWorld with a typical configuration, you can change the configuration.

For information about a typical OneWorld configuration, see the following:

- ❑ Understanding your environments
- ❑ Understanding typical data sources
- ❑ Understanding the configuration at a glance

Understanding Your Environments

You should understand the basic environments of a OneWorld configuration. For each environment, this chapter describes:

- The path code
- The purpose
- The object mappings

All OneWorld environments you receive from J.D. Edwards share the following data sources:

- System - B7333
- Object Librarian - B7333
- Data Dictionary - B7333

The following tables show the information for OneWorld environments. The tables are separated according to enterprise server platform. The UNIX and Windows NT platforms are presented first, and then the AS/400 platform. The AS/400 tables assume that you run OneWorld in coexistence with WorldSoftware.

Shared Data Sources for UNIX and Windows NT Environments

Production Environment (PD7333)

Path Code	PD7333
Central Objects data source	Central Objects - PD7333
Versions Tables data source	Versions - PD7333
Business Data data source	Business Data - PROD
Next Numbers data source	Control Tables - Production
Menu/UDCs data source	OneWorld Local - PD7333[1]

1 After the installation or the upgrade finishes, you will need to replicate the menus and UDCs from the associated Control Tables data source.

Conference Room Pilot Environment (PY7333)

Path Code	PY7333
Central Objects data source	Central Objects - PY7333
Versions Tables data source	Versions - PY7333
Business Data data source	Business Data - CRP
Next Numbers data source	Control Tables - CRP
Menu/UDCs data source	OneWorld Local - PY7333[1]

1 After the installation or the upgrade finishes, you will need to replicate the menus and UDCs from the associated Control Tables data source.

Test Environment (TS7333)

Path Code	PY7333
Central Objects data source	Central Objects - PY7333
Versions Tables data source	Versions - PY7333

Business Data data source	Business Data - TEST
Next Numbers data source	Control Tables - Test
Menu/UDCs data source	Control Tables - Test

Development Environment (DV7333)

Path Code	DV7333
Central Objects data source	Central Objects - DV7333
Versions Tables data source	Versions - DV7333
Business Data data source	Business Data - TEST
Next Numbers data source	Control Tables - Test
Menu/UDCs data source	OneWorld Local - DV7333[1]

1 After the installation or the upgrade finishes, you will need to replicate the menus and UDCs from the associated Control Tables data source.

Pristine Environment (JD7333)

Path Code	JD7333
Central Objects data source	Central Objects - JD7333
Versions Tables data source	Versions - JD7333
Business Data data source	Business Data - JDE
Next Numbers data source	Business Data - JDE
Menu/UDCs data source	OneWorld Local - JD7333

Planner Environment (JDEPLAN)

Path Code	PLANNER
Central Objects data source	Not applicable
Versions Tables data source	OneWorld Local
Business Data data source	OneWorld Planner - B7333
Next Numbers data source	OneWorld Planner - B7333
Menu/UDCs data source	OneWorld Local

Deployment Environment (DEP7333)

Path Code	PLANNER
Central Objects data source	Not applicable
Versions Tables data source	Versions - PD7333
Business Data data source	OneWorld Local
Next Numbers data source	OneWorld Local
Menu/UDCs data source	OneWorld Local[1]

1 You should not replicate menus and UDCs from another data source.

Shared Data Sources for AS/400 Environments

The following data source lists assume you are running OneWorld in coexistence with WorldSoftware.

Production Environment (PD7333)

Path Code	PD7333
Central Objects data source	Central Objects - PD7333
Versions Tables data source[1]	Versions - PD7333
Business Data data source[1]	Business Data - PROD
Next Numbers data source	AS/400 COMMON - Production
Menu/UDCs data source	OneWorld Local - PD7333[2]

1 For tables that contain BLOBs, a separate data source marked Do Not Translate is used. For example, you can map F983051 to Versions - PD7333 - DNT. The Versions data source and Business Data data source are affected.

2 After the installation or upgrade finishes, you will need to replicate the UDCs from the associated AS/400 COMMON data source and the menus from the associated control table data source (Control Tables - Production, Control Tables - CRP, and Control Tables - Test).

Conference Room Pilot Environment (PY7333)

Path Code	PY7333
Central Objects data source	Central Objects - PY7333
Versions Tables data source[1]	Versions - PY7333
Business Data data source[1]	Business Data - CRP
Next Numbers data source	AS/400 COMMON - CRP
Menu/UDCs data source	OneWorld Local - PY7333[2]

1 For tables that contain BLOBs, a separate data source marked Do Not Translate is used. For example, you can map F983051 to Versions - PD7333 - DNT. The Versions data source and Business Data data source are affected.

2 After the installation or upgrade finishes, you will need to replicate the UDCs from the associated AS/400 COMMON data source and the menus from the associated control table data source (Control Tables - Production, Control Tables - CRP, and Control Tables - Test).

Test Environment (TS7333)

Path Code	PY7333
Central Objects data source	Central Objects - PY7333
Versions Tables data source[1]	Versions - PY7333
Business Data data source[1]	Business Data - TEST
Next Numbers data source	AS/400 COMMON - Test
Menu/UDCs data source	Control Tables - Test

1 For tables that contain BLOBs, a separate data source marked Do Not Translate is used. For example, you can map F983051 to Versions - PD7333 - DNT. The Versions data source and Business Data data source are affected.

Development Environment (DV7333)

Path Code	DV7333
Central Objects data source	Central Objects - DV7333
Versions Tables data source[1]	Versions - DV7333
Business Data data source[1]	Business Data - TEST

Next Numbers data source	Control Tables - Test
Menu/UDCs data source	OneWorld Local - DV7333[2]

1 For tables that contain BLOBs, a separate data source marked Do Not Translate is used. For example, you can map F983051 to Versions - PD7333 - DNT. The Versions data source and Business Data data source are affected.

2 After the installation or upgrade finishes, you will need to replicate the UDCs from the associated AS/400 COMMON data source and the menus from the associated control table data source (Control Tables - Production, Control Tables - CRP, and Control Tables - Test).

Pristine Environment (JD7333)

Path Code	JD7333
Central Objects data source	Central Objects - JD7333
Versions Tables data source[1]	Versions - JD7333
Business Data data source[1]	Business Data - JDE
Next Numbers data source	Business Data - JDE
Menu/UDCs data source	OneWorld Local - JD7333[2]

1 For tables that contain BLOBs, a separate data source marked Do Not Translate is used. For example, you can map F983051 to Versions - PD7333 - DNT. The Versions data source and Business Data data source are affected.

2 After the installation or upgrade finishes, you will need to replicate the menus and UDCs from Business Data - JDE to the OneWorld Local - JD7333 data source.

Planner Environment (JDEPLAN)

Path Code	PLANNER
Central Objects data source	Not applicable
Versions Tables data source[1]	OneWorld Local
Business Data data source[1]	OneWorld Planner - B7333
Next Numbers data source	OneWorld Planner - B7333
Menu/UDCs data source	OneWorld Local[2]

1 For tables that contain BLOBs, a separate data source marked Do Not Translate is used. For example, you can map F983051 to Versions - PD7333 - DNT. The Versions data source and Business Data data source are affected.

2 You should not replicate menus and UDCs from another data source.

Deployment Environment (DEP7333)

Path Code	PLANNER
Central Objects data source	Not applicable
Versions Tables data source[1]	Versions - PD7333
Business Data data source[1]	OneWorld Local
Next Numbers data source	OneWorld Local
Menu/UDCs data source	OneWorld Local[2]

1 For tables that contain BLOBs, a separate data source marked Do Not Translate is used. For example, you can map F983051 to Versions - PD7333 - DNT. The Versions data source and Business Data data source are affected.

2 You should not replicate menus and UDCs from another data source.

Remote Environments

Remote environment names are preceded by a 3-character location code followed by a J for Java application server or W for Windows terminal server.

For example:

- *xxx*JPD7333: Indicates a Java application server for the PD7333 environment at location *xxx*
- *xxx*WTS7333: Indicates a Windows terminal server for the TS7333 environment at location *xxx*

The path codes and data sources for remote environments are identical to those for base environments.

The PD7333 Environment

The following tables list the data sources according to the enterprise and data server environment in which the workstation is operating:

- PD7333 - Object Mappings (non-AS/400)

- PD7333 - Object Mappings (AS/400)

Path Code PD7333

Purpose PD7333 is the live production environment for your end users who will have a tested and released package on their machine. At a minimum, replicated data includes user defined codes, menus, and the data dictionary. Batch applications execute on the server. Eventually there might be more than one production environment established for different types of distributed data, logic, and modes of processing.

PD7333 - Object Mappings (non-AS/400)

PD7333 Object Mappings (non-AS/400)	
Mapping	**Explanation**
Business Data - PROD	This is the default object mapping. Tables that are not specifically mapped by other data sources use this data source as their default mapping. This mapping includes the table: • F00165 - Media objects storage
Central Objects - PD7333	Maps to the central object tables including: • F98950 - User Overrides
Control Tables - Production	Maps to the next number tables.
Data Dictionary - B7333	Maps to the data dictionary tables. Also maps to the data dictionary media object text in the data structure: • GT92002 - Data Dictionary - Glossary Information
Logic Data Source	Maps to the machine on which batch applications are run.
Object Librarian - B7333	Maps to the object librarian tables. Also maps to the object librarian media object text in these data structures: • GT9860A - Object Librarian Generic Text Structure • GT9862A - Business Function Notes • GT98DSA - Data Structure Notes - Structure and Item • GT98DSB - Data Structure Notes - Structure and Item • GT98TMPL - Media Object Templates
OneWorld Local - PD7333	Maps to the User Defined Codes (UDCs) and menus that are replicated from the data source: Control Tables - Production.
System - B7333	Maps to the system tables.
Versions - PD7333	Maps to the version tables including: • F983051 - Versions List • F98306 - Processing Option Text

PD7333 - Object Mappings (AS/400)

PD7333 Object Mappings (AS/400)	
Mapping	**Explanation**
AS400 COMMON - Production	Maps to the next number tables. This mapping is required only for coexistence environments.
Business Data - PROD	This is the default object mapping. Tables that are not specifically mapped by other data sources use this data source as their default mapping. This mapping includes the following table: • F00165 - Media objects storage
Business Data - PROD - DNT	Maps to the Do Not Translate (DNT) Workflow Activity Specifications table (F98811).
Central Objects - PD7333	Maps to the central object tables including: • F98950 - User Overrides
Control Tables - Production	Maps to the next number tables. This mapping is required only for noncoexistence environments.
Data Dictionary - B7333	Maps to the data dictionary tables. Also maps to the data dictionary media object text in the following data structure: • GT92002 - Data Dictionary - Glossary Information
Logic Data Source	Maps to the machine on which batch applications are run.
Object Librarian - B7333	Maps to the object librarian tables. Also maps to the object librarian media object text in the following data structures: • GT9860A - Object Librarian Generic Text Structure • GT9862A - Business Function Notes • GT98DSA - Data Structure Notes - Structure and Item • GT98DSB - Data Structure Notes - Structure and Item • GT98TMPL - Media Object Templates
OneWorld Local - PD7333	Maps to the User Defined Codes (UDCs) and menus that are replicated as follows: • UDCs - Coexistence. Replicated from AS400 COMMON - Production • UDCs - Noncoexistence. Replicated from Control Tables - Production • Menus. Replicated from Control Tables - Production

Mapping	Explanation
System - B7333	Maps to the system tables in the WorldSoftware security library.
System - B7333 - DNT	Maps to the Do Not Translate (DNT) system tables including: • F98DRPCN - Data Replication Pending Change Notification • F98DRLOG - Data Replication Change Log • F98OWSEC - OneWorld Security
Versions - PD7333	Maps to the version tables including; • F98306 - Processing Option Text
Versions - PD7333 - DNT	Maps to the Do Not Translate (DNT) version tables including: • F983051 - Versions List

The TS7333 Environment

The following tables list the data sources according to the enterprise and data server environment in which the workstation is operating:

- TS7333 - Object Mappings (non-AS/400)
- TS7333 - Object Mappings (AS/400)

Path Code PY7333

Purpose You can use the TS7333 environment to test:

- The software during a conference room pilot (CRP)
- Modifications you made in the development path code and transferred to PY7333, but have not yet transferred to your production path code and, therefore, have not been released to end users

You can also test conversions in this environment (converting foreign tables into J.D. Edwards tables). If you are authorized to use the test environment, you can safely change data and versions that will never be copied to production. You can also conduct training classes in this environment.

Customer application testing and conversion testing populate this environment. After you run the Installation Workbench, no business data exists until you enter it. On a routine basis (such as monthly or quarterly), you should refresh this environment from PD7333, which represents your production data.

In this environment, the menus and UDCs are not replicated.

TS7333 - Object Mappings (non-AS/400)

TS7333 Object Mappings (non-AS/400)	
Mapping	**Explanation**
Business Data - Test	This is the default object mapping. Tables that are not specifically mapped by other data sources will use this data source as their default mapping. This mapping includes the following table: • F00165 - Media objects storage
Central Objects - PY7333	Maps to the central object tables including: • F98950 - User Overrides
Control Tables - Test	Maps to the next number tables, User Defined Codes (UDCs) and menus.
Data Dictionary - B7333	Maps to the data dictionary tables. Also maps to the data dictionary media object text in the following data structure: • GT92002 - Data Dictionary - Glossary Information
Logic Data Source	Maps to the machine on which batch applications are run.
Object Librarian - B7333	Maps to the object librarian tables. Also maps to the object librarian media object text in the following data structures: • GT9860A - Object Librarian Generic Text Structure • GT9862A - Business Function Notes • GT98DSA - Data Structure Notes - Structure and Item • GT98DSB - Data Structure Notes - Structure and Item • GT98TMPL - Media Object Templates
System - B7333	Maps to the system tables.
Versions - PY7333	Maps to the version tables including: • F983051 - Versions List • F98306 - Processing Option Text

TS7333 - Object Mappings (AS/400)

TS7333 Object Mappings (AS/400)	
Mapping	**Explanation**
AS400 COMMON - Test	Maps to the next number tables and user defined codes (UDCs) tables. This mapping is required only for coexistence environments.
Business Data - Test	This is the default object mapping. Tables that are not specifically mapped by other data sources will use this data source as their default mapping. This mapping includes the following table: • F00165 - Media objects storage
Business Data - Test - DNT	Maps to the Do Not Translate (DNT) Workflow Activity Specifications table (F98811).
Central Objects - PY7333	Maps to the central object tables including: • F98950 - User Overrides
Control Tables - Test	For coexistence, maps to the menus tables. For noncoexistence, maps to next numbers tables, menus tables, and the user defined codes (UDCs) tables.
Data Dictionary - B7333	Maps to the data dictionary tables. Also maps to the data dictionary media object text in the following data structure: • GT92002 - Data Dictionary - Glossary Information
Logic Data Source	Maps to the machine on which batch applications are run.
Object Librarian - B7333	Maps to the object librarian tables. Also maps to the object librarian media object text in the following data structures: • GT9860A - Object Librarian Generic Text Structure • GT9862A - Business Function Notes • GT98DSA - Data Structure Notes - Structure and Item • GT98DSB - Data Structure Notes - Structure and Item • GT98TMPL - Media Object Templates
System - B7333	Maps to the system tables in the WorldSoftware security library.
System - B7333 - DNT	Maps to the Do Not Translate (DNT) system tables including: • F98DRPCN - Data Replication Pending Change Notification • F98DRLOG - Data Replication Change Log • F98OWSEC - OneWorld Security
Versions - PY7333	Maps to the version tables including: • F98306 - Processing Option Text

Mapping	Explanation
Versions - PY7333 - DNT	Maps to the Do Not Translate (DNT) version tables including: • F983051 - Versions List

The PY7333 Environment

The following tables list the data sources according to the enterprise and data server environment in which the workstation is operating:

- PY7333 - Object Mappings (non-AS/400)
- PY7333 - Object Mappings (AS/400)

Path Code	PY7333
Purpose	The Conference Room Pilot (CRP) environment, PY7333, is the staging environment for production. Constants tables and master tables (such as company constants, fiscal date patterns, and item master) are populated with customer data during the CRP process. You copy the tables to the production environment before you go live. After you run the Installation Workbench, no business data exists until you enter it. When appropriate, you should refresh the test data from PD7333, which represents your production data.

PY7333 - Object Mappings (non-AS/400)

PY7333 Object Mappings (non-AS/400)	
Mapping	**Explanation**
Business Data - CRP	This is the default object mapping. Tables that are not specifically mapped by other data sources will use this data source as their default mapping. This mapping includes the following table: • F00165 - Media objects storage
Central Objects - PY7333	Maps to the central object tables, including: • F98950 - User Overrides
Control Tables - CRP	Maps to the next number tables.
Data Dictionary - B7333	Maps to the data dictionary tables. Also maps to the data dictionary media object text in the following data structure: • GT92002 - Data Dictionary - Glossary Information
Logic Data Source	Maps to the machine on which batch applications are run.
Object Librarian - B7333	Maps to the object librarian tables. Also maps to the object librarian media object text in the data structures: • GT9860A - Object Librarian Generic Text Structure • GT9862A - Business Function Notes • GT98DSA - Data Structure Notes - Structure and Item • GT98DSB - Data Structure Notes - Structure and Item • GT98TMPL - Media Object Templates
OneWorld Local - PY7333	Maps to the User Defined Codes (UDCs) and menus that are replicated from the data source: Control Tables - CRP.
System - B7333	Maps to the system tables.
Versions - PY7333	Maps to the version tables including: • F983051 - Versions List • F98306 - Processing Option Text

PY7333 - Object Mappings (AS/400)

PY7333 Object Mappings (AS/400)	
Mapping	**Explanation**
AS400 COMMON - CRP	Maps to the next number tables. This mapping is required only for coexistence environments.
Business Data - CRP	This is the default object mapping. Tables that are not specifically mapped by other data sources will use this data source as their default mapping. This mapping includes the following table: • F00165 - Media objects storage
Business Data - CRP - DNT	Maps to the Do Not Translate (DNT) Workflow Activity Specifications table (F98811).
Central Objects - PY7333	Maps to the central object tables, including: • F98950 - User Overrides
Control Tables - CRP	Maps to the next number tables. This mapping is required only for noncoexistence environments.
Data Dictionary - B7333	Maps to the data dictionary tables. Also maps to the data dictionary media object text including the data structure: • GT92002 - Data Dictionary - Glossary Information
Logic Data Source	Maps to the machine on which batch applications are run.
Object Librarian - B7333	Maps to the object librarian tables. Also maps to the object librarian media object text in the data structures: • GT9860A - Object Librarian Generic Text Structure • GT9862A - Business Function Notes • GT98DSA - Data Structure Notes - Structure and Item • GT98DSB - Data Structure Notes - Structure and Item • GT98TMPL - Media Object Templates
OneWorld Local - PY7333	Maps to the User Defined Codes (UDCs) and menus that are replicated as follows: • UDCs - Coexistence. Replicated from AS400 COMMON - CRP • UDCs - Noncoexistence. Replicated from Control Tables - CRP • Menus. Replicated from Control Tables - CRP
System - B7333	Maps to the system tables in the WorldSoftware security library.

Mapping	Explanation
System - B7333 - DNT	Maps to the Do Not Translate (DNT) system tables including: • F98DRPCN - Data Replication Pending Change Notification • F98DRLOG - Data Replication Change Log • F98OWSEC - OneWorld Security
Versions - PY7333	Maps to the version tables including: • F98306 - Processing Option Text
Versions - PY7333 - DNT	Maps to the Do Not Translate (DNT) version tables including: • F983051 - Versions List

The DV7333 Environment

If you are not developing custom modifications, you do not need this environment.

The following tables list the data sources according to the enterprise and data server environment in which the workstation is operating:

- DV7333 - Object Mappings (non-AS/400)
- DV7333 - Object Mappings (AS/400)

Path Code DV7333

Purpose The development environment (DV7333) is the testing environment for development objects. This environment shares the test data that TS7333 uses. Developers sign onto this environment to modify objects and test before transferring changed objects to the PD7333 path code.

Once you have transferred objects into PD7333 a user can install a recent PRD package that has not been released to end users and sign onto either CRP or TST for additional testing.

DV7333 - Object Mappings (non-AS/400)

DV7333 Object Mappings (non-AS/400)	
Mapping	**Explanation**
Business Data - Test	This is the default object mapping. Tables that are not specifically mapped by other data sources will use this data source as their default mapping. This mapping includes the following table: • F00165 - Media objects storage
Central Objects - DV7333	Maps to the central object tables including: • F98950 - User Overrides
Control Tables - Test	Maps to the next number tables.
Data Dictionary - B7333	Maps to the data dictionary tables. Also maps to the data dictionary media object text including the following data structure: • GT92002 - Data Dictionary - Glossary Information
Logic Data Source	Maps to the machine on which batch applications are run.
Object Librarian - B7333	Maps to the object librarian tables. Also maps to the object librarian media object text in the data structures: • GT9860A - Object Librarian Generic Text Structure • GT9862A - Business Function Notes • GT98DSA - Data Structure Notes - Structure and Item • GT98DSB - Data Structure Notes - Structure and Item • GT98TMPL - Media Object Templates
OneWorld Local - DV7333	Maps to the User Defined Codes (UDCs) and menus that are replicated from the data source: Control Tables - Test.
System - B7333	Maps to the system tables.
Versions - DV7333	Maps to the version tables including: • F983051 - Versions List • F98306 - Processing Option Text

DV7333 - Object Mappings (AS/400)

DV7333 Object Mappings (AS/400)	
Mapping	**Explanation**
AS/400 COMMON - Test	Maps to the next number tables. This mapping is required only for coexistence environments.
Business Data - Test	This is the default object mapping. Tables that are not specifically mapped by other data sources will use this data source as their default mapping. This mapping includes the table: • F00165 - Media objects storage
Business Data - CRP - DNT	Maps to the Do Not Translate (DNT) Workflow Activity Specifications table (F98811).
Central Objects - DV7333	Maps to the central object tables including: • F98950 - User Overrides
Control Tables - Test	Maps to the next number tables. This mapping is required only for noncoexistence environments.
Data Dictionary - B7333	Maps to the data dictionary tables. Also maps to the data dictionary media object text including the data structure: • GT92002 - Data Dictionary - Glossary Information
Logic Data Source	Maps to the machine on which batch applications are run.
Object Librarian - B7333	Maps to the object librarian tables. Also maps to the object librarian media object text in the data structures: • GT9860A - Object Librarian Generic Text Structure • GT9862A - Business Function Notes • GT98DSA - Data Structure Notes - Structure and Item • GT98DSB - Data Structure Notes - Structure and Item • GT98TMPL - Media Object Templates
OneWorld Local - DV7333	Maps to the User Defined Codes (UDCs) and menus that are replicated as follows: • UDCs - Coexistence. Replicated from AS400 COMMON - Test • UDCs - Noncoexistence. Replicated from Control Tables - Test • Menus. Replicated from Control Tables - Test
System - B7333	Maps to the system tables in the WorldSoftware security library.

Mapping	Explanation
System - B7333 - DNT	Maps to the Do Not Translate (DNT) system tables including; • F98DRPCN - Data Replication Pending Change Notification • F98DRLOG - Data Replication Change Log • F98OWSEC - OneWorld Security
Versions - DV7333	Maps to the version tables including: • F98306 - Processing Option Text
Versions - DV7333 - DNT	Maps to the Do Not Translate version tables including: • F983051 - Versions List

The JD7333 Environment

The following tables list the data sources according to the enterprise and data server environment in which the workstation is operating:

- JD7333 - Object Mappings (non-AS/400)
- JD7333 - Object Mappings (AS/400)

Path Code JD7333

Purpose You can use the JD7333 environment to test J.D. Edwards pristine (unaltered) objects with J.D. Edwards demonstration data. You can also use this environment for training classes. You must have this environment to compare modified objects to pristine objects.

When you encounter a software problem that J.D. Edwards Worldwide Customer Support cannot duplicate, they will ask you to sign on to the pristine environment to duplicate the problem. On a routine basis (such as monthly or quarterly), you should refresh the J.D. Edwards data this environment uses with the J.D. Edwards demonstration data shipped with the software.

JD7333 - Object Mappings (non-AS/400)

JD7333 Object Mappings (non-AS/400)	
Mapping	**Explanation**
Business Data - JDE	This is the default object mapping. Tables that are not specifically mapped by other data sources will use this data source as their default mapping. This mapping includes the following table: • F00165 - Media objects storage
Data Dictionary - B7333	Maps to the data dictionary tables. Also maps to the data dictionary media object text including the following data structure: • GT92002 - Data Dictionary - Glossary Information
Logic Data Source	Maps to the machine on which batch applications are run.
Object Librarian - B7333	Maps to the object librarian tables. Also maps to the object librarian media object text in the following data structures: • GT9860A - Object Librarian Generic Text Structure • GT9862A - Business Function Notes • GT98DSA - Data Structure Notes - Structure and Item • GT98DSB - Data Structure Notes - Structure and Item • GT98TMPL - Media Object Templates
OneWorld Local - JD7333	Maps to the User Defined Codes (UDCs) and menus that are replicated from the following data source: • Business Data - JDE
System - B7333	Maps to the system tables.
Versions - JD7333	Maps to the version tables including: • F983051 - Versions List • F98306 - Processing Option Text

JD7333 - Object Mappings (AS/400)

JD7333 Object Mappings (AS/400)	
Mapping	**Explanation**
Business Data - JDE	This is the default object mapping. Tables that are not specifically mapped by other data sources use this data source as their default mapping. This mapping includes the following table: • F00165 - Media objects storage

Mapping	Explanation
Business Data - JDE - DNT	Maps to the Do Not Translate (DNT) Workflow Activity Specifications table (F98811).
Central Objects - PD7333	Maps to the central object tables, including: • F98950 - User Overrides
Data Dictionary - B7333	Maps to the data dictionary tables. Also maps to the data dictionary media object text including the following data structure: • GT92002 - Data Dictionary - Glossary Information
Local Data Source	Maps to the machine on which batch applications are run.
Object Librarian - B7333	Maps to the object librarian tables. Also maps to the object librarian media object text in the following data structures: • GT9860A - Object Librarian Generic Text Structure • GT9862A - Business Function Notes • GT98DSA - Data Structure Notes - Structure and Item • GT98DSB - Data Structure Notes - Structure and Item • GT98TMPL - Media Object Templates
OneWorld Local - JD7333	Maps to the User Defined Codes (UDCs) and menus that are replicated as follows: • UDCs: Replicated from Business Data - JDE • Menus: Replicated from Business Data - JDE
System - B7333	Maps to the system tables in the WorldSoftware security library.
System - B7333 - DNT	Maps to the Do Not Translate (DNT) system tables including: • F98DRPCN - Data Replication Pending Change Notification • F98DRLOG - Data Replication Change Log • F98OWSEC - OneWorld Security
Versions - JD7333	Maps to the version tables including: • F98306 - Processing Option Text
Versions - JD7333 - DNT	Maps to the Do Not Translate version tables including: • F983051 - Versions List

What You Should Know About

Environment control tables and mappings	TS7333 and DV7333 share a set of control tables. Pristine has its own set of control tables that are included in Business Data - JDE. You do not map central objects through Object Configuration Manager because you handle them through Object Path Master. All environments (regardless of path code) share the same Object Librarian tables.

Understanding Typical Data Sources

If the enterprise server has Oracle Server or Microsoft SQL Server and is a more powerful machine than the deployment server, for performance reasons J.D. Edwards recommends that you put all data sources (except OneWorld Local) on the enterprise server instead of the deployment server.

See the *OneWorld Oracle Consideration Guide* and the *OneWorld SQL Server Consideration Guide* for information about the instances in which each data source should reside.

Data sources are listed according to enterprise and data server type:

The following data sources are grouped according to the enterprise and data server environment in which the workstation is operating:

- Understanding non-AS/400 Data Sources
- Understanding AS/400 Data Sources

Unless otherwise stated, the database name for the data source is the following:

For servers with Oracle Server	The connect string for the database.
For servers with Microsoft SQL Server	The data source name. For example, for data source Business Data - PROD, the database name would also be Business Data - PROD.
For the AS/400	The data source name. For example, for data source Versions - PD7333 - DNT, the database name would also be Versions -PD7333 - DNT.

Understanding Non-AS/400 Data Sources

The following shows the data source name followed by the owner and the purpose:

Understanding Non-AS/400 Data Sources	
Data Source Name	**Owner and Purpose**
Business Data - CRP	The owner is CRPDTA. This is the CRP business data. Before going live, you should copy much of this data to Business Data - PROD.
Business Data - JDE	The owner is PRISTDTA. This is the pristine data shipped with OneWorld.
Business Data - PROD	The owner is PRODDTA. This is the production business data.
Business Data - TEST	The owner is TESTDTA. This is the test data entered during CRP or that is converted from non-J.D. Edwards systems.
Central Objects - PY7333	The owner is PY7333. This is the central objects data source associated with the PY7333 path code. After you make and test the modifications in this path code, transfer them to the PD7333 path code.
Central Objects - DV7333	The owner is DV7333. This is the central objects data source associated with the DV7333 path code. After you make and test the modifications in this path code, transfer them to the PD7333 path code.
Central Objects - JD7333	The owner is JD7333. This is the central objects data source associated with the JD7333 path code.

Data Source Name	Owner and Purpose
Central Objects - PD7333	The owner is PD7333. This is the central objects data source that goes with the PD7333 path code. Transfer objects into this data source after you have tested them in path code DV7333.
Control Tables - CRP	The owner is CRPCTL. These are the control tables used in the CRP environment.
Control Tables - Prod	The owner is PRODCTL. These are the control tables used in the production environment.
Control Tables - Test	The owner is TESTCTL. These are the control tables used in the test environment.
Data Dictionary - B7333	The owner is DD7333. This is the one data dictionary that all environments use.
LOCAL	There is no owner. This data source defines the local machine to OneWorld. Use it to override reports to the workstation.
machine name	The owner is SVM7333. This defines the logic host to OneWorld.
machine name - Server Map	The owner is SVM7333. This is the server map for the logic server.
Object Librarian - B7333	The owner is OBJ7333. This is the Object Librarian, which is OneWorld release specific.
OneWorld Local	There is no owner. This is replicated data found in the JDEB7 Microsoft Access database. This is used only in the planner.

Data Source Name	Owner and Purpose
OneWorld Local - PY7333	There is no owner. This contains the User Defined Codes (UDCs) and menus that are replicated from the data source: Control Tables - CRP.
OneWorld Local - DV7333	There is no owner. This contains the User Defined Codes (UDCs) and menus that are replicated from the data source: Control Tables - Test.
OneWorld Local - JD7333	There is no owner. This contains the User Defined Codes (UDCs) and menus that are replicated from the data source: Business Data - JDE.
OneWorld Local - PD7333	There is no owner. This contains the User Defined Codes (UDCs) and menus that are replicated from the data source: Control Tables - Production.
System - B7333	The owner is SYS7333. This is the one set of system tables that all environments use.
Versions - PY7333	The owner is JD7333. This contains the versions list and processing option text tables for the PY7333 environment.
Versions - DV7333	The owner is DV7333. This contains the versions list and processing option text tables for the DV7333 environment.
Versions - JD7333	The owner is JD7333. This contains the versions list and processing option text tables for the JD7333 environment.
Versions - PD7333	The owner is PD7333. This contains the versions list and processing option text tables for the PD7333 environment.

Understanding AS/400 Data Sources

The following shows the data source name followed by the owner and the purpose:

Understanding AS/400 Data Sources	
Data Source Name	**Owner and Purpose**
AS/400 COMMON - CRP	The default library is CRPCOM. This is the common library for user defined codes used by the PY7333 environment. This mapping only applies to coexistence environments.
AS/400 COMMON - Production	The default library is CLTCOM. This is the common library for user defined codes used by the Production environment. This mapping only applies to coexistence environments.
AS/400 COMMON - Test	The default library is TESTCOM. This is the common library for user defined codes, which are shared by production and CRP. This mapping only applies to coexistence environments.
Business Data - CRP	The default library is CRPDTA. This is the CRP business data. Before going live, you should copy much of this data to Business Data - PROD.
Business Data - PROD	The default library is PRODDTA. This is the production business data.
Business Data - TEST	The default library is TESTDTA. This is the test data entered during CRP or that is converted from non-J.D. Edwards systems.
Business Data - JDE	The default library is PRISTDTA. This is the pristine data shipped with OneWorld. This database includes all application data, as well as next numbers, user defined codes, and menus.

Data Source Name	Owner and Purpose
Central Objects - PY7333	The owner is PY7333. This is the central objects data source associated with the PY7333 path code. After you make and test the modifications in this path code, transfer them to the PD7333 path code.
Central Objects - DV7333	The owner is DV7333. This is the central objects data source associated with the DV7333 path code. After you make and test the modifications in this path code, transfer them to the PD7333 path code.
Central Objects - JD7333	The owner is JD7333. This is the central objects data source associated with the JD7333 path code.
Central Objects - PD7333	The owner is PD7333. This is the central objects data source associated with the PD7333 path code. Transfer objects into this data source after you have tested them in path code DV7333.
Control Tables - Prod	The default library is PRODCTL This is the master copy of OneWorld menus. Although menu tables also exist in WorldSoftware, for this release of OneWorld they cannot be shared between WorldSoftware and OneWorld.
Control Tables - Test	The default library is TESTCTL. These are the control tables used in the test environment.
Data Dictionary - B7333	The default library is DD7333. This is the one data dictionary that all environments use.
LOCAL	There is no library. This defines the local machine to OneWorld, and you use it to override reports to the workstation.
machine name	The default library is B7333CTL. This points to the OneWorld logic server which is defined by the variable *machine name*.

Data Source Name	Owner and Purpose
machine name - Server Map	The default library is B7333CTL. This points to the server map tables on the OneWorld logic server which is defined by the variable *machine name*.
Object Librarian - B7333	The default library is OL7333. This is the Object Librarian, which is OneWorld release specific.
OneWorld Local	There is no library. This points to replicated data found in the JDEB7 Microsoft Access database.
System - B7333	The default library is SYS7333 or the existing WorldSoftware security library. This points to the one set of system tables, except for the Do Not Translate (DNT) tables that all environments use.
System - B7333 - DNT	The default library is SYS7333 or the existing WorldSoftware security library. This points to the Do Not Translate (DNT) system tables including the Data Replication Pending Change Notification (F98DRPCN), Data Replication Change Log (F98DRLOG), and OneWorld Security (F98OWSEC) tables.
Versions - PY7333	The default library is PY7333DNT. This points to the processing option text table (F98306).
Versions - PY7333 - DNT	The default library is PY7333DNT. This points to the version list table (F983051) that contains BLOBs that cannot be translated.
Versions - DV7333	The default library is DV7333DNT. This points to the processing option text table (F98306).
Versions - DV7333 - DNT	The default library is DV7333DNT. This points to the version list table (F983051) that contains BLOBs that cannot be translated.

Data Source Name	Owner and Purpose
Versions - JD7333	The default library is JD7333DNT. This points to the processing option text table (F98306).
Versions - JD7333 - DNT	The default library is JD7333DNT. This points to the version list table (F983051) that contains BLOBs that cannot be translated.
Versions - PD7333	The default library is PD733DNT. This points to the processing option text table (F98306).
Versions - PD7333 - DNT	The default library is PD7333DNT. This points to the version list table (F983051) that contains BLOBs that cannot be translated.

Understanding the Configuration at a Glance

The following table summarizes the data, environments, central objects (path codes), and packages that you need:

Requirement	Explanation
Data	You might have several sets of data: • Conference Room Pilot (CRP) business data • Conference Room Pilot (CRP) control tables • Data Dictionary (all environments share) • J.D. Edwards pristine data • Production business data • Production control tables (used by the PD7333 environment) • Object Librarian (all environments share) • System (technical data all environments share) • Test business data • Test control tables (used by TS7333 and DV7333 environments) • Versions For a coexistence configuration, you have these additional sets of data: • AS/400 COMMON - (Production, Test, and CRP) • Business Data - DNT (for each path code) • Versions - DNT (for each path code)

Requirement	Explanation
Central Objects (Path Codes)	You should have the following sets of central objects or path codes. These path codes are release specific: • PY7333 central objects • DV7333 central objects • JD7333 central objects • PD7333 central objects
Environments	The installation process defines environments: • PY7333 - CRP objects (possibly testing a package you have not released to production users), with data mapped to CRP. The path code is PY7333 with table objects mapped to CRP data. • DV7333 - Development objects with test data. The path code is DV7333, with table objects mapped to test data. • JD7333 - Pristine objects with J.D. Edwards pristine demo data. The path code is JD7333, with objects mapped to J.D. Edwards pristine data • PD7333 - Production objects with production data. The path code is PD7333, with table objects mapped to production data. • TS7333 - CRP objects with test data. The path code is PY7333, with table objects mapped to test data

Requirement	Explanation
Packages	If you are not planning any development projects, you need only two path codes: PY7333 and PD7333. You should create a development path code if you plan to do extensive software modification. The fewer path codes you use, the better. With each additional path code comes version control maintenance that is not worth the effort unless there is a true reason for the additional path code. Even when making extensive software modifications, you should have only four path codes (sets of central objects): • **PY7333** This path code contains a practice set of objects that are tested during conference room pilot before transferring objects to production. It is for deploying quick fixes or making minor modifications that you will quickly transfer to production. It can also be used as a place to test modifications that were done in the development path code before taking the risk of transferring them to the production path code. • **DV7333** Use this path code for normal development. Upon successful testing, transfer the objects to your PY7333 path code, using Object Transfer, and distribute to your users through a package build and a workstation installation. • **JD7333** This is the set of pristine objects shipped from J.D. Edwards. You should not make changes to this path code other than paper fixes from J.D. Edwards. This path code is used to compare J.D. Edwards standard software to any custom solutions you have implemented in other path codes. You should keep a copy of this path code so that you have a clean copy of OneWorld in case you need to refresh anything. • **PD7333** This is the production path code. Just-in-time installations come directly from this location, and production server objects are also deployed from here. After testing software changes in PY7333, transfer them to PD7333 and then deploy the changes to your enterprise servers and workstations.

Requirement	Explanation
	All path codes share the same Object Librarian tables and the same system data source and normally the same data dictionary. The only distinct tables across path codes are the central objects/specifications (F987*), the version list (F983051), and the processing options text (F98306). At J.D. Edwards we have found that each package should have an A and B version, and that you alternate between these when you build packages. If you are using both full and partial packages, you would have four packages per path code. This setup gives you two full packages (A and B) for production, and two partial packages (A and B) for production. For example: • PD7333FA (Standard Production Full A) • PD7333FB (Standard Production Full B) • PD7333PA (Standard Production Partial A) • PD7333PB (Standard Production Partial B)

Glossary

action message. With OneWorld, users can receive messages (system-generated or user-generated) that have shortcuts to OneWorld forms, applications, and appropriate data. For example, if the general ledger post sends an action error message to a user, that user can access the journal entry (or entries) in error directly from the message. This is a central feature of the OneWorld workflow strategy. Action messages can originate either from OneWorld or from a third-party e-mail system.

alphabetic characters. Characters on the keyboard including letters of the alphabet and all other symbols (such as *, &, #), but excluding numerals 0 through 9. For example, "ABC*" is a string of alphabetic characters, but "ABC123" is not. Also referred to as an alpha character. Contrast with alphanumeric characters and numeric characters.

alphanumeric characters. The complete set of characters on the keyboard including letters of the alphabet, symbols, and numerals. For example, "ABC*123" is a string of alphanumeric characters. Contrast with alphabetic characters and numeric characters.

alternate language. A language other than English, which is designated in the user profile. A language preference code is used to set the preferred language for each user to display data for online and printed output.

applet. A small application, such as a utility program or a limited-function spreadsheet. It is generally associated with the programming language Java, and in this context refers to Internet-enabled applications that can be passed from a Web browser residing on a workstation.

application. A computer program or set of programs used to accomplish a task. In OneWorld, there are interactive applications and batch applications. Interactive applications are made up of a set of forms through which the user interacts with OneWorld. Interactive application identifiers begin with "P." For example, Address Book Revisions (P01012) is an interactive application. Batch applications run without user interaction. Reports and table conversions are examples of batch applications. Batch application identifiers begin with "R." For example, the Print Mailing Labels report (R01401) is a batch application.

application programming interface (API). A software function call that can be made from a program to access functionality provided by another program.

application server. A server in a local area network (LAN) that contains applications used by network clients.

application workspace. The area on a workstation display in which all related forms within an application appear.

architecture. The underlying design of a computer that defines data storage methods, operations, and compatibility requirements with other systems and software. It also refers to specific components of a computer system, the way they interact with each other, and the type of CPU chip that is used as the basis of a computer system.

AS/400 common. The AS/400 library that typically contains WorldSoftware control files. Can also refer to as an AS/400 Common data source used in OneWorld.

AS/400 COMMON. A data source that resides on an AS/400 and holds data that is common to the coexistent library allowing OneWorld to share information with World.

asynchronous. A method of running table conversions in which starting one conversion does not rely on another conversion's successful completion.

audit trail. The detailed, verifiable history of a processed transaction. The history consists of the original documents, transaction entries, and posting of records and usually concludes with a report.

automatic accounting instruction (AAI). A code that refers to an account in the chart of accounts. AAIs define rules for programs that automatically generate journal entries, including interfaces between the Accounts Payable, Accounts Receivable, Financial Reporting, and General Accounting systems. Each system that interfaces with the General Accounting system has AAIs. For example, AAIs can direct the General Ledger Post program to post a debit to a specific expense account and a credit to a specific accounts payable account.

base release. The first generally available software for a OneWorld release. See also release, release level.

batch. A group of similar records or transactions that the computer treats as a single unit during processing. For identification purposes, the system usually assigns each batch a unique identifier known as a batch number.

batch application. A single task or groups of tasks that the system treats as a single unit during processing. The computer performs batch applications (jobs) with little or no user interaction. Printing reports and purging files are examples.

batch control. A feature that verifies the number of transactions and the total amount in each batch that you enter into the system.

batch input. A group of transactions loaded from an external source.

batch job. A task or group of tasks you submit for processing that the system treats as a single unit during processing, for example, printing reports and purging files. The system performs a batch job with little or no user interaction.

batch processing. A method by which the system selects jobs from the job queue, processes them, and sends output to the outqueue. Contrast with interactive processing.

batch server. A server on which OneWorld batch processing requests (also called UBEs) are run instead of on a client, an application server, or an enterprise server. A batch server typically does not contain a database nor does it run interactive applications.

batch type. A code assigned to a batch job that designates to which system the associated transactions pertain, thus controlling which records are selected for processing. For example, the Post General Journal program selects for posting only unposted transaction batches with a batch type of O.

batch-of-one immediate. A transaction method that allows a client application to perform work on a client workstation, then submit the work all at once to a server application for further processing. As a batch process is running on the server, the client application can continue performing other tasks. See also direct connect, store-and-forward.

binary large object (BLOB). A database field that has no maximum size limit and holds any digitized information. This field is often used to store objects, such as graphical representations or data structures, rather than standard alphanumeric data.

binary string (BSTR). A length prefixed string used by OLE automation data manipulation functions. Binary strings are wide, double-byte (Unicode) strings on 32-bit Windows platforms.

broadcast message. 1) An e-mail message that you send to multiple recipients. 2) A message that appears on a form instead of in your mailbox.

browser. A client application that translates information sent by the Worldwide Web. A client must use a browser to receive, manipulate, and display Worldwide Web information on the desktop. Also known as a Web browser.

business function. An encapsulated set of business rules and logic that can normally be reused by multiple applications. Business functions can execute a transaction or a subset of a transaction (check inventory, issue work orders, and so on). Business functions also contain the APIs that allow them to be called from a form, a database trigger, or a non-OneWorld application. Business functions can be combined with other business functions, forms, event rules, and other components to make up an application. Business functions can be created through event rules or third-generation languages, such as C. Examples of business functions include Credit Check and Item Availability.

business function event rule. Encapsulated, reusable business logic created using through

event rules rather than C programming. Contrast with embedded event rule. See also event rule.

business view. Used by OneWorld applications to access data from database tables. A business view is a means for selecting specific columns from one or more tables whose data will be used in an application or report. It does not select specific rows and does not contain any physical data. It is strictly a view through which data can be handled.

Business View Design Aid (BDA). A OneWorld GUI tool for creating, modifying, copying, and printing business views. The tool uses a graphical user interface.

category code. A type of user defined code for which you can provide the title. For example, if you were adding a code that designated different sales regions, you could change category code 4 to Sales Region, and define E (East), W (West), N (North), and S (South) as the valid codes. Sometimes referred to as reporting codes. See also user defined code.

central objects. Objects that reside in a central location and consist of two parts: the central objects data source and central C components. The central objects data source contains OneWorld specifications, which are stored in a relational database. Central C components contain business function source, header, object, library, and DLL files and are usually stored in directories on the deployment server. Together they make up central objects.

Central Objects merge. A process that blends a customer's modifications to the objects in a current release with objects in a new release.

central processing unit (CPU). Computer component which carries out the logic, calculation and decision–making functions. Interprets and executes instructions upon receipt.

character conversion. The process of converting characters of the same language from one encoding scheme to another while sending and receiving data in a heterogeneous environment.

character set. An ordered set of characters representing any particular language.

chart. OneWorld term for tables of information that appear on forms in the software. See forms.

chart of accounts. The structure for general ledger accounts. The chart of accounts lists types of accounts, describes each account, and includes account numbers and posting edit codes.

check-in location. The directory structure location for the package and its set of replicated objects. This is usually \\deploymentserver\release\path_code\package\packagename. The subdirectories under this path are where the central C components (source, include, object, library, and DLL file) for business functions are stored.

child. See parent/child form.

client. 1) A workstation or PC in a client/server environment. 2) The receiving end of the spectrum in a request/supply relationship between programs.

client workstation. The computer on which a user operates OneWorld software applications.

client/server. A relationship between processes running on separate machines. The server process is a provider of software services. The client is a consumer of those services. In essence, client/server provides a clean separation of function based on the idea of service. A server can service many clients at the same time and regulate their access to shared resources. There is a many-to-one relationship between clients and a server, respectively. Clients always initiate the dialog by requesting a service. Servers passively wait for requests from clients.

cluster. A group of two or more servers with identical configurations used to provide protection against failure. If one server fails, the other can continue processing.

code page. An ordered set of characters in which an alphanumeric value (code point) is associated with each character set.

code point. The numeric identifier assigned to a character. Its value is usually expressed in a hexadecimal notation.

coexistence. An AS/400 configuration of J.D. Edwards software that allows shared data interface operation to occur between OneWorld and WorldSoftware.

commit. A process that ensures that all database changes for a single transaction occur

simultaneously. The changes are treated as a single unit; either all changes occur or none of the changes occur, thereby maintaining the integrity of the database.

common object request broker architecture. An object request broker standard endorsed by the Object Management Group.

component object model (COM). A specification developed by Microsoft for building software components that can be assembled into programs or add functionality to existing programs running on Microsoft Windows platforms. COM components can be written in a variety of languages, although most are written in C++, and can be unplugged from a program at runtime without having to recompile the program.

Conference Room Pilot environment. A OneWorld environment used as a staging environment for production data, which includes constants and masters tables, such as company constants, fiscal date patterns, and item master. Use this environment along with the test environment to make sure your configuration works before you release changes to end–users.

configurable client engine. Allows user flexibility at the interface level. Users can easily move columns, set tabs for different data views, and size grids according to their needs. The configurable client engine also enables the incorporation of Web browsers in addition to the Windows 95- and Windows NT-based interfaces.

Configurable Network Computing (CNC). An application architecture that allows interactive and batch applications, composed of a single code base, to run across a TCP/IP network of multiple server platforms and SQL databases. The applications consist of reusable business functions and associated data that can be configured across the network dynamically. The overall objective for businesses is to provide a future-proof environment that enables them to change organizational structures, business processes, and technologies independently of each other.

constants. Parameters or codes that you set and that the system uses to standardize the processing of information by associated programs.

control. Any data entry point allowing the user to interact with an application. For example, check boxes, pull-down lists, hyper-buttons, entry fields, and similar features are controls.

Control Table Workbench. During the Installation Workbench process, Control Table Workbench runs the batch applications for the planned merges that update the data dictionary, user defined codes, menus, and user overrides tables.

Control Tables merge. A process that blends a customer's modifications to the control tables with the data that accompanies a new release.

cumulative update. A version of OneWorld software that includes fixes and enhancements made since the last release or update.

custom gridlines. A grid row that does not come from the database, for example, totals. To display a total in a grid, sum the values and insert a custom gridline to display the total. Use the system function Insert Grid Row Buffer to accomplish this.

custom installation. One of the two types of installations you can set up in the Installation Planner application. A custom installation gives the customer flexibility in creating a plan with Java and Windows terminal servers, custom environments, and custom data sources. See also typical installation.

custom modifications. Changes customers make to OneWorld to make the software run more efficiently for them or to meet their particular requirements.

customer. The company that purchases and uses OneWorld. A customer contains individual users.

data dictionary. A database table that OneWorld uses to manage the definitions, structures, and guidelines for the usage of fields, messages, and help text. J.D. Edwards has an active data dictionary, which means that it is accessed at runtime.

Data Dictionary merge. A process that updates a customer's data dictionary tables with the data that accompanies a new release.

data mart. Department-level decision support databases. They usually draw their data from an enterprise data warehouse that serves as a source of consolidated and reconciled data from

around the organization. Data marts can be either relational or multidimensional databases.

data only upgrade. A process that preserves customer business data when moving from a previous release of OneWorld to a new release. This shortens the upgrade process by eliminating the need to perform the merges and table conversions that incorporate J.D. Edwards data into a customer's existing data.

data replication. In a replicated environment, multiple copies of data are maintained on multiple machines. There must be a single source that "owns" the data. This ensures that the latest copy of data can be applied to a primary place and then replicated as appropriate. This is in contrast to a simple copying of data, where the copy is not maintained from a central location, but exists independently of the source.

data server. A machine required for AS/400 users who need to put central objects in SQL Server or Oracle. Putting central objects on a data server tells OneWorld they are not on the enterprise server.

data source. A specific instance of a database management system running on a computer. Data source management is accomplished through Object Configuration Manager (OCM) and Object Map (OM).

Data Source Workbench. During the Installation Workbench process, Data Source Workbench copies all data sources that are defined in the installation plan from the Data Source Master and Table and Data Source Sizing tables in the Planner data source to the System – release number data source. It also updates the Data Source Plan detail record to reflect completion.

data structure. A group of data items that can be used for passing information between objects, for example, between two forms, between forms and business functions, or between reports and business functions.

data warehouse. A database used for reconciling and consolidating data from multiple databases before it is distributed to data marts for department-level decision support queries and reports. The data warehouse is generally a large relational database residing on a dedicated server between operational databases and the data marts.

database. A continuously updated collection of all information that a system uses and stores. Databases make it possible to create, store, index, and cross-reference information online.

database administrator. The person who has special skills or training in one or more types of database software, for example, SQL Server or Oracle.

database driver. Software that connects an application to a specific database management system.

database management system (DBMS). A computer program that manages data by providing centralized control, data independence, and complex physical structures for efficient access, integrity, recovery, concurrence, control, privacy, and security.

database server. A server that stores data. A database server does not have OneWorld logic.

default. A code, number, or parameter that the system supplies when the user does not specify one.

default. A value that the system assigns when the user does not enter a value. For example, if the default value for an input is N and nothing is entered in that field, the system assumes the default is an N value.

deployment environment. A OneWorld environment used to run OneWorld on the deployment server. This environment administers information for the system data source, such as user profiles, packages, and environments.

deployment server. The computer used to install, maintain, and distribute OneWorld software to one or more enterprise servers and client workstations.

detail. The specific information that makes up a record or transaction. Contrast with summary.

detail area. An area of a form that displays detailed information associated with the records or data items displayed on the form. See also grid.

development environment. A OneWorld environment used to test modified development

objects before transferring them to the conference room pilot environment.

direct connect. A transaction method in which a client application communicates interactively and directly with a server application. See also batch-of-one immediate, store-and-forward.

director. A OneWorld user interface that guides a user interactively through a OneWorld process.

disk. A direct access storage device.

distributed computing environment (DCE). A set of integrated software services that allows software running on multiple computers to perform in a manner that is seamless and transparent to the end-users. DCE provides security, directory, time, remote procedure calls, and files across computers running on a network.

distributed database management system (DDBMS). A system for distributing a database and its control system across many geographically dispersed machines.

Do Not Translate (DNT). A type of data source that must exist on the AS/400 because of BLOB restrictions.

double–byte character set (DBCS). A method of representing some characters using one byte and other characters using two bytes. Double-byte character sets are necessary to represent some characters in the a Japanese, Korean, and Chinese languages.

double–byte enabled. A data storage feature that allows a computer to store ideographic characters from Asian languages. J.D. Edwards coding techniques accommodate both ideographic and alphabetic characters, making it easier to translate an application into another language.

driver. A program or portion of a program that controls the transfer of data from an input or output device.

duplicated database. A decision support database that contains a straightforward copy of operational data. The advantages involve improved performance for both operational and reporting environments. See also enhanced analysis database, enterprise data warehouse.

dynamic link library (DLL). A set of program modules that are designed to be invoked from executable files when the executable files are run, without having to be linked to the executable files. They typically contain commonly used functions.

dynamic partitioning. The ability to dynamically distribute logic or data to multiple tiers in a client/server architecture.

Electronic Data Interchange (EDI). The paperless, computer-to-computer exchange of business transactions, such as purchase orders and invoices, in a standard format with standard content.

embedded event rule. An event rule that is specific to a particular table or application. Examples include form-to-form calls, hiding a field based on a processing option value, and calling a business function. Contrast with business function event rule. See also event rule.

Employee Work Center. This is a central location for sending and receiving all OneWorld messages (system and user generated) regardless of the originating application or user. Each user has a mailbox that contains workflow and other messages, including Active Messages. With respect to workflow, the Message Center is MAPI compliant and supports drag and drop work reassignment, escalation, forward and reply, and workflow monitoring. All messages from the message center can be viewed through OneWorld messages or Microsoft Exchange.

encapsulation. The ability to confine access to and manipulation of data within an object to the procedures that contribute to the definition of that object.

end user. An individual who uses OneWorld software.

enhanced analysis database. A database containing a subset of operational data. The data on the enhanced analysis database performs calculations and provides summary data to speed generation of reports and query response times. This solution is appropriate when external data must be added to source data, or when historical data is necessary for trend analysis or regulatory reporting. See also duplicated database, enterprise data warehouse.

enterprise. Every server, PC, and database that is on an organization's network.

enterprise data warehouse. A complex solution that involves data from many areas of the enterprise. This environment requires a large relational database (the data warehouse) that is a central repository of enterprise data, which is clean, reconciled, and consolidated. From this repository, data marts retrieve data to provide department-level decisions. See also duplicated database, enhanced analysis database.

enterprise server. A database server and logic server. See database server. Also referred to as host.

Enterprise Workflow Management. A OneWorld system that provides a way of automating tasks, such as notifying a manager that a requisition is waiting for approval, using an e-mail-based process flow across a network.

environment. A path code with a set of Object Configuration Manager (OCM) mappings that allow a user to locate data and a specific set of objects. Examples include Conference Room Pilot (CRP), development, production, and pristine.

Environment Checker. An application you can run before installing or upgrading OneWorld to diagnose configuration and setup issues that may exist at the operating system level.

Environment Workbench. During the Installation Workbench process, Environment Workbench copies the environment information and Object Configuration Manager tables for each environment from the Planner data source to the System release number data source. It also updates the Environment Plan detail record to reflect completion.

Ethernet. A commonly used, shared media LAN technology, which broadcasts messages to all nodes on the network segment. Ethernet connects up to 1,024 nodes at 10MB per second over twisted pair cable, coaxial cable, and optical fiber.

event. An action that occurs when an interactive or batch application is running. Example events are tabbing out of an edit control, clicking a push button, initializing a form, or performing a page break on a report. The GUI operating system uses miniprograms to manage user activities within a form. Additional logic can be attached to these miniprograms and used to give greater functionality to any event within a OneWorld application or report using event rules.

event rule. Used to create complex business logic without the difficult syntax that comes with many programming languages. These logic statements can be attached to applications or database events and are executed when the defined event occurs, such as entering a form, selecting a menu bar option, page breaking on a report, or selecting a record. An event rule can validate data, send a message to a user, call a business function, as well as many other actions. There are two types of event rules:

1 Embedded event rules.
2 Business function event rules.

executable file.

A computer program that can be run from the computer's operating system. Equivalent terms are "application" and "program.".

facility. An entity within a business for which you want to track costs. For example, a facility might be a warehouse location, job, project, work center, or branch/plant. Sometimes referred to as a business unit.

field. 1) An area on a form that represents a particular type of information, such as name, document type, or amount. 2) A defined area within a record that contains a specific piece of information. For example, a supplier record consists of the fields Supplier Name, Address, and Telephone Number.

file. A set of information stored under one name. See also table.

file transfer protocol (FTP). A set of TCP/IP commands used to log on to a network, list directories, and copy files. FTP is also a communications protocol used to transmit files without data loss.

find/browse. A type of form used to:

1 Search, view, and select multiple records in a detail area.
2 Delete records.
3 Exit to another form.
4 Serve as an entry point for most applications.

firewall.

A set of technologies that allows an enterprise to test, filter, and route all incoming messages. Firewalls are used to keep an enterprise secure.

fix/inspect. A type of form used to view, add, or modify existing records. A fix/inspect form has no detail area.

form. The element of the OneWorld graphical user interface by which the user exchanges data with interactive applications. Forms are made up of controls, such as fields, options, and the grid. These controls allow the user to retrieve information, add and revise information, and navigate through an application to accomplish a task.

Form Design Aid (FDA). The OneWorld GUI development tool for building interactive applications and forms.

form interconnection. Allows one form to access and pass data to another form. Form interconnections can be attached to any event; however, they are normally used when a button is clicked.

form type. The following form types are available in OneWorld:

1 Find/browse.
2 Fix/inspect.
3 Header detail.
4 Headerless detail.
5 Message.
6 Parent/child.
7 Search/select.

fourth generation language (4GL).

A programming language that focuses on what you need to do and then determines how to do it. Structured Query Language is an example of a 4GL.

general release. See release.

generic text structures. See Media Storage Objects.

graphical user interface (GUI). A computer interface that is graphically based as opposed to being character-based. An example of a character-based interface is that of the AS/400. An example of a GUI is Microsoft Windows. Graphically based interfaces allow pictures and other graphic images to be used in order to give people clues on how to operate the computer.

grid. A control that displays detail information on a form. The grid is arranged into rows, which generally represent records of data, and columns, which generally represent fields of the record. See also detail area.

header. Information at the beginning of a table or form. Header information is used to identify or provide control information for the group of records that follows.

header detail. A type of form used to add, modify, or delete records from two different tables. The tables usually have a parent/child relationship.

headerless detail. A type of form used to work with multiple records in a detail area. The detail area is capable of of receiving input.

host. In the centralized computer model, a large timesharing computer system that terminals communicate with and rely on for processing. It contrasts with client/server in that those users work at computers that perform much of their own processing and access servers that provide services such as file management, security, and printer management.

hypertext markup language (HTML). A markup language used to specify the logical structure of a document rather than the physical layout. Specifying logical structure makes any HTML document platform independent. You can view an HTML document on any desktop capable of supporting a browser. HTML can include active links to other HTML documents anywhere on the Internet or on intranet sites.

index. Represents both an ordering of values and a uniqueness of values that provide efficient access to data in rows of a table. An index is made up of one or more columns in the table.

inheritance. The ability of a class to recieve all or parts of the data and procedure definitions from a parent class. Inheritance enhances developement through the reuse of classes and their related code.

install. To load a full or partial set of OneWorld software on a machine that has existing or nonexisting OneWorld software, such as install OneWorld for the 0first time, install an upgrade, or install an update.

installation. The process of putting OneWorld software on your computer for the first time. An example of an installation is B73.2. As in OneWorld Installation Guide.

Installation Planner. OneWorld program that runs on the deployment server as a system administration tool. Installation Planner guides

you through the installation setup process, including defining the enterprise server and deployment platform information, setting up environments, and defining data sources.

Installation Workbench. OneWorld system administration program that allocates and configures software and resources on servers and workstations according to the plan you created in Installation Planner.

installer. The person who can perform most tasks and processes during a OneWorld installation, upgrade, or update.

integrated toolset. Unique to OneWorld is an industrial-strength toolset embedded in the already comprehensive business applications. This toolset is the same toolset used by J.D. Edwards to build OneWorld interactive and batch applications. Much more than a development environment, however, the OneWorld integrated toolset handles reporting and other batch processes, change management, and basic data warehousing facilities.

integration. A situation in which J.D. Edwards software and the software of another company access the same database.

integrity test. A process used to supplement a company's internal balancing procedures by locating and reporting balancing problems and data inconsistencies.

interactive processing. Processing actions that occur in response to commands that you enter directly into the system. During interactive processing, you are in direct communication with the system, and it might prompt you for additional information while processing your request. Contrast with batch processing.

interactive processing. A job the computer performs in response to a command. During interactive processing, the user communicates directly with the computer, which may prompt the user to input additional information during the processing of a request.

Internet. The worldwide constellation of servers, applications, and information available to a desktop client through a phone line or other type of remote access.

Internet address. A specified path used to send and receive messages on the Internet. The parts of the address identify the contact, company, and type of business.

interoperability. The ability of different computer systems, networks, operating systems, and applications to work together and share information.

intranet. A small version of the Internet usually confined to one company or organization. An intranet uses the functionality of the Internet and places it at the disposal of a single enterprise.

IP. A connectionless communication protocol that by itself provides a datagram service. Datagrams are self-contained packets of information that are forwarded by routers based on their address and the routing table information contained in the routers. Every node on a TCP/IP network requires an address that identifies both a network and a local host or node on the network. In most cases the network administrator sets up these addresses when installing new workstations. In some cases, however, it is possible for a workstation, when booting up, to query a server for a dynamically assigned address.

IServer Service. Developed by J.D. Edwards, this Internet server service resides on the Web server, and is used to speed up delivery of the Java class files from the database to the client.

J.D. Edwards Database. See JDEBASE Database Middleware.

Java. An Internet executable language that, like C, is designed to be highly portable across platforms. This programming language was developed by Sun Microsystems. Applets, or Java applications, can be accessed from a Web browser and executed at the client, provided that the operating system or browser is Java-enabled. (Java is often described as a scaled-down C++). Java applications are platform independent.

Java application server. A server through which a user can interact with OneWorld applications using a Web browser.

Java Database Connectivity (JDBC). The standard way to access Java databases, set by Sun Microsystems. This standard allows you to use any JDBC driver database.

JDBNET. A database driver that allows heterogeneous servers to access each other's data.

jde.ini. J.D. Edwards file (or member for AS/400) that provides the runtime settings required for OneWorld initialization. Specific versions of the file/member must reside on every machine running OneWorld. This includes workstations and servers.

JDE.LOG. The main diagnostic log file of OneWorld. Always located in the root directory on the primary drive. Contains status and error messages from the startup and operation of OneWorld.

JDEBASE Database Middleware. J.D. Edwards proprietary database middleware package that provides two primary benefits:

1. Platform-independent APIs for multidatabase access. These APIs are used in two ways:

a. By the interactive and batch engines to dynamically generate platform-specific SQL, depending on the data source request.

b. As open APIs for advanced C business function writing. These APIs are then used by the engines to dynamically generate platform-specific SQL.

2. Client-to-server and server-to-server database access. To accomplish this OneWorld is integrated with a variety of third-party database drivers, such as Client Access 400 and open database connectivity (ODBC).

JDECallObject. An application programming interface used by business functions to invoke other business functions.

JDEIPC. Communications programming tools used by server code to regulate access to the same data in multiprocess environments, communicate and coordinate between processes, and create new processes.

JDENET communications middleware. J.D. Edwards proprietary communications middleware package for OneWorld. It is a peer-to-peer, message-based, socket based, multiprocess communications middleware solution. It handles client-to-server and server-to-server communications for all OneWorld supported platforms.

job. A single identifiable set of processing actions that user directs the computer to perform. Jobs are initiated by selecting menu options, entering commands, or pressing designated function keys.

job queue. A group of jobs waiting to be batch processed. See also batch processing.

just in time installation (JITI). OneWorld's method of dynamically replicating objects from the central object location to a workstation.

just in time replication (JITR). OneWorld's method of replicating data to individual workstations. OneWorld replicates new records (inserts) only at the time the user needs the data. Changes, deletes, and updates must be replicated using Pull Replication.

landscape. A printer orientation for a page with greater width than height. Contrast with portrait.

language preference code. An abbreviation that identifies the preferred language to be used for the text for online and printed output. This code is used in the user profile to designate the user's preferred language(s).

local area network (LAN). A short distance network consisting of workstations, servers, a NOS, and a communications link. It is distinguished by the absence of telecommunications service.

location. The method by which OneWorld manages the organizational entities within an enterprise. The differentiation between locations can be physical (for example, New York and Tokyo) or virtual (for example, Headquarters and Accounting). A location is identified by a three-character location code, which is set up during OneWorld installation.

Location Workbench. During the Installation Workbench process, Location Workbench copies all locations that are defined in the installation plan from the Location Master table in the Planner data source to the System data source.

log files. Files that track operations for a process or application. Reviewing log files is helpful for troubleshooting problems. The file extension for log files is .LOG.

master table. A database table used to store data and information that is permanent and necessary to the system's operation. Master tables might contain data, such as paid tax

amounts, supplier names, addresses, employee information, and job information.

media storage objects. Files that use one of the following naming conventions that are not organized into table format: Gxxx, xxxGT or GTxxx.

menu masking. A security feature that lets you prevent individual users from accessing specified menus or menu selections.

menu merge. A process that blends a customer's modifications to the menu tables with the data that accompanies a new release.

merge. A OneWorld process that takes a customer's custom modifications and blends them into the data that accompanies a new release.

Messaging Application Programming Interface (MAPI). An architecture that defines the components of a messaging system and how they behave. It also defines the interface between the messaging system and the components.

middleware. A general term that covers all the distributed software needed to support interactions between clients and servers. Think of it as the software that's in the middle of the client/server system or the "glue" that lets the client obtain a service from a server.

modal. A restrictive or limiting interaction created by a given condition of operation. Modal often describes a secondary window that restricts a user's interaction with other windows. A secondary window can be modal with respect to it's primary window or to the entire system. A modal dialog box must be closed by the user before the application continues.

modeless. Not restricting or limiting interaction. Modeless often describes a secondary window that does not restrict a user's interaction with other windows. A modeless dialog box stays on the screen and is available for use at any time but also permits other user activities.

multitier architecture. A client/server architecture that allows multiple levels of processing. A tier defines the number of computers that can be used to complete some defined task.

named event rules (NER). Also called business function event rules. Encapsulated, reusable business logic created using event rules, rather than C programming.

National Language Support (NLS). Mechanisms provided to facilitate internationalization of both system and application user interfaces.

network addresses. A unique position assigned to a node operating in a network that other nodes use when communicating with it. For Ethernet and Token Ring network adapters, unique addresses are assigned at the factory and consist of a 6-byte address. Half of this address identifies the board's manufacturer, while the last half is unique to the board and is assigned when the board is manufactured. Communication errors are prevented, because no two Ethernet or Token Ring NICs will have identical addresses.

network computer. As opposed to the personal computer, the network computer offers (in theory) lower cost of purchase and ownership and less complexity. Basically, it is a scaled-down PC (very little memory or disk space) that can be used to access network-based applications (Java applets, ActiveX controls) via a network browser.

network computing. Often referred to as the next phase of computing after client/server. While its exact definition remains obscure, it generally encompasses issues such as transparent access to computing resources, browser-style front-ends, platform independence, and other similar concepts.

next numbers. A feature used to control the automatic numbering of items such as new G/L accounts, vouchers, and addresses. Next numbers provides a method of incrementing numbers.

node. A termination point for two or more communications links. A node can serve as the control location for forwarding data among the elements of a network or multiple networks, as well as perform other networking and, in some cases, local processing.

normalized. In database management, normalization applies a body of techniques to a relational database in order to minimize the

inclusion of duplicate information. Normalization significantly simplifies query and update management, including security and integrity considerations.

numeric characters. Digits 0 through 9 that are used to represent data. Contrast with alphanumeric characters.

object. A self-sufficient entity that contains data as well as the structures and functions used to manipulate the data. For OneWorld purposes, an object is a reusable entity that is based on software specifications created by the OneWorld toolset. See also Object Librarian.

Object Configuration Manager (OCM). OneWorld's object request broker and the control center for the runtime environment. It keeps track of the runtime locations for business functions, data, and batch applications. When one of these objects is called, the Object Configuration Manager directs access to it using defaults and overrides for a given environment and user.

object embedding. When an object is embedded in another document, an association is maintained between the object and the application that created it; however, any changes made to the object are also only kept in the compound document. See also object linking.

Object Librarian. A repository of all versions, applications, and business functions reusable in building applications. It provides check-out and check-in capabilities for developers, and it controls the creation, modification, and use of OneWorld objects. The Object Librarian supports multiple environments (such as production and development) and allows objects to be easily moved from one environment to another.

Object Librarian merge. A process that blends any modifications to the Object Librarian in a previous release into the Object Librarian in a new release.

object linking. When an object is linked to another document, a reference is created with the file the object is stored in, as well as with the application that created it. When the object is modified, either from the compound document or directly through the file it is saved in, the change is reflected in that application as well as anywhere it has been linked. See also object embedding.

object linking and embedding (OLE). A way to integrate objects from diverse applications, such as graphics, charts, spreadsheets, text, or an audio clip from a sound program. See also object embedding, object linking.

object-based technology (OBT). A technology that supports some of the main principles of object-oriented technology: classes, polymorphism, inheritance, or encapsulation.

object-oriented technology (OOT). Brings software development past procedural programming into a world of reusable programming that simplifies development of applications. Object orientation is based on the following principles: classes, polymorphism, inheritance, and encapsulation.

OneWorld. A combined suite of comprehensive, mission-critical business applications and an embedded toolset for configuring those applications to unique business and technology requirements. OneWorld is built on the Configurable Network Computing technology, J.D. Edwards' own application architecture, which extends client/server functionality to new levels of configurability, adaptability, and stability.

OneWorld application. Interactive or batch processes that execute the business functionality of OneWorld. They consist of reusable business functions and associated data that are platform independent and can be dynamically configured across a TCP/IP network.

OneWorld object. A reusable piece of code that is used to build applications. Object types include tables, forms, business functions, data dictionary items, batch processes, business views, event rules, versions, data structures, and media objects. See also object.

OneWorld process. Allows OneWorld clients and servers to handle processing requests and execute transactions. A client runs one process, and servers can have multiple instances of a process. OneWorld processes can also be dedicated to specific tasks (for example, workflow messages and data replication) to ensure that critical processes don't have to wait if the server is particularly busy.

OneWorld Web development computer. A standard OneWorld Windows developer

computer with the additional components installed:

- Sun's JDK 1.1.
- JFC (0.5.1).
- Generator Package with Generator.Java and JDECOM.dll.
- R2 with interpretive and apllication controls/form.

open database connectivity (ODBC). Defines a standard interface for different technologies to process data between applications and different data sources. The ODBC interface is made up of a set of function calls, methods of connectivity, and representation of data types that define access to data sources.

Open Systems Interconnection (OSI). The OSI model was developed by the International Standards Organization (ISO) in the early 1980s. It defines protocols and standards for the interconnection of computers and network equipment.

operating system (OS). The software that runs on the hardware. For example, AIX 4.1 is a version of an operating system.

Oracle. A relational DBMS from Oracle. Runs on a broad variety of computers, which allows data to be entered and maintained on multiple hardware platforms.

output queue. See print queue.

package. OneWorld objects are installed to workstations in packages from the deployment server. A package can be compared to a bill of material or kit that indicates the necessary objects for that workstation and where on the deployment server the installation program can find them. It is a point-in-time "snap shot" of the central objects on the deployment server.

package location. The directory structure location for the package and its set of replicated objects. This is usually \\deployment server\release\path_code\package\ package name. The subdirectories under this path are where the replicated objects for the package will be placed. This is also referred to as where the package is built or stored.

Package Workbench. During the Installation Workbench process, Package Workbench transfers the package information tables from the Planner data source to the System - release number data source. It also updates the Package Plan detail record to reflect completion.

parallel release. A configuration of OneWorld software that lets multiple release or update levels run in separate environments on the same machine for testing, training, or development purposes. For release levels running in parallel, no tables or data are shared. For cumulative update levels running in parallel, system and server map data are shared.

parameter. A number, code, or character string you specify in association with a command or program. The computer uses parameters as additional input or to control the actions of the command or program.

parent/child form. A type of form that presents parent/child relationships in an application on one form. The left portion of the form presents a tree view that displays a visual representation of a parent/child relationship. The right portion of the form displays a detail area in browse mode. The detail area displays the records for the child item in the tree. The parent/child form supports drag and drop functionality.

partitioning. A technique for distributing data to local and remote sites to place data closer to the users who access. Portions of data can be copied to different database management systems.

path code. A pointer to a specific set of objects. A path code is used to locate:

1. Central objects.
2. Replicated objects.

plan. Refers to an installation plan. A plan is the standard means for installing, upgrading, or updating a OneWorld configuration. Plans, which are used in various phases of installation, contain information about data sources you will use, environments you will install, and packages.

planner environment. A OneWorld environment in which you prepare the main components of a OneWorld configuration.

platform. The hardware, operating system, and database on which your software is operating, for example, an HP 9000 processor using HP-UX as the operating system and Oracle as the database.

platform independence. A benefit of open systems and Configurable Network Computing. Applications that are composed of a single code base can be run across a TCP/IP network consisting of various server platforms and SQL databases.

polymorphism. A principle of object-oriented technology in which a single mnemonic name can be used to perform similar operations on software objects of different types.

port number. A numeric code that identifies a unique process for which a service can be provided on a machine.

portability. Allows the same application to run on different operating systems and hardware platforms.

portrait. The default printer orientation for a page with greater height than width. Contrast with landscape.

primary key. A column or combination of columns that uniquely identifies each row in a table.

print queue. A list of tables, such as reports, that you have submitted to be written to an output device, such as a printer. The computer spools the tables until it writes them. After the computer writes the table, the system removes the table identifier from the print queue.

pristine environment. A OneWorld environment used to test unaltered objects with J.D. Edwards demonstration data or for training classes. You must have this environment so you can compare pristine objects that you modify.

process. A complete unit of work with a defined start and end, which a computer performs. Some operating systems, such as Windows NT, HP-UX, and AIX, track processes by assigning identifiers to them. In Windows NT, a process is a running instance of an executable file.

processing option. A feature that allows you to direct the functions of a program. For example, processing options allow you to specify defaults for certain forms, control the format in which information prints on reports, and change how information appears on a form or in a report.

production environment. A OneWorld environment in which users operate OneWorld software.

protocol. A set of formalized rules specifying how hardware and software on a network should interact when transmitting and receiving information.

published table. Also called a “Master” table, this is the central copy to be replicated to other machines. Resides on the “publisher” machine. the Data Replication Publisher Table (F98DRPUB) identifies all of the published tables and their associated publishers in the enterprise.

publisher. The server that is responsible for the published table. The Data Replication Publisher Table (F98DRPUB) identifies all of the published tables and their associated publishers in the enterprise.

pull replication. One of the OneWorld methods for replicating data to individual workstations. Such machines are set up as pull subscribers using OneWorld’s data replication tools. The only time pull subscribers are notified of changes, updates, and deletions is when they request such information. The request is in the form of a message that is sent, usually at startup, from the pull subscriber to the server machine that stores the Data Replication Pending Change Notification table (F98DRPCN).

purge. The process of removing records or data from a system table.

push. Technology used to force information from a centralized server to another server or client.

push installation. A process that allows a system administrator to schedule the automatic installation of OneWorld on workstations.

push replication. A server-to-server method of data replication that notifies subscriber machines when a change is made to the publisher table. If the subscriber machine is not running when the notification is sent, the subscriber receives the message at startup.

query by example (QBE). Located at the top of a detail area, it is used to search for data to be displayed in the detail area.

queue. A stored arrangement of computer data or program waiting to be processed in the order in which they were submitted. A queue may

refer to a print queue, job queue, or message queue.

record. A collection of related, consecutive fields of data that the system treats as a single unit of information.

redundancy. Storing exact copies of data in multiple databases.

referential integrity. Ensures that a parent record cannot be deleted from the database when a child record for exists.

refresh. To modify OneWorld software, or subset of it, such as a table or business data, so that it functions at a new release or cumulative update level, such as B73.2 or B73.2.1.

regenerable. Source code for OneWorld business functions can be regenerated from specifications (business function names). Regeneration occurs whenever an application is recompiled, either for a new platform or when new functionality is added.

relationship. Links tables together and facilitates joining business views for use in an application or report. Relationships are created based on indexes.

release. A release of OneWorld regardless of any updates that might be applied. For example, the term Release B73.2 refers generically to B73.2, B73.2.1, and B73.2.2. Sometimes referred to as a general release. See also base release, release level.

release level. A specific level of OneWorld software. A release level is achieved by installing a base release and applying one or more updates. A release level also can be installed directly. See also base release, release.

release/release update. A "release" contains major new functionality, and a "release update" contains an accumulation of fixes and performance enhancements, but no new functionality.

replicated object. A copy or replicated set of the central objects must reside on each client and server that run OneWorld. The path code indicates the directory where these objects are located.

replication. A copy of an object, usually a table in a relational database, which is placed in another location. As part of replication, the object may undergo a transformation from one type of table, such as an Oracle table, to another, such as a TAM file on a client machine.

Report Design Aid (RDA). The OneWorld GUI tool for operating, modifying and copying report batch applications.

retrofitting. The process of integrating a customer's modifications into a new release of OneWorld.

rollback. A process which changes data back to a previous state after it has been committed to a database.

runtime objects. Packages of objects that are deployed to any machine that will run OneWorld.

scalability. Allows software, architecture, network, or hardware growth that will support software as it grows in size or resource requirements. The ability to reach higher levels of performance by adding microprocessors.

scripts. A collection of SQL statements that perform a specific task.

search/select. A type of form used to search for a value and return it to the calling field.

security server. A dispatched kernel process running on a server for security validation. A security server protects computer resources using security applications and redundant functionality.

server. Provides the essential functions for furnishings services to network users (or clients) and provides management functions for network administrators. Some of these functions are storage of user programs and data and management functions for the file systems. It may not be possible for one server to support all users with the required services. Some examples of dedicated servers that handle specific tasks are backup and archive servers, application and database servers.

Server Administration Workbench. A OneWorld application that provides the server administrator with vital statistics about the internal functions of OneWorld.

Server Workbench. During the Installation Workbench process, Server Workbench copies the server configuration files from the Planner data source to the System release number data

source. It also updates the Server Plan detail record to reflect completion.

service. A type of Microsoft Windows NT process that does not require anyone to be logged on to the operating system. Examples are jdesnet.exe and jdesque.exe.

servlet. Servlets provide a Java-based solution used to address the problems currently associated with doing server-side programming, including inextensible scripting solutions. Servlets are objects that conform to a specific interface that can be plugged into a Java-based server. Servlets are to the server what applets are to the client.

single–byte character set (SBCS). An encoding scheme in which each alphabetic character is represented by one byte. Most Western languages, such as English can be represented using a single-byte character set.

socket. A communications end point through which an application sends or receives packets of data across a network. Also known as Berkley Socket, developed by the University of California at Berkley.

software action request (SAR). An entry in the AS/400 database used for requesting modifications to J.D. Edwards software.

Specification merge. The Specification merge is comprised of three merges: Object Librarian merge, Versions List merge, and Central Objects merge. The merges blend customer modifications with data that accompanies a new release.

Specification Table Merge Workbench. During the Installation Workbench process, Specification Table Merge Workbench runs the batch applications that update the specification tables.

specifications. A complete description of a OneWorld object. Each object has its own specification, or name, which is used to build applications.

specifications. A description of a OneWorld object, such as a table's width and depth, placement of fields, and fonts used.

spool. The function by which the system stores generated output to await processing.

static text. Short, descriptive text that appears next to a control variable or field. When the variable or field is enabled, the static text is black; when the variable or field is disabled, the static text is gray.

store-and-forward. A transaction method that allows a client application to perform work and, at a later time, complete that work by connecting to a server application. This often involves uploading data residing on a client to a server. See also batch-of-one immediate, direct connect.

structured query language (SQL). A fourth generation language used as an industry standard for relational database access. It can be used to create databases and to retrieve, add, modify, or deleta data from databases. SQL is not a complete programming language because it does not contain control flow logic.

subscriber. The server that is responsible for the replicated copy of a published table. Such servers are identified in the Subscriber Table.

subscriber table. The Subscriber table (F98DRSUB), which is stored on the Publisher Server with the Data Replication Publisher table (F98DRPUB) identifies all of the subscriber machines for each published table.

summary. The presentation of data or information in a cumulative or totaled manner in which most of the details have been removed. Many systems offer forms and reports that summarize information stored in certain tables. Contrast with detail.

synchronous. A method of running processes in which one process must finish before the next one can begin.

system. A group of related applications identified by a name and a system code. For example, the Address Book system code is 01. All applications, tables, and menus within a system can be identified by the system code.

system administrator. The person who has access to perform tasks such as issue signon names or maintain security.

system code. A code that identifies a system, for example, 01 for the Address Book system and 31 for the Shop Floor Management system.

system function. A program module, provided by OneWorld, available to applications and reports for further processing.

table. In database environments, a two-dimensional entity made up of rows and columns. All physical data in a database are stored in tables. See also file.

table. A two-dimensional entity made up of rows and columns. All physical data in a database are stored in tables. A row in a table contains a record of related information. An example would be a record in an Employee table containing the Name, Address, Phone Number, Age, and Salary of an employee. Name is an example of a column in the employee table.

table. A file in OneWorld software.

table access management (TAM). The OneWorld component that handles the storage and retrieval of user defined data. TAM stores information such as data dictionary definitions; application and report specifications; event rules; table definitions; business function input parameters and library information; and data structure definitions for running applications, reports, and business functions.

table conversion. During an upgrade or update, this process changes OneWorld technical and application tables to the format for the new release.

Table Conversion Workbench. During the Installation Workbench process, Table Conversion Workbench runs the table conversions that change the technical and application tables to the format for the new release of OneWorld. It also updates the Table Conversions and Controls detail records to reflect completion.

Table Design Aid (TDA). A OneWorld GUI tool for creating, modifying, copying, and printing database tables.

table event rules. Use table event rules to attach database triggers (or programs) that automatically run whenever an action occurs against the table. An action against a table is referred to as an event. When you create a OneWorld database trigger, you must first determine which event will activate the trigger. Then, use Event Rules Design to create the trigger. Although OneWorld allows event rules to be attached to application events, this functionality is application specific. Table event rules provide embedded logic at the table level.

TCP/IP. Transmission Control Protocol/Internet Protocol. The original TCP protocol was developed as a way to interconnect networks using many different types of transmission methods. TCP provides a way to establish a connection between end systems for the reliable delivery of messages and data.

TCP/IP services port. Used by a particular server application to provide whatever service the server is designed to provide. The port number must be readily known so that an application programmer can request it by name.

technical data. A type of OneWorld data source that contains information about how OneWorld operates.

technical tables. Tables used for technical processes such as installation and upgrade of OneWorld, in contrast with tables used by applications.

Telnet. A terminal emulation protocol frequently used on the Internet that allows a user to log on and run a program from a remote computer. Telnet is part of the TCP/IP communications.

test environment. A OneWorld environment used along with the Conference Room Pilot environment to test OneWorld software or the modifications made in the development path code before you release changes to the end user.

third generation language (3GL). A programming language that requires detailed information about how to complete a task. Examples of 3GLs are COBOL, C, Pascal and FORTRAN.

third–party. Describes other software that is used in conjunction with J.D. Edwards software.

token. A bit configuration circulated among workstations, which lets workstation send data to the network.

token ring. A LAN access mechanism in which all stations attached to a bus wait for a broadcast token to be passed to them before they are able to transmit. However, though token-passing technology is in a physical ring, the next

receiving station might not be the next physical station.

TP monitor. Transaction Processing monitor. A monitor that controls data transfer between local and remote terminals and the applications that originated them. TP monitors also protect data integrity in the distributed environment and may include programs that validate data and format terminal screens.

trace. A process that helps the user troubleshoot problems.

trigger. Allow you to attach default processing to a data item in the data dictionary. When that data item is used on an application or report, the trigger is invoked by an event associated with the data item. OneWorld also has three visual assist triggers: calculator, calendar and search form.

typical installation. One of the two types of installations you can set up in the Installation Planner application. A typical installation is the quickest way to create an installation plan, because it uses all of the J.D. Edwards default information for environment and data sources. See also custom installation.

uniform resource locator (URL). Names the address of a document on the Internet or an intranet. The following is an example of URL:http://www.jdedwards.com. This is J.D. Edwards Internet address.

unnormalized. Data that is a random collection of data elements with repeating record groups scattered throughout. Also see Normalized.

update. The process of refreshing OneWorld software to a new release level, such as from B73.2 to B73.3.

upgrade. The process of refreshing OneWorld software to a new release level, such as from B73.2 to B73.3.

user. An individual who uses OneWorld software.

user defined code (UDC). A code that users can define, assign code descriptions, and assign valid values. Examples of such codes are unit-of-measure codes, state names, and employee type codes.

user defined code type. The identifier for a table of codes with a meaning that you define for the system, such as ST for the Search Type codes table in Address Book. OneWorld provides a number of these tables and allows you to create and define tables of your own.

User Defined Codes merge. The User Defined Codes merge blends a customer's modifications to the user defined code tables with the data that accompanies a new release.

user display preferences. A set of values that represents a user's preferred language, date format, decimal format, and other country specific conventions.

User Overrides merge. The User Overrides merge adds new user override records into a customer's user override table.

user profile. The predefined characteristics required for each user. The user profile includes a library list, default print queue, and default job queue, as well as several other characteristics.

Versions List merge. The Versions List merge preserves any non-XJDE and non-ZJDE version specifications for objects that are valid in the new release as well as their processing options data.

visual assist. Forms that can be invoked from a control to assist the user in determining what data belongs in the control.

vocabulary overrides. A feature that you can use to override field, row, or column title text on forms and reports.

wchar_t. Internal type of a wide character. Used for writing portable programs for international markets.

Web client. Any workstation that contains an internet browser. The Web client communicates with the web server for OneWorld data.

Web server. Any workstation that contains the IServer service, SQL server, Java menus and applications, and Internet middleware. The Web server receives data from the web client, and passes the request to the enterprise server. When the enterprise server processes the information, it sends it back to the Web server, and the Web server sends it back to the Web client.

wide area network (WAN). A network that extends beyond an area served by the dedicated communication lines of a LAN and is capable of

covering long distance. It is distinguished by the requirement that a phone company or telecommunications provider be part of the transmission.

workflow. According to the Workflow Management Coalition, worlflow means “the automation of a business process, in whole or part, during which documents, information, or tasks are passed from one participant to another for action, according to a set of procedural rules.”.

workgroup server. A remote database server usually containing subsets of data replicated from a master database server. This server does not performance an application or batch processing. It may or may not have OneWorld running (in order to replicate data).

WorldSoftware Architecture. The broad spectrum of application design and programming technology that J.D. Edwards uses to achieve uniformity, consistency, and complete integration throughout its software.

Worldwide Web. A part of the Internet that can transmit text, graphics, audio, and video. The Worldwide Web allows clients to launch local or remote applications.

z file. For store and forward (network disconnected) user, OneWorld store-and-forward applications perform edits on static data and other critical information that must be valid to process an order. After the initial edits are complete, OneWorld stores the transactions in work tables on the workstation. These work table are called Z files. When a network connection is established, Z files are uploaded to the enterprise server and the transactions are edited again by a master business function. The master business function will then update the records in your transaction files.

Index

A

B

C

D

E

F

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

Z