

PeopleSoft®

---

EnterpriseOne Xe  
Product Packaging  
PeopleBook

---

**September 2000**

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

Portions of this document were reproduced from material prepared by J.D. Edwards.

Copyright ©J.D. Edwards World Source Company, 2000

All Rights Reserved

SKU XeEAPT

J.D. Edwards is a registered trademark of J.D. Edwards & Company. The names of all other products and services of J.D. Edwards used herein are trademarks or registered trademarks of J.D. Edwards World Source Company.

All other product names used are trademarks or registered trademarks of their respective owners.

The information in this guide is confidential and a proprietary trade secret of J.D. Edwards World Source Company. It may not be copied, distributed, or disclosed without prior written permission. This guide is subject to change without notice and does not represent a commitment on the part of J.D. Edwards & Company and/or its subsidiaries. The software described in this guide is furnished under a license agreement and may be used or copied only in accordance with the terms of the agreement. J.D. Edwards World Source Company uses automatic software disabling routines to monitor the license agreement. For more details about these routines, please refer to the technical product documentation.

# Table of Contents

---

<b>Welcome</b> .....	<b>1-1</b>
Welcome .....	1-1
Product Packaging Definitions .....	1-1
Prerequisites .....	1-3
J.D. Edwards Worldwide Customer Support .....	1-4
Conventions .....	1-6
<b>Software Master Creation</b> .....	<b>2-1</b>
Software Master Creation .....	2-1
Checklist for Creating a Software Master .....	2-3
Checklist for Creating a Software Master .....	2-3
Prerequisites to Creating a Software Master .....	2-5
Prerequisites to Creating a Software Master .....	2-5
Adding ODBC Data Sources .....	2-5
Creating a Data Dictionary in the Pristine Data Source .....	2-7
Creating a Software Master .....	2-10
Creating a Software Master .....	2-10
Defining a Software Master .....	2-10
Configuring Change Tables .....	2-17
Validating a Software Master Definition .....	2-24
Creating a Final Software Master .....	2-25
Using a Software Master to Update OneWorld .....	2-27
<b>Software Master</b> .....	<b>3-1</b>
Software Master .....	3-1
Working with a Software Master .....	3-3
Working with a Software Master .....	3-3
Adding a Repair to a Software Master .....	3-3
Deleting a Repair to a Software Master .....	3-10
Deleting a Software Master .....	3-11
Working with a Software Master Definition .....	3-13
Working with a Software Master Definition .....	3-13
Revising a Software Master Definition .....	3-13
Copying a Software Master Definition .....	3-23
Proofing a Software Master Definition .....	3-25
Validating a Software Master Definition .....	3-26
Deleting a Software Master Definition .....	3-27
<b>Software Master Templates</b> .....	<b>4-1</b>
Software Master Templates .....	4-1
Understanding Software Master Templates .....	4-3
Understanding Software Master Templates .....	4-3
Details of all Build Step Processes .....	4-3

Undefined	4-3
Create CD File Structure	4-4
Package Build	4-4
Manual Build Step	4-4
Pre-Build Package	4-4
Pre-Build Pristine	4-4
Build Cross-Reference	4-5
Build Data Dictionary	4-5
Build In-House Package Database	4-5
Build Planner Database	4-5
Build Pristine Database	4-5
Build Store & Forward Database	4-6
Build Standalone Database	4-6
Build Change Tables	4-6
Master Central Objects	4-6
Compress	4-6
Create INF	4-7
Copy CD Directory Files	4-7
Virus Check	4-7
Windows Executable	4-7
Data Cleanup	4-8
Build Package Database	4-8
Batch Process	4-8
ASU CD Template	4-8
Additional Build Step Information for the ASU CD Template	4-9
Custom CD	4-11
Working with Software Master Templates	4-12
Working with Software Master Templates	4-12
Adding a Software Master Template	4-12
Revising a Software Master Template	4-13
Copying a Software Master Template	4-18
Proofing a Software Master Template	4-19
Deleting a Software Master Template	4-20
<b>Change Table Configuration</b>	<b>5-1</b>
Change Table Configuration	5-1
Revising or Submitting Change Tables	5-3
Revising or Submitting Change Tables	5-3
Revising or Submitting Change Tables	5-3
Copying a Change Table Configuration	5-7
Copying a Change Table Configuration	5-7
Copying a Change Table Configuration	5-7
Deleting a Change Table Configuration	5-9
Deleting a Change Table Configuration	5-9
<b>Data Cleanup</b>	<b>6-1</b>
Data Cleanup	6-1
Working with Data Cleanup Procedures	6-3
Working with Data Cleanup Procedures	6-3
Adding a Data Cleanup Procedure	6-3
Revising a Data Cleanup Procedure	6-9

Copying a Data Cleanup Procedure .....	6-11
Executing a Data Cleanup Procedure and Build Step .....	6-12
Deleting a Data Cleanup Procedure .....	6-13
Working with Data Cleanup Templates .....	6-15
Working with Data Cleanup Templates .....	6-15
Adding a Data Cleanup Template .....	6-15
Revising a Data Cleanup Template .....	6-18
Copying a Data Cleanup Template .....	6-21
Deleting a Data Cleanup Template .....	6-21
<b>Appendix A - Relative and Absolute Paths .....</b>	<b>B-1</b>
Appendix A - Relative and Absolute Paths .....	B-1
Using Relative Path Tokens .....	B-1
Converting Relative and Absolute Paths .....	B-2

**Glossary**

**Index**



Welcome







## Welcome

This guide explains how to use the Product Packaging Tools to create a software master image of OneWorld objects. This section provides definitions of product packaging, a process overview, and prerequisites that are required to use the tools. The rest of the guide describes how to use the tools to create your own software master that can be installed onto any enterprise that contains a OneWorld installation at the same release level.

For the complete list of tasks on how to create a software master from start to finish, see the “Software Master Creation” section of this guide. The rest of the sections go into further detail about the different aspects of software masters and the Product Packaging Tools, such as the software master definition, change table configurations, and the final software master.

This chapter contains the following:

- Product packaging definitions
- Prerequisites
- J.D. Edwards Worldwide Customer Support
- Conventions

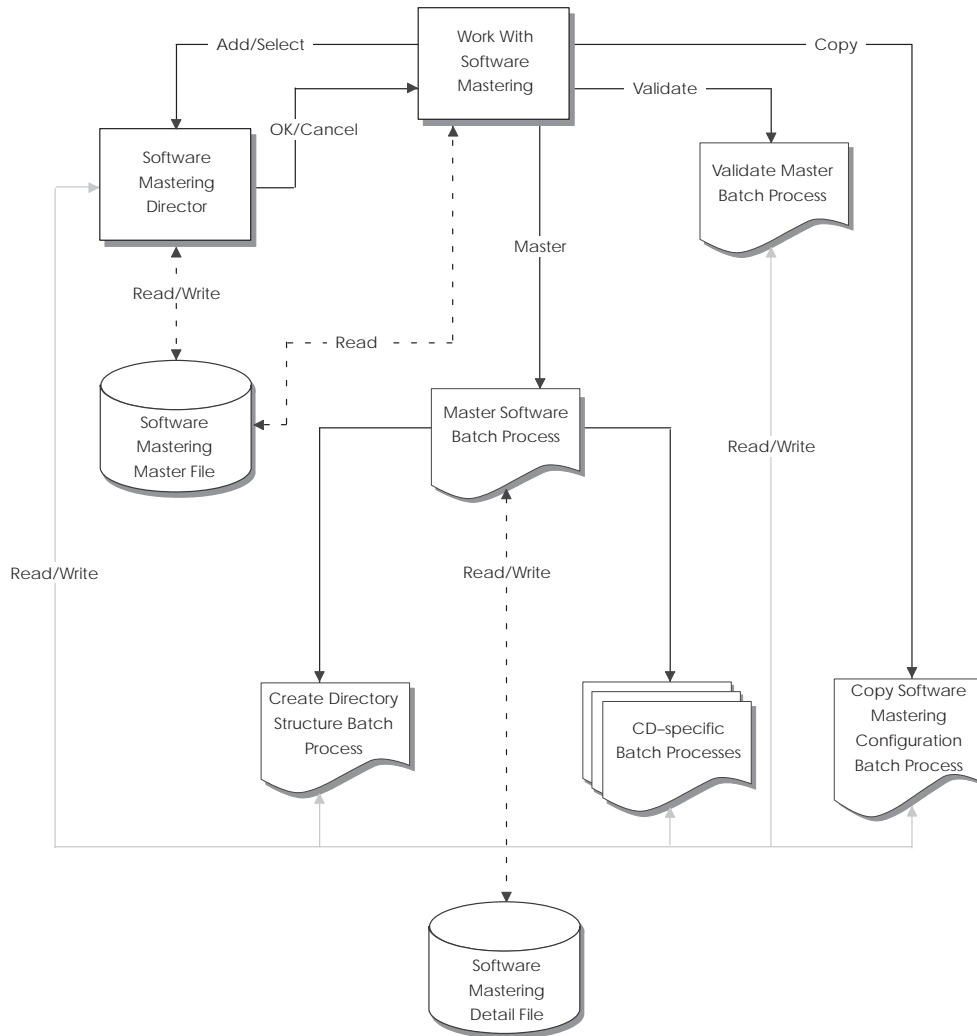
## Product Packaging Definitions

Product packaging is the ability to master and package a group of OneWorld objects. J.D. Edwards business partners can use the Product Packaging Tools to develop and market software modules that are based in OneWorld. In addition, customers who modify OneWorld can use the Product Packaging Tools to deliver changes to other locations within their company that are not directly connected to the central enterprise.

The Product Packaging Tools manage the task of mastering OneWorld software. In this capacity, the tools oversee the entire mastering process from defining a software master to mastering the CD. The tools start with the definition process, where the user provides as much information as possible up front. After the user enters the critical information, the tools provide a batch process to validate the software mastering information before mastering. Next, the tools provide a batch process to master the software. Finally, the tools provide a batch process to validate the final software master image.



The following graphic details the product packaging flow:



To better understand the overall process, you should be familiar with the following key terms and processes:

**Software Mastering Director**

This director provides OneWorld administrators with the ability to build a mastered version of OneWorld objects or modules. The director handles this through a series of batch processes and manual tasks that build all of the components of a OneWorld master. The final product is an image of their product, using a tree format, that they can burn onto a CD.

**Change Table Configuration Director**

This director provides OneWorld administrators with the ability to define how to build the software master's change tables. See the Glossary for a definition of Change Tables.

---

## **Distributed Development**

Distributed development refers to the ability to perform development using one installation of OneWorld and deploying that development to another installation of OneWorld. For example, a developer in the Paris office could create new objects using the Paris installation of OneWorld. A OneWorld administrator in Paris could then use the Product Packaging Tools to create a CD that includes these new objects. The CD could then be shipped to the London office, where an administrator updates the London installation of OneWorld, assuming both installations are at the same OneWorld release level. The Product Packaging Tools described in this guide provide this functionality.

This ability should not be confused with “remote development.” Remote development is the creation of OneWorld objects by a developer who is completely disconnected from any OneWorld installation. The developer could then connect to a OneWorld installation at a later time and merge those objects. Remote development will be available in a later release of OneWorld.

## **Prerequisites**

This guide is designed for Management Information Systems (MIS) managers and administrators of OneWorld. Success depends upon your understanding of the concepts and procedures. The recommended method for obtaining this information is to attend the relevant training courses. Information about course offerings, dates, and locations is available at the J.D. Edwards Web site. At a minimum, read the following guides before you begin:

- *OneWorld Foundation Guide*
- Configuration Planning and Setup Suite:
  - *Configurable Network Computing Implementation Guide*
  - *System Administration Guide*
  - *Package Management Guide*
  - *Server and Workstation Administration Guide*

## J.D. Edwards Worldwide Customer Support

**Important:** J.D. Edwards compiles important information that affects this guide and the update process at the following Internet addresses:

For J.D. Edwards employees:

<https://knowledge.jdedwards.com/jdecontent/customerknowledge/knowledge/breakingnews.htm>

For business partners:

<https://businesspartner2.jdedwards.com/OneSource/SharedSrc/knowledge/breakingnews.htm>

To navigate to this site within the Knowledge Garden, use the browse tab. Drill down through the menu structure: Product Support, SARs/Knowledge/Calls, to Breaking News. On Breaking News, click the OneWorld menu and choose the appropriate release of OneWorld from the list given.

For business partners and J.D. Edwards employees only, the minimum technical requirements for running OneWorld can be accessed at:

<https://knowledge.jdedwards.com/jdecontent/technicalmarketing/tminformation/bjtechreqs.htm>

To navigate to this site within the Knowledge Garden, use the browse tab. Drill down through the menu structure: Sales and Marketing, OneWorld Online. On OneWorld Online, there is a link to Minimum Technical Requirements.

Review this information before you begin to update a OneWorld release.

If you cannot resolve problems when you install OneWorld, contact J.D. Edwards Worldwide Customer Support for assistance. Limit your questions to those regarding J.D. Edwards software. For questions about operating systems, databases, and other software products, contact the appropriate vendor.

The J.D. Edwards Web site is available for support issues submitted to:  
<http://www.jdedwards.com>.

During normal business hours, you will receive a confirmation within an hour of submitting issues via the Web, e-mail, or fax. An issue number is assigned immediately when you report an issue by telephone, and this number references all activity related to your issue.

Premier maintenance customers can submit support issues via e-mail address:  
[premier\\_customer\\_support@jdedwards.com](mailto:premier_customer_support@jdedwards.com). Outside normal business hours

(Monday-Friday, 8:00 A.M.-5:00 P.M. MST), premier customers should use the telephone for the most prompt after-hours service.

Standard maintenance customers should send support issues via fax or e-mail to the appropriate service provider location listed below:

<b>Location</b>	<b>Telephone Number</b>	<b>Fax Number; E-mail Address</b>
North America	1-800-289-2999	303-334-4141; denver_customer_support@jdedwards.com
Latin America (except below)	00-1-303-334-4000	303-334-4141; denver_customer_support@jdedwards.com
Mexico	95-800-784-4260	303-334-4141; denver_customer_support@jdedwards.com
Chile	1-230-020-5124	303-334-4141; denver_customer_support@jdedwards.com
Colombia	980-153-560	303-334-4141; denver_customer_support@jdedwards.com
Venezuela	800-1-2743	303-334-4141; denver_customer_support@jdedwards.com
Argentina	001-800-333-1111, then 303-488-4639-9194, then 1-800-784-4260	303-334-4141; denver_customer_support@jdedwards.com
Ecuador	999-170, then 303-488-4639-9194, then 1-800-784-4260	303-334-4141; denver_customer_support@jdedwards.com
Peru	170 (199 for cellular phone), then 303-488-4639-9194, then 1-800-784-4260	303-334-4141; denver_customer_support@jdedwards.com
Brazil	00081-4-550-2779	303-334-4141; denver_customer_support@jdedwards.com
Europe EMEA (except below)	44-1494-682-682	44-1494-682-698; london_customer_support@jdedwards.com
France	33-1-44-74-20-15	44-1494-682-698; london_customer_support@jdedwards.com
Germany	49-6103-762-110	44-1494-682-698; london_customer_support@jdedwards.com
Holland	31-35-548-0222	44-1494-682-698; london_customer_support@jdedwards.com

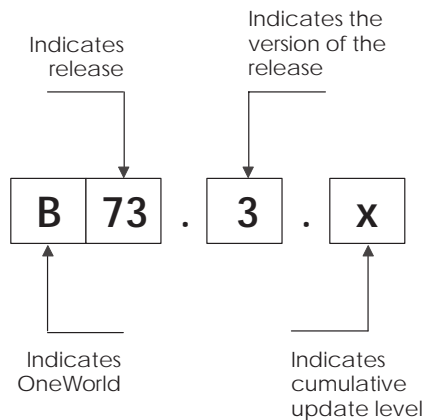
Location	Telephone Number	Fax Number; E-mail Address
Italy	39-02-2696-7612	44-1494-682-698; london_customer_support@ jdedwards.com
Asia Pacific (except below)	65-895-9656	65-227-2698; singapore_customer_support@ jdedwards.com
China (Beijing, Shanghai)	10-800-650-0099	65-227-2698; singapore_customer_support@ jdedwards.com
China (other provinces)	108-657, or 108-650, then follow voice prompts to dial Singapore at 895-9656	65-227-2698; singapore_customer_support@ jdedwards.com
Hong Kong	800-90-3055	
Malaysia	1-800-80-1557	
Japan	81-3-3265-7143	81-3-3265-7145; japan_customer_support@jdedwards.com
Australia	1-800-145-200	

## Conventions

The following explain the typographic and design conventions used in this guide.

### Version Numbering

The following graphic explains how J.D. Edwards numbers its releases:



---

## Fonts

The following special fonts are used in this guide:

*Italic font* designates variables used in the guide. For example, if you see the variable *deploymentserver* in a command you must enter, substitute the phrase *deploymentserver* for the actual name of your deployment server. Also, the names of other J.D. Edwards OneWorld guides are in italic font. For example, the *OneWorld Installation Guide*.

`Courier font` designates commands or other information that you must type into the system. For example:

Type the name of the environment, such as `DEVTEMP`.





# Software Master Creation





## Software Master Creation

This section explains how to create a software master from start to finish. This includes defining the master, selecting the product package, validating the software master definition, mastering the software, and validating the finished master. Use the checklist to ensure that you complete all of the processes necessary to create a software master.

This section contains the following:

- Checklist for creating a software master
- Prerequisites to creating a software master
- Creating a software master





## Checklist for Creating a Software Master

---

Use the following checklist to ensure that you complete all of the processes necessary to creating a software master, as explained in this section.

- Ensure that at least 1GB of space is available for the TEMP directory of the workstation from which you use the Product Packaging Tools. This memory requirement is the most memory you might need for a software master. The actual size of the master depends upon the size of the package you create.
- Complete the tasks included in the “Prerequisites to Creating a Software Master” chapter in this section.
- Define a software master using the CD Configuration Director:
  - Provide information about the type of master you want to create, such as its name, release number, etc.
  - Select the package you want built into the master.
- Configure change tables using the Change Table Director:
  - Provide the target release number for the change tables.
  - Provide source environment information about the change tables.
  - Select the change tables and their versions.
- Validate the software master definition.
- Create the final software master, which automatically validates the master.
- Use the software master to update OneWorld



## Prerequisites to Creating a Software Master

---

This chapter explains how to perform tasks that enable the Product Packaging Tools. These are one-time processes. After completing them for the first time you will not need to perform them again.

This topic contains the following:

- Adding ODBC data sources
- Creating a data dictionary in the Pristine data source

### Adding ODBC Data Sources

You need to add ODBC data sources.

For complete information about adding ODBC data sources see the “Creating Third-party Data Sources for SQL Server” chapter in the appendix of the *OneWorld Installation Guide*. This is a one-time process. After adding ODBC data sources, you will not need to add them again.

#### **To add ODBC data sources**

---

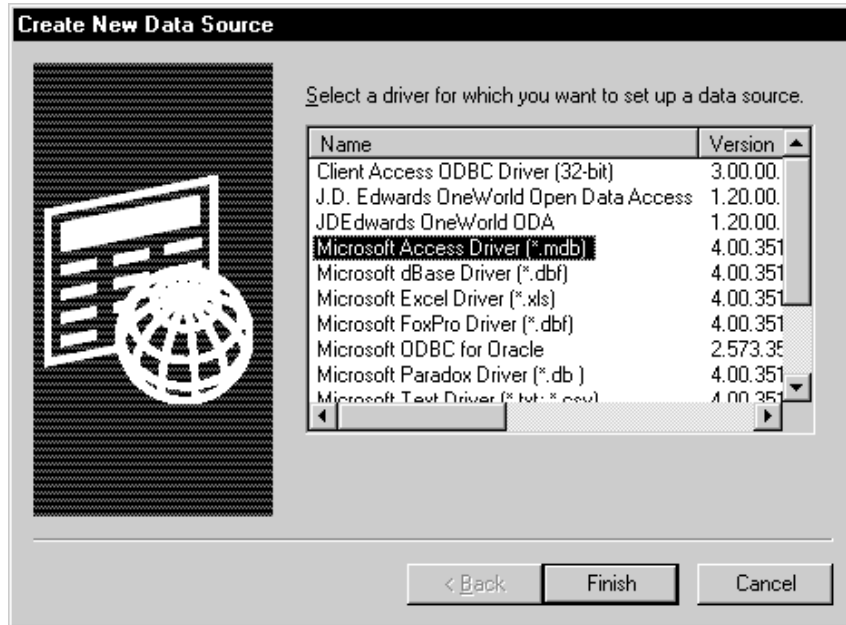
From the workstation where you will run the Product Packaging Tools

1. Create an empty jdeb7.mdb Microsoft Access database. Create the empty database in the \temp\master directory.
2. Access the workstation’s Control Panel and double-click 32-bit ODBC.

The ODBC Data Source Administrator window appears.

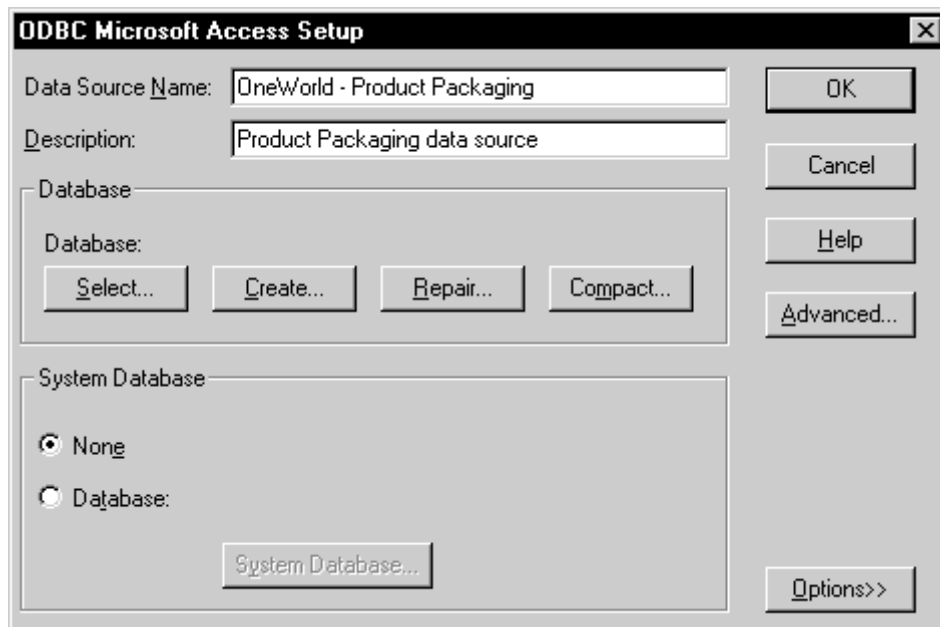
3. Click Add.

The Create New Data Source window appears.



4. Choose Microsoft Access Driver (\*.mdb), and click Finish.

The ODBC Microsoft Access Setup window appears.



5. Enter the following data source into the Data Source Name field:

OneWorld - Product Packaging

6. Enter a description into the Description field.
7. In the Database group box, click Select.



The Select Database window appears.

8. Select the empty jdeb7.mdb database that you created on the client workstation in the \temp\master directory.
9. Click OK.

The ODBC Microsoft Access Setup window reappears.

10. Click OK.

The ODBC Data Source Administrator window reappears.

OneWorld should automatically create the data source for the pristine copy of the data dictionary.

11. Verify that OneWorld created the following pristine data dictionary, and that the data source is similar to the Data Dictionary - B7333 ODBC data source. There should be two differences between the data sources:
  - The data source name should be: `Data Dictionary - JDE`
  - The owner should be: `PRISTCTL`

You need the pristine data dictionary data source if your data dictionary is stored in either SQL Server or DB2/400. An Oracle database does not require an ODBC data source.

12. When finished adding the data sources, click OK from the ODBC Data Source Administrator window.

## Creating a Data Dictionary in the Pristine Data Source

You need to create a data dictionary in your OneWorld Pristine data source. This is a one-time process. After creating these tables, you will not need to create them again.

### **To create a data dictionary in the Pristine data source**

---

From the deployment server

1. Sign onto the DEPB7333 environment. Use JDE as your user name and JDE as your password.
2. From the System Administration Tools (GH9011) menu, choose Batch Versions.
3. On the Work With Batch Versions - Available Versions form, type R98403 into the Batch Application field and click Find.

4. Choose version XJDE0509 and click Copy.
5. On Version Copy, enter a new version name and a new version title, then click OK.
6. On Work With Batch Versions - Available Versions, choose the version you just created, and from the Row menu, choose Processing Options.
7. Complete the following processing options:

- “2. or Enter the name of the target Data Source for the database to be created for.”

Type DATA DICTIONARY - PRISTB733

- “4. Enter the name of the source Data Source for Loading the Data.”

Type OneWorld Local

This data source exists only the deployment server.

These processing options set up the batch process to copy the data dictionary tables from planner/jdeb7.mdb to the new pristine data dictionary.

8. Run the version you just created in proof mode first, as explained in the “Submitting a Report” section of the *Enterprise Report Writing Guide*.
9. Verify the results of this report.
10. When satisfied with the proof-mode results, run the version you just created in final mode.
11. Run OCM from the fastpath and select the Enterprise server machine.
12. For the JD733 environment, add OCM mappings for F00165, F9202, F9203, F9207, F9210, F9211 that point these tables to Data Dictionary - JD7333.
13. Activate the new OCM mappings.

# Creating a Software Master

---

This chapter explains how to create a software master from start to finish. Use the checklist provided earlier in this section to ensure that you complete all of the tasks.

This topic contains the following:

- Defining a software master
- Configuring change tables
- Validating a software master definition
- Creating a final software master
- Using a software master to update OneWorld

## Defining a Software Master

This task explains how to use the CD Configuration director to define a software master. If you have already defined a software master and you need to revise the definition, see “Revising a Software Master Definition” in this guide.

A software master definition provides the structure for your finished software master. It establishes what template and packages to use, what build steps to follow, and the directory structure of your final master. The software master definition is only the structure or outline of the master. Once you set up your software master definition, you validate it and then create the actual software master.

### **To define a software master**

---

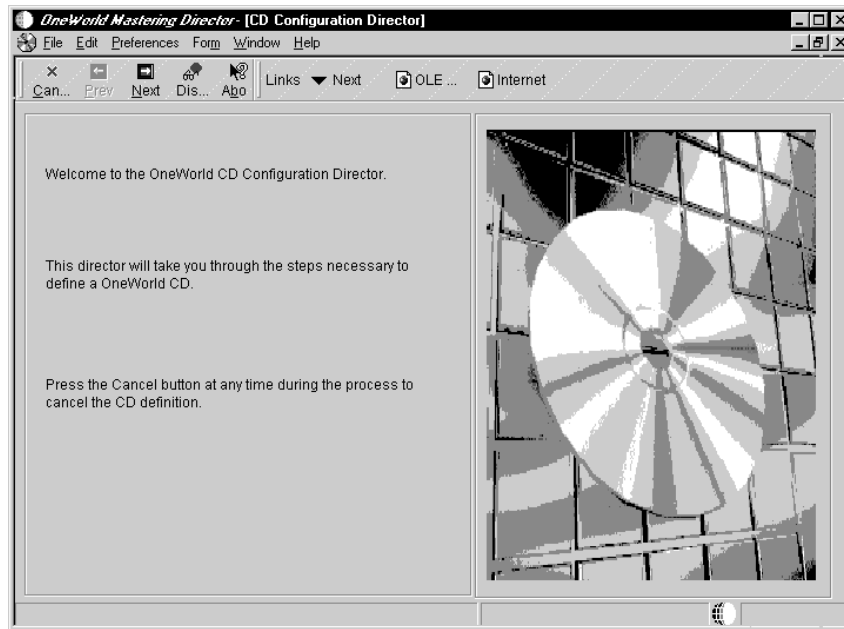
Sign onto OneWorld using the development environment where your modifications exist, such as DEV733.

1. From Product Packaging (GH962), choose OneWorld Product Packaging (P9640).

The Work With Software Mastering form appears.

2. Click Add.

The CD Configuration Director form appears.



3. Click Next.

The CD Information form appears.

The screenshot shows a window titled "OneWorld Product Packaging - [CD Information]". The menu bar includes "File", "Edit", "Preferences", "Form", "Window", and "Help". The toolbar contains "Can...", "Prev", "Next", "Dis...", "Abo", "Links", "Prev", "OLE...", and "Internet". The main area contains the following text:

Enter information about your new CD.

Provide a unique name, the OneWorld release level, type and a description. Finally, select the directory where you want the CD image to be placed.

The form includes the following fields:

Name	<input type="text" value="Software Master"/>
Release	<input type="text" value="B7333"/> B73.3.3 OneWorld
Type	<input type="text" value="09"/> Application Software Update
Build Phase	<input type="text" value="03"/> General Availability
Description	<input type="text" value="GA of a Software Master"/>
Image Path...	<input type="text" value="C:\B7"/>

On the right side of the window, there is a 3D rendering of a CD disc.

4. Complete the following fields:

- Name

Enter a unique name for the software master definition you are adding, such as B7333 Update.

- Release

Enter the OneWorld release for the software master definition, such as B7333. This is the release currently installed on the client workstation you are using.

- Type

Enter which type of CD you want to master. In this case, it is an ASU CD, which is the OneWorld default. This field indicates which software template to use when you further define the master. Templates include the build steps and directory structure necessary to create a software master. For information about templates, see the “Software Master Templates” section in this guide.

- Build Phase

Enter which development phase your software master definition represents, such as an alpha, beta, or general availability phase.

- Description

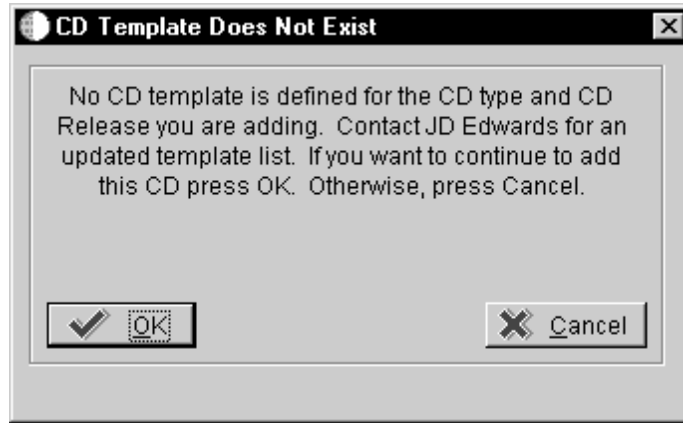
Optional. Enter a description of the software master definition.

- Image Path

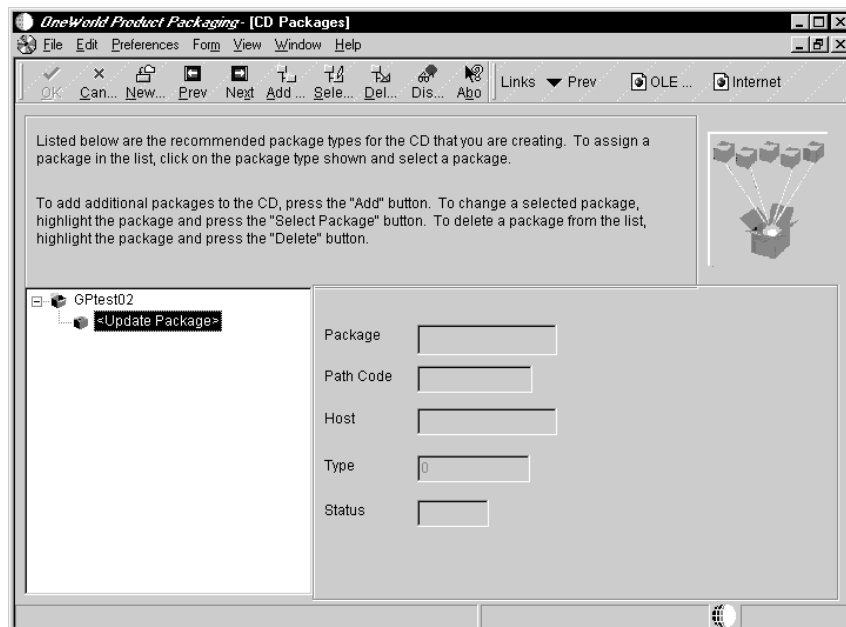
Enter the directory path where you want your software master definition to reside in the database on your enterprise. If the directory does not exist, OneWorld will create it for you when you run the software mastering process. The directory you indicate will be the root directory for the software master you create. It will contain the contents of your master.

5. Click Next.

If you chose a template that appears on the list of values when you click the visual assist button from the Type field but the template is not on your enterprise, the CD Template Does Not Exist form appears. You have probably entered the incorrect release number. Click OK to continue adding your software master definition, or click Cancel to stop adding the definition.

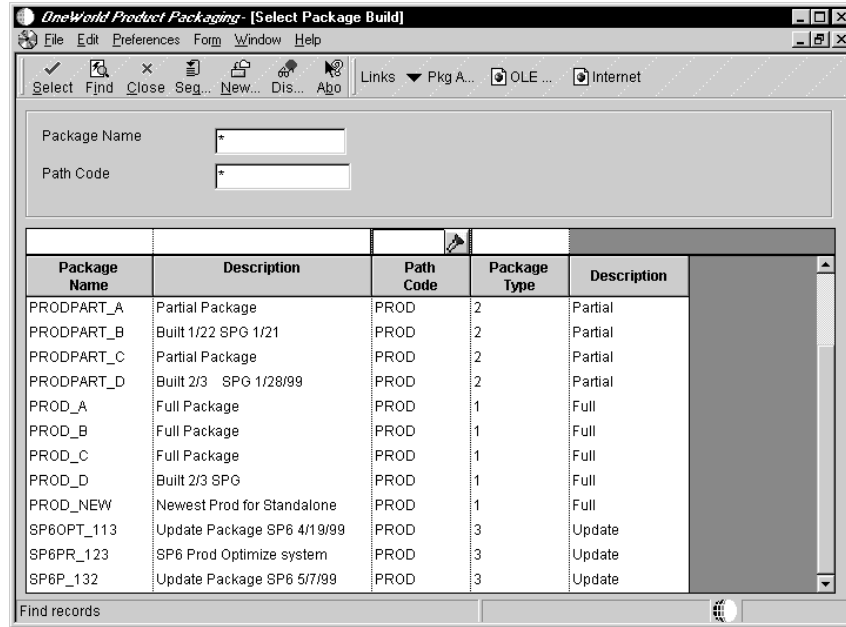


If the template you chose via the Type field was set up to include packages, the CD Packages form appears.



6. Choose a package marker, such as Update Package. If the package is undefined, the Select Package Build form automatically appears; otherwise, click Select Package. You can also add packages by clicking New Package Build.

The Select Package Build form appears.



7. If you need to assemble or define a package or both, choose one of the following from the Form menu:

- Pkg Assembly

See “Assembling a Package” in the *Package Management Guide* for information about how to assemble a package.

- Pkg Build

See “Defining a Package Build” in the *Package Management Guide* for information about how to define a package.

**Note:** Make sure that you compress any packages that you build.

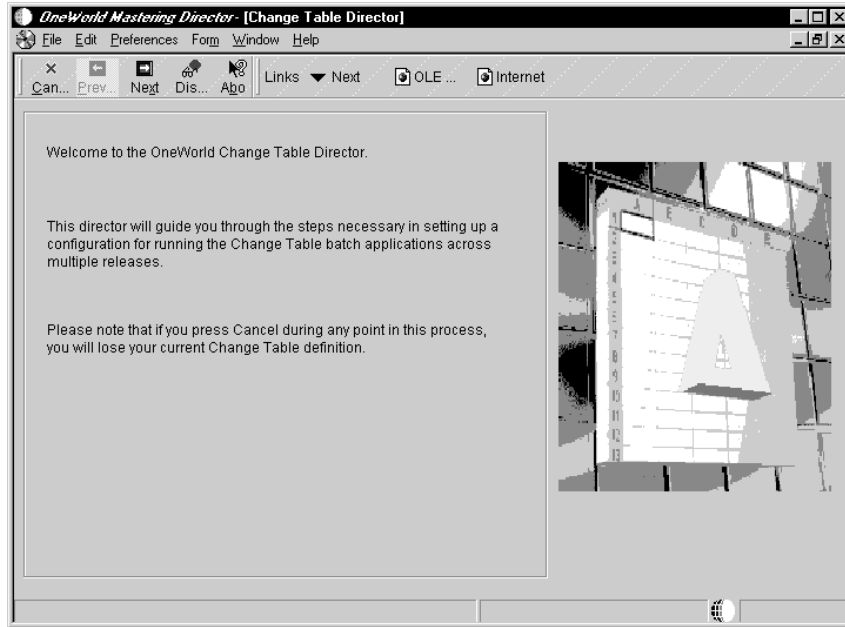
8. On the Select Package Build form, find and choose a package, then click Select.

The CD Packages form reappears.

9. Choose the package marker that appears. For an ASU CD, there is only one package required. For other types of software masters there might be more than one package; see “Working with a Software Master Definition” in this guide for other features of this form.

10. Click Next.

After you click Next, and if the template you chose was set up to include change table configurations, the Change Table Director form appears.



11. Continue to the next task, which explains how to configure your change tables.

Field	Explanation
Name	The unique name given to a software master.
Release	The OneWorld release number, such as B7331, of the software master. Use the visual-assist button to view a complete list of values for this field.
Type	The type of software master you want to master, such as an Update CD. The type corresponds to the software master template. For example, the Update CD template includes the build steps and directory structure necessary to define an update software master. Use the visual-assist button to view a complete list of values for this field.
Build Phase	This is the phase of development that your software master represents, such as an alpha, a beta, or a general availability phase. You can add or change the phases, since this is a user defined code. Use the visual-assist button to view a complete list of values for this field.
Description	A short description of the software master.
Image Path	An enterprise directory path that you specify, which will be the root directory containing the contents of your software master.



Field	Explanation
Package	<p>A package describes where on the server to find the components you want to deploy to workstations. There are three package types:</p> <p>Full: Contains the full suite of OneWorld applications (all specifications).</p> <p>Partial: A minimum configuration of OneWorld. This package type allows users to load the desired applications at run-time rather than initially installing all applications.</p> <p>Update: OneWorld objects contained in this type of package are loaded after the workstation receives the package and the user signs on to OneWorld. If the update package includes objects without the corresponding specifications, old versions of the application are deleted from the workstation and replaced by the current version the next time the user accesses that application. Update packages are always deployed on the date and time specified by the OneWorld administrator.</p>
Path Code	<p>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.</p> <p>For OneWorld, a Path Code is a pointer to a specific set of objects. A Path Code is used to locate:</p> <ul style="list-style-type: none"> <li>• Central objects - The central location where objects are deployed from. Objects such as specifications are stored in a relational database, other objects such as DLLs and sources are stored on a file server. Both types make up a Path Code.</li> <li>• Replicated objects - A copy or replicated set of the central objects must reside on each client and server that run OneWorld. The Path Code indicated in the directory that these objects are located in.</li> </ul>
Type	Used for transaction processing of CD Revisions.
Status	<p>While a package is being assembled, the status for the package is In Definition. When the assembly has been completed, the user can change the status to Definition Complete. Only packages with a status of Definition Complete can be submitted for a package build.</p>

## Configuring Change Tables

This task explains how to set up a change table configuration using the Change Table director. Create change table configurations to indicate to the Product Packaging Tools the change tables you want included with the software master. When you install the finished software master, the installation process will update the enterprise's control tables with the change tables you indicate. See the "Glossary" in this guide for definitions of change and control tables.

The following example illustrates the need for change table configurations: The home office in Denver, USA makes some software changes and wants to update an enterprise that is not networked to the Denver enterprise. The other enterprise, which has its own set of OneWorld software and control tables, is in Paris, France. The Denver enterprise creates a software master that includes the package containing the OneWorld software changes as well as the change tables that contain data dictionary and user defined codes changes, which for this example are the only control table changes since the Paris enterprise was last updated. The Denver enterprise writes the software master to a CD and ships it to Paris. When the Paris enterprise installs the software master, the change tables will update the Paris control tables, which will make the tables concurrent with the Denver control tables.

### **To configure change tables**

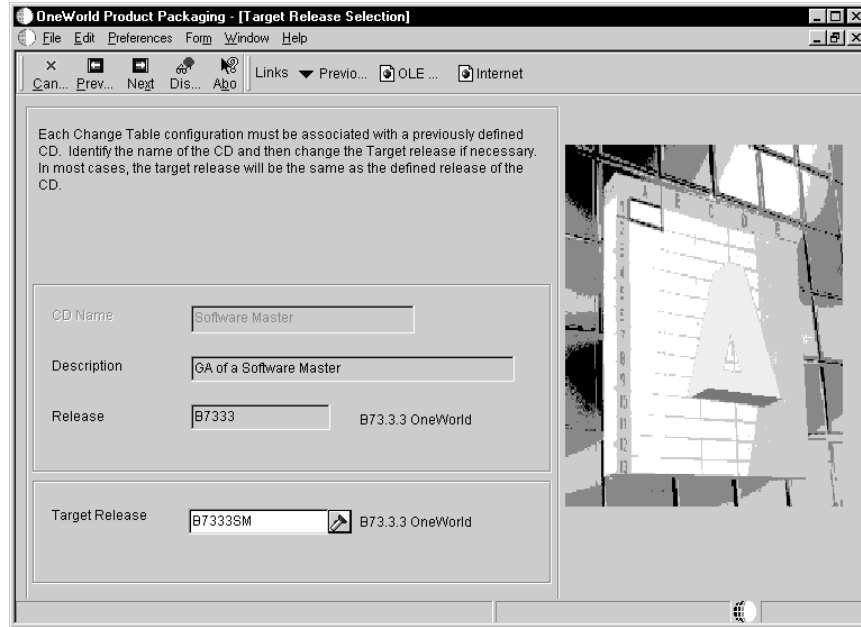
---

From the Change Table Director, which automatically appears during the mastering process if required by the template

1. Click Next.

Alternatively, you can access the director from Product Packaging (GH962). Choose Change Table Configurations (P9642) and when it appears, click Add. On the Change Table Director, click Next.

The Target Release Selection form appears.



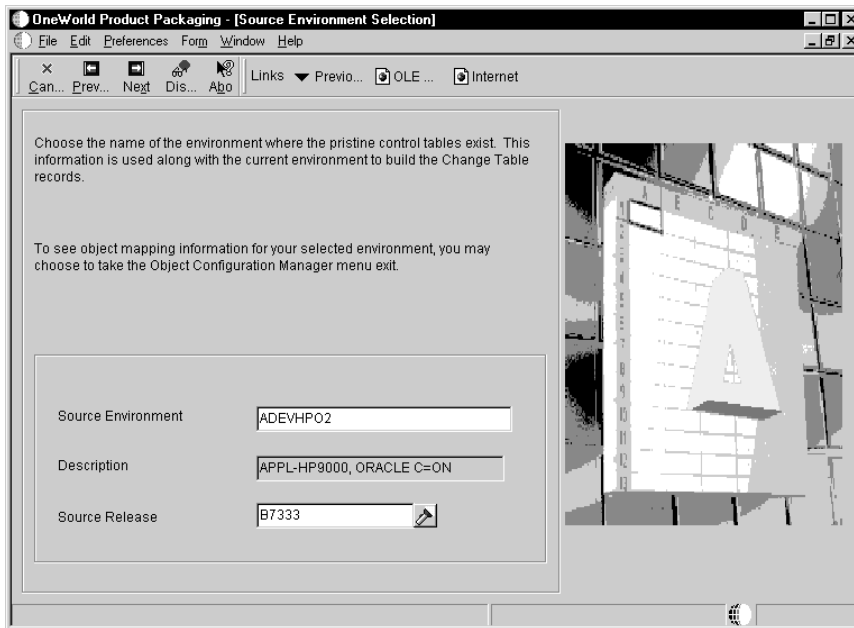
2. Complete the following field, and then click Next:

- Target Release

Change the target release to a unique release name that you are mastering. This name needs to be unique, such as B7333TAX for the 1099 tax update. Choose the release name using the visual-assist button.

If the release name does not appear in the user defined codes table, you need to create the name. On the Select User Defined Code form (which appears when you choose the visual-assist button), from the Form menu, choose Revisions. See “Adding a User Defined Code” in the *OneWorld Foundation Guide* for information.

The Source Environment Selection form appears.



3. Complete the following fields, and then click Next:

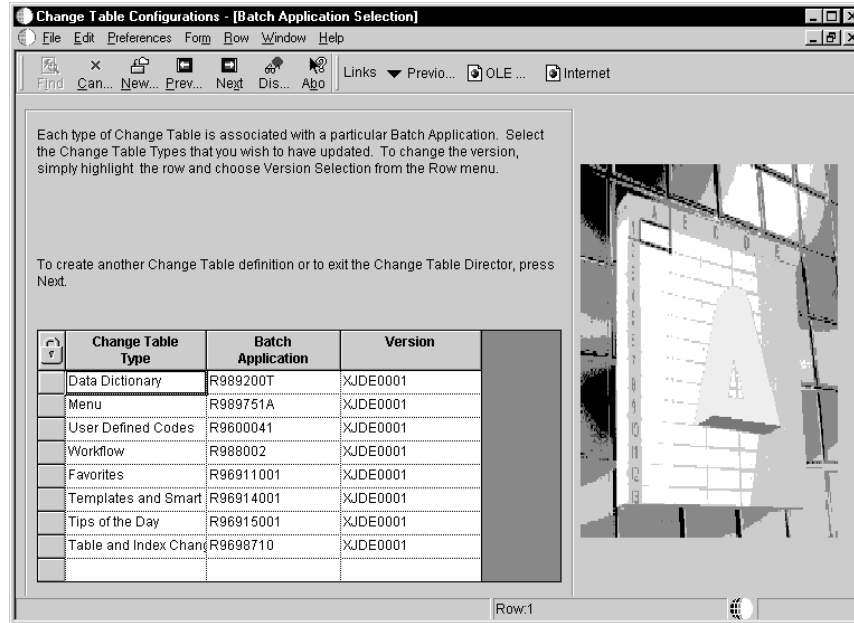
- Source Environment

Enter the name of the environment where the pristine control tables exist. OneWorld builds the change table records from this information.

- Source Release

Verify that the source release matches the release and the cumulative update level, such as B7333, of the source environment that you want. OneWorld automatically populates this field based upon the source environment.

The Batch Application Selection form appears. This form lists the available change tables along with the batch application and version that will create the change table.



4. To choose the default change tables that you want with your software master, double-click the gray button to the left of the change table row. A checkmark appears on the button. You can choose from the following change tables:

- Data Dictionary

Choose this change table if you made any adds or changes to your data dictionary. The batch process compares the source and target data dictionary tables and creates a change table that contains any adds and changes.

- Menu

Choose this change table if you made any adds, changes, or deletes to your menus. The batch process compares the source and target menu tables and creates a change table that contains any adds, changes, and deletes.

- User Defined Codes

Choose this change table if you made any adds or changes to your user defined codes. The batch process compares the source and target user defined codes tables and creates a change table that contains any adds and changes.

- Workflow

Choose this change table if you made any adds or deletes to your workflow table. The batch process compares the source and target workflow tables and creates a change table that contains any adds and deletes.

- Favorites

Choose this change table if you made any adds or deletes to your favorites table. The batch process compares the source and target favorites tables and creates a change table that contains any adds and deletes.

- Templates and Smart Fields

Choose this change table if you made any adds, changes, or deletes to your report director templates or smart fields tables. The batch process compares the source and target tables and creates a change table that contains any adds, changes, and deletes.

- Tips of the Day

Choose this change table if you made any adds, changes, or deletes to your tips of the day tables. The batch process compares the source and target tables and creates a change table that contains any adds, changes, and deletes.

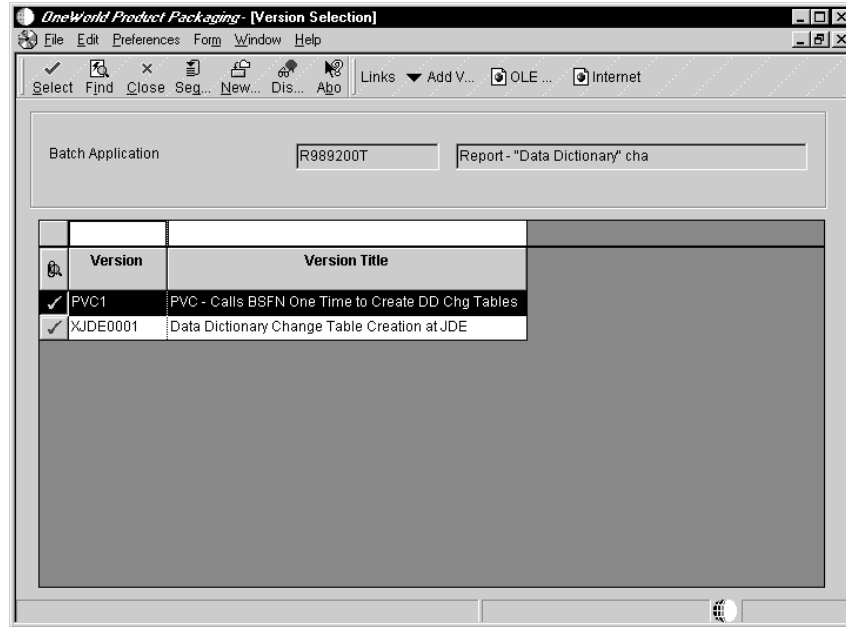
- Table and Index Changes

Choose this change table if you made any adds or changes to your table and index specifications. The batch process compares the source and target specifications and creates a change table that contains any adds and changes.

Double-click on the checkmark to remove it.

5. To choose a different version or to choose multiple versions, choose a change table row and from the Row menu, choose Version Selection.

The Version Selection form appears.



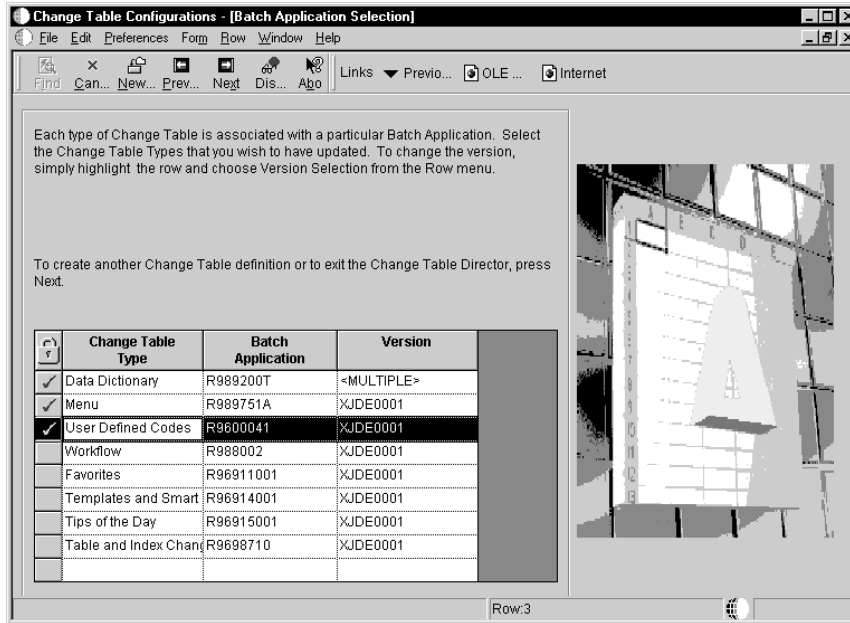
6. Choose one or more versions and click Select, or double-click the gray button to the left of a version.

A checkmark appears to the left of the versions you chose.

7. After choosing the versions that you want, click Close.

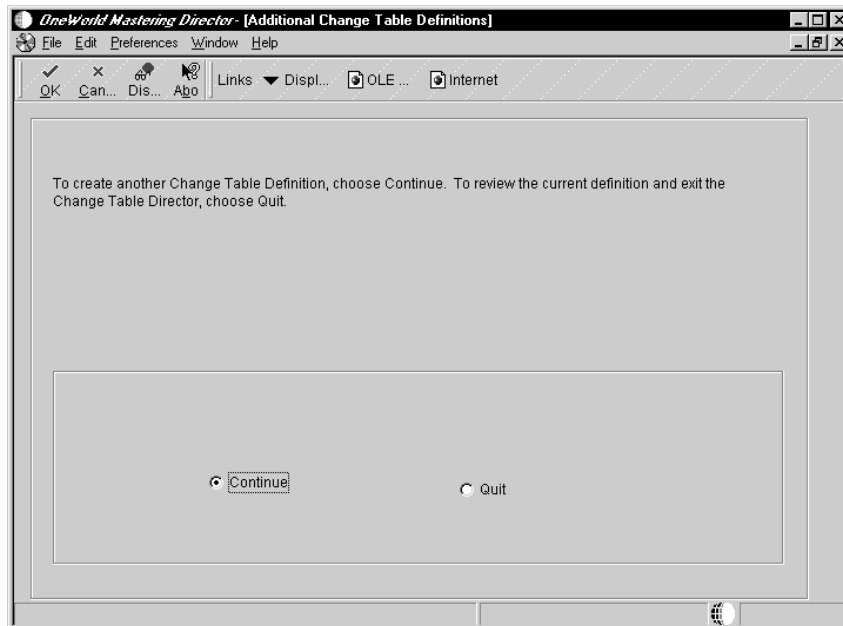
The Batch Application Selection form reappears.

A checkmark appears to the left of the change table row you chose. The change tables that you want built for the software master must have checkmarks next to them or OneWorld will ignore them. If you chose more than one version, the word <MULTIPLE> appears under the Version column for that change table.



8. Either choose another change table row and from the Row menu choose Version Selection, or click Next.

If you click Next, the Additional Change Table Definitions form appears.



9. Complete one of the following:
  - Choose Continue and click OK to return to the Target Release Selection form to define additional change table configurations for a new source and target release combination.
  - Choose Quit and click OK.



The CD Revisions form appears. Use this form to revise your software master definition. See “Revising a Software Master Definition” and “Revising or Submitting Change Tables” in this guide for information about revising what you just created.

10. On CD Revisions, click OK.

This saves your software master definition and the CD Revisions form disappears.

11. Continue to the next task, which explains how to validate the definition you just created.

Field	Explanation
Target Release	The OneWorld release number, such as B73.3.1, for which you are building the software master.
Source Environment	The name of the source environment where the pristine control tables reside. OneWorld creates the change tables for the software master from this source.
Source Release	For World, the release number as defined in the Software Versions Repository.  For OneWorld, the release number as defined in the Release Master.
Change Table Type	A user defined name or remark.  ..... <i>Form-specific information</i> .....  The type of change table, such as data dictionary or user defined codes.

## Validating a Software Master Definition

This task explains how to run a report to validate the software master definition that you set up and what to look for in the report to help with your validation. The report runs in final mode. The report lists whether the build step is validated or not. The report will change the status of those build steps that do not have errors.

 **To validate a software master definition**

From Product Packaging (GH962)

1. Choose OneWorld Product Packaging (P9640).

The Work With Software Mastering form appears.

2. Click Find.

Existing software master definitions appear.

3. Choose the master you want to validate, then click Validate.

The Report Output Destination form appears.

4. Specify whether to send the report to a printer or to view an online version of the report using Acrobat Reader, and click OK.

The report runs. All steps without errors whose status was In Definition or Defined are promoted to the status of Validated and they will work properly during the software mastering process. Otherwise, the build step will have either a warning or an error message:

- Warning messages will not stop the creation of a software master. The message specifies what the warning is. If a build step has a warning message, you should verify that the build step is how you want it to be before proceeding.
- Error messages will stop the creation of a software master. You must fix any build steps that contain error messages. The message specifies the error.

5. Continue to the next task, which explains how to create the final software master.

## Creating a Final Software Master

This task explains how to create the final software master image from a definition that you previously created. You can master a single definition, or choose to master all definitions of a specific release number.

### Before You Begin

- Ensure that at least 1GB of space is available for the TEMP directory of the workstation from which you use the Product Packaging Tools. This memory requirement is the most memory you might need for a software master. The actual size of the master depends upon the size of the package you create.

### **To create a final software master**

---

From Product Packaging (GH962)

1. Choose OneWorld Product Packaging (P9640).

The Work With Software Mastering form appears.

2. Complete one of the following:
  - To create a software master from one specific definition, choose the definition you want to create, then click Master CD.
  - To create software masters for all definitions of a specific release, click Master Release. A form appears in which you enter the release number of the definitions you want to master.

Either choice runs a batch process that creates your software master based upon the definition you chose.

**Note:** The software mastering process will stop when it encounters a manual build step. Complete the manual build step, choose that build step, and click Execute Step. This advances the status of that step to 60 (Built). Click Master CD or Master Release to continue the software master process.

See “Revising a Software Master Definition” in this guide for more information about build steps.

3. To view the status of the mastering process while it is running, from the View menu, choose Refresh.
4. The last build step automatically validates your software master. Review the report that this step creates.

If the validation report does not show any errors, you have successfully created a software master. This report validates that the number of directories match between the software master definition and the final software master. It also provides details about the file counts and the total size of the master in megabytes.

This process automatically creates a self-extracting executable file and a CAB file and places them in the \$TEMP/final directory using the name of the software master followed with the .exe and .cab extensions. Use the executable (.exe) file to update OneWorld. Both files include the full software master with all of its directories. If you have a size restriction, such as when downloading from the Internet, you can use the CAB file (which is always smaller than the executable file), but you will need to use a third-party application to uncompress the CAB file.

5. Continue to the next task, which explains how to update OneWorld using the software master you just created.

## Using a Software Master to Update OneWorld

This task explains how to update OneWorld using the software master you created.

### ► **To use a software master to update OneWorld**

---

On the deployment server

1. Run the self-extracting executable file. You can either copy the executable file to your deployment server and run it there, or you can run the executable off the Internet, as long as you access the Internet from your deployment server.

The executable file locates the OneWorld PLANNER directory on your deployment server and creates the software master image beneath that directory.

**Note:** You can also use the CAB file to create the software master image on your deployment server, but you need to use a third-party application to uncompress the file.

2. Use the *Application Software Update Guide* for the information needed to update OneWorld with the software master.

# Software Master





## Software Master

This section explains how to repair and delete a software master after you have created it. This section also explains how to revise, copy, proof, validate, and delete a software master definition after you have created it.

### See Also

- “Creating a Final Software Master” in this guide for information on how to convert your software master definition into the final software master that you can deploy to your enterprise
- “Defining a Software Master” in this guide for information on how to create a software master definition

This section contains the following:

- Working with a software master
- Working with a software master definition







## Working with a Software Master

---

This chapter explains how to repair and delete a software master. Repair the master if you find that something is wrong with an object or that an object is missing. Objects in the software master refer to the following:

- OneWorld objects (such as applications and business functions)
- Control/change tables (such as data dictionary items or user defined codes)
- Application data (such as the Address Book table)

This chapter contains the following:

- Adding a repair to a software master
- Deleting a repair to a software master
- Deleting a software master

### Adding a Repair to a Software Master

This task explains how to repair a software master. You accomplish this by adding or revising objects and submitting them back into the final software master. Once you submit repairs back to the software master, you need to recreate the master.

See the “Software Master Creation” section in this guide for information on the software master process.

#### **To add a repair to a software master**

---

From Product Packaging (GH962)

1. Choose OneWorld Product Packaging (P9640).

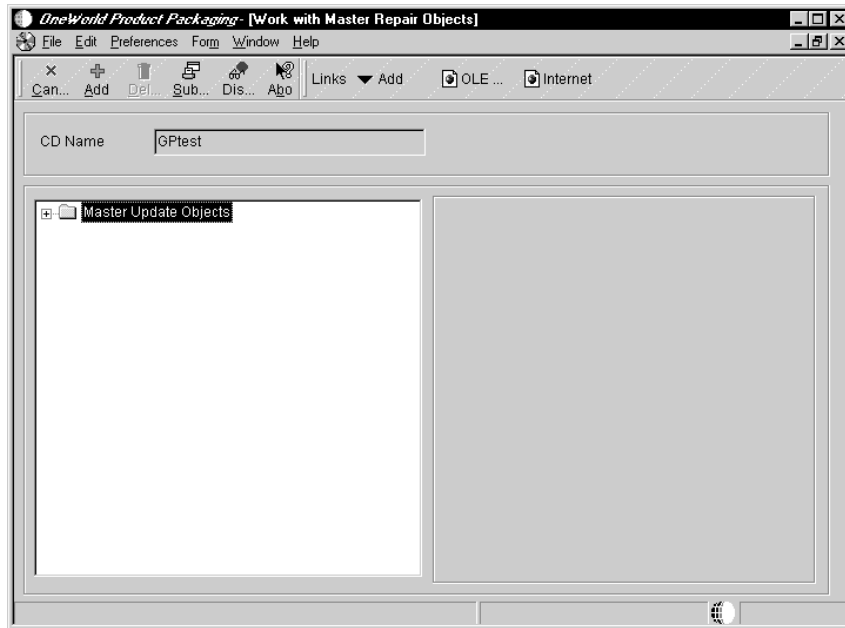
The Work With Software Mastering form appears.

2. Click Find.

Existing software masters and definitions appear.

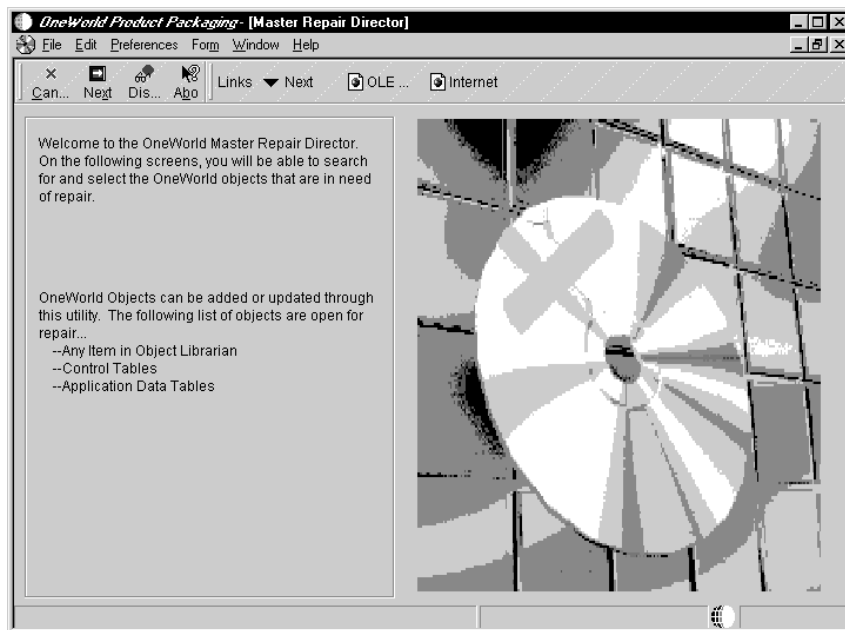
3. Choose the master you want to repair. To repair, the master should have a status of 60 (Built).
4. Click Repair CD.

The Work with Master Repair Objects form appears. Use the tree structure to view what objects that you chose for repair to the master.



5. Click Add.

The Master Repair Director appears.



6. Click Next.

The Master Data Locations form appears.

7. Click the CD Information tab.

8. Complete the following fields:

- Comment

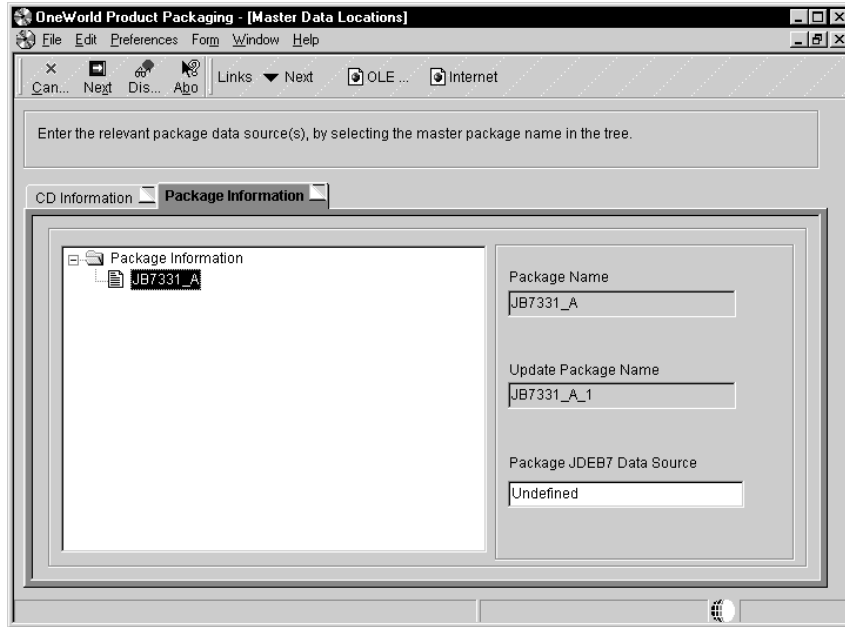
Enter or revise the comment for the software master.

- Pristine Data

Enter the name of the data source where you want to store the software master's pristine data.

- SQL Data - **For J.D. Edwards Internal Use Only.** You can have these fields appear on this form by using the processing options for this application
- DB/2 Data - **For J.D. Edwards Internal Use Only.** You can have these fields appear on this form by using the processing options for this application

9. Click the Package Information tab. Use the tree structure to view the package(s) that are part of the software master. OneWorld populates the Update Package Name field based upon your current package name and the repair number.

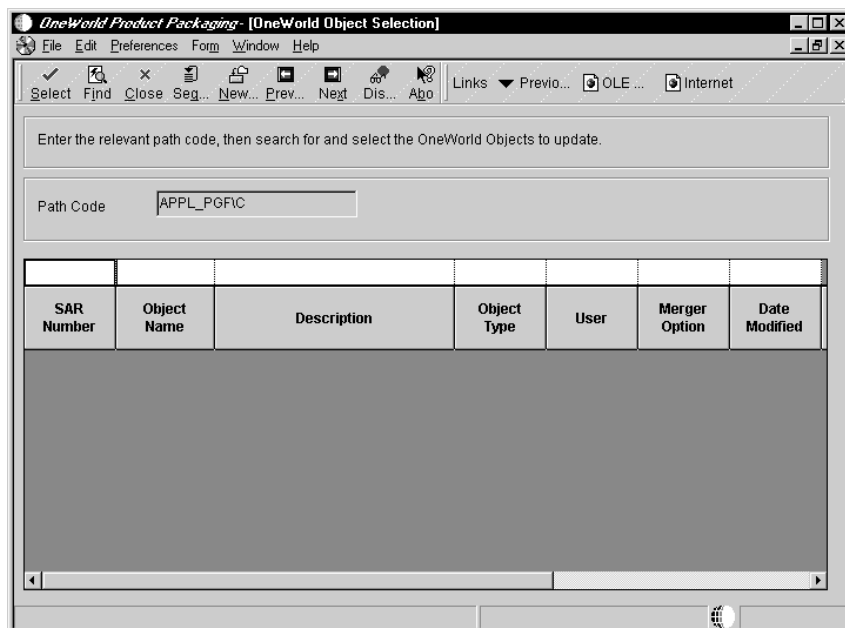


10. Complete the following field:
  - Package JDEB7 Data Source

Enter the name of the data source for the package you chose.

11. Click Next.

The OneWorld Object Selection form appears.



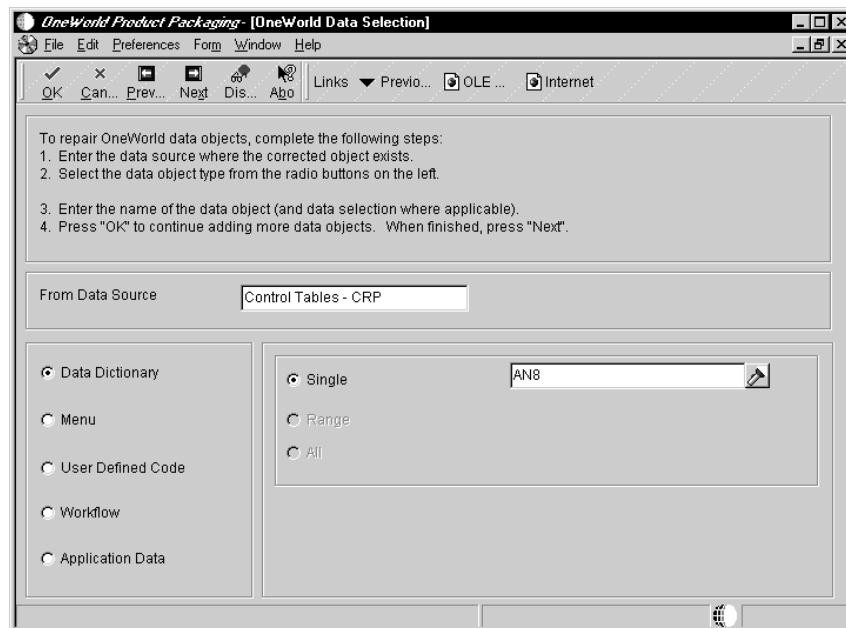
12. Find the OneWorld object you want repaired to the software master.

13. Choose the object, then click Select.

With each object you choose, that object is added to the list of objects you want repaired to the software master.

14. If you chose a batch application, the OneWorld Version Selection form appears. Choose the version or versions you want included with the batch application, click Select, and then click Close
15. Click Next.

The OneWorld Data Selection form appears. Use this form to choose control/change tables and application data you want repaired to the software master.



16. Complete the following field:

- From Data Source

Enter the name of the data source that contains the object you want repaired to the software master.

17. Choose one of the following data-object types and enter the specific value(s) for that object:

- Data Dictionary

Use the Single field to enter a data dictionary item that you want repaired to the software master. For example, enter AN8 to repair the Address Number data dictionary item.

- Menu

Use the Single field to enter a menu that you want repaired to the software master. For example, enter G021 to repair the ActivEra menu.

- User Defined Code

Two fields appear. To choose a user defined code, complete the Product and UDC fields, indicating the specific list of user defined codes you want repaired to the software master. For example, enter product code 98 and user defined code SY to repair the product code list.

- Workflow

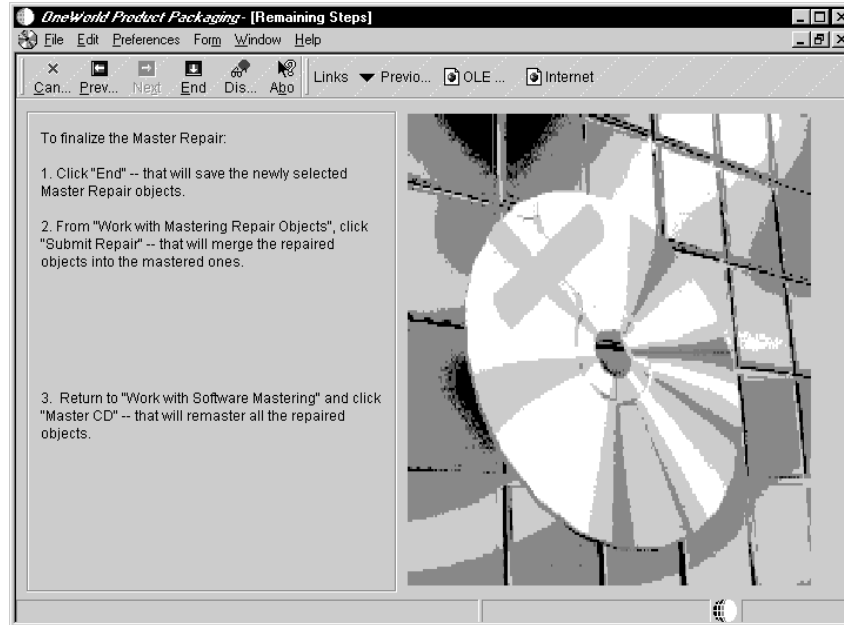
Choose the Single, Range, or All fields to choose a workflow process or processes. For example, enter 090003 into the Single field to repair the Set Up Business Unit workflow process.

- Application Data

Use the Single field to enter application data that you want repaired to the software master. For example, enter F0101 to repair the Address Book table.

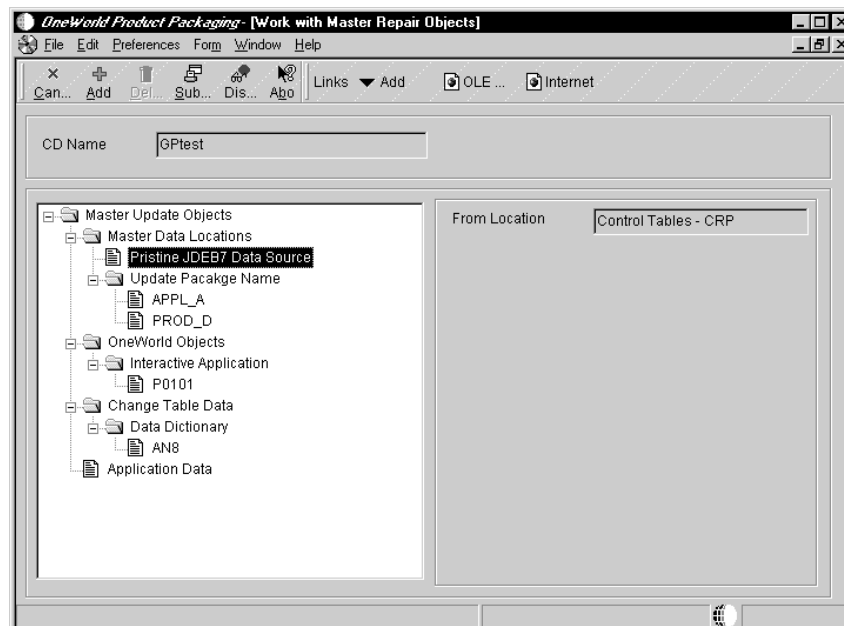
18. Click OK after providing the appropriate information for each data-object type that you want to repair. This allows you to continue choosing data objects.
19. Click Next after you finish choosing data objects.

The Remaining Steps form appears.



20. Click End.

The Work with Master Repair Objects form appears. If you chose objects to repair to the software master, OneWorld activates the Submit button on this form.



21. Click Submit Repair to move the chosen objects to the software master.

This automatically runs a report that lists all of the items you chose for this repair. You can also run this report independently using the Batch Versions application; see “Submitting a Report” in the *Enterprise Report*

*Writing Guide* for information about using the Batch Versions application. When running the report independently of repairing a software master, enter R9645 into the Batch Versions application and use data selection to enter the name of the software master for which you want the report.

22. You need to recreate your software master to build the objects. See the “Creating a Software Master” topic in this guide for information.

## Deleting a Repair to a Software Master

This task explains how to delete objects you chose for repair to the software master.

### **To delete a repair to a software master**

---

From Product Packaging (GH962)

1. Choose OneWorld Product Packaging (P9640).

The Work With Software Mastering form appears.

2. Click Find.

Existing software masters appear.

3. Choose the master you want to repair, then click Repair CD.

The Work with Master Updates form appears. Use the tree structure to view what objects you chose for repair to the master.

4. Choose an object, and click Delete.

## Deleting a Software Master

This task explains how to delete a software master.

**Important:** Everything associated with the definition will also be deleted, such as the change table configurations and repair director records.

### **To delete a software master definition**

---

From Product Packaging (GH962)

1. Choose OneWorld Product Packaging (P9640).



The Work With Software Mastering form appears.

2. Click Find.

Existing software masters appear.

3. Choose the master you want to delete, then click Delete.

A message box appears asking if you are sure you want to delete the software master definition.

4. Click OK.



## Working with a Software Master Definition

---

This chapter explains how to work with a software master definition after you have created one. Once you have completed the definition of a software master you can revise any of the information, which includes the build steps and directory structure. You can also copy the definition information to a new software master definition, delete the definition, and proof and validate it.

This section contains the following:

- Revising a software master definition
- Copying a software master definition
- Proofing a software master definition
- Validating a software master definition
- Deleting a software master definition

### Revising a Software Master Definition

This task explains how to revise a software master definition that you already created. This allows you to revise the information you entered into the CD Configuration Director, as well as the build steps and directory structure included with the template that you chose.

If you need to add a new software master definition, see “Defining a Software Master” in this guide.

#### **To revise a software master definition**

---

From Product Packaging (GH962)

1. Choose OneWorld Product Packaging (P9640).

The Work With Software Mastering form appears.

2. Click Find.

Existing software master definitions appear.

Release	Build Phase	Build Phase Description	CD Type	Type Description	CD Name	
B7333	03	General Availability	01	Setup CD	B7333 Change Tables	Pre-alpha
B7333	03	General Availability	09	Application Software U	QINGASU	Qing ASU
B7333	03	General Availability	09	Application Software U	Software Master	GA of a So
B7333	03	General Availability	09	Application Software U	TSTASUCD	Test ASU
B7333	03	General Availability	09	Application Software U	TSTASUCD1	T
B7333	03	General Availability	11	Custom CD	Change Table	
B7333	03	General Availability	11	Custom CD	Qing Test	Change T
B7333	01	Alpha	01	Setup CD	B7333 Alpha Masters	B7333 Alp
B7333	01	Alpha	01	Setup CD	B7333 JanAlpha Masters	Mastering
B7333	01	Alpha	01	Setup CD	B7333 Pre-Alpha Testing	Pre-Alpha
B7333	01	Alpha	01	Setup CD	testing versions	
B7332	03	General Availability	01	Setup CD	B7332 GA Masters-SetupCD	GA- B733
B7332	03	General Availability	09	Application Software U	10	Tools SP f
B7332	03	General Availability	09	Application Software U	3298324	Freegoods
B7332	03	General Availability	09	Application Software U	4	Model
B7332	03	General Availability	09	Application Software U	7	Tools Mod
B7332	03	General Availability	09	Application Software U	ASU Creation	Applicatio

- Choose the master you want to revise, then click Select.

The CD Revisions form appears.

OneWorld Product Packaging - [CD Revisions]

File Edit Preferences Form View Window Help

OK Can... Prev... Pro... Add... Del... Loa... Dis... Abo

Links Proof... OLE... Internet

CD Information Build Steps Directory Structure Director Steps

Name: Software Master

Type: 09 Application Software Update

Release: B7333 B73.3.3 OneWorld

Build Phase: 03 General Availability

Status: 20 Defined

Description: GA of a Software Master

Comment:

Image Path... C:\B7

- To refresh the information shown on any tabs on the CD Revisions form from the View menu, choose Refresh.
- To validate the information shown on any tabs, click Validate. A report runs in proof mode to validate the definition. To run this report in final mode, see “Validating a Software Master Definition” in this section.

## 6. Click the CD Information tab and revise the following fields:

- Status

Verify the status of the software master definition.

- Description

Optional. Enter a description of the software master definition.

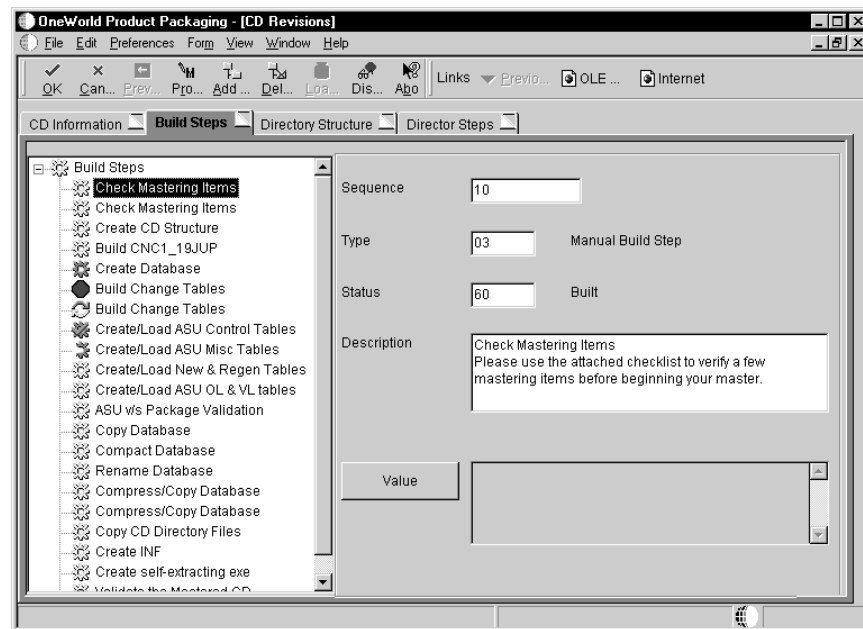
- Comment

Optional. The mastering process populates this field with status information about the process. You can also enter any additional information about the software master definition, but OneWorld will overwrite that information when a subsequent process completes.

- Image Path

Verify the directory path where you want your software master to reside on your enterprise. If the directory does not exist, OneWorld will create it for you when you run the software mastering process. The directory you indicate will be the root directory for when you create your final software master. See the “Software Master Creation” section in this guide for information about running the software master process.

## 7. Click the Build Steps tab.



8. To revise a build step, choose the step and complete the following fields:

- Sequence

Verify the order in which you want the step performed during the creation of the software master definition. You can use the same sequence number for multiple build steps.

- Type

Verify which component type you want the build step to perform. For example, you can define steps to create file structures, build a package, or create INF files.

- Status

Verify the current status of the build step, which can indicate whether or not to run the particular step during the mastering process. For example, if there are manual steps that you need to perform during the mastering process, you can manually complete the step, change the step's status to 60 (Built) by choosing the step, and then, from the View menu, choose Execute Step.

The icons to the left of each build step illustrate the status of that step:



The status code is 10, In Definition



Gray Cog. The status code is 20, Defined.



The status code is 30, Validated.



The status code is 40, Processing.



The status code is 50, Failed.



Gold Cog. The status code is 60, Built.

The following fields are dynamic and appear depending upon the build step you chose:

- Description

Enter a description about the build step. The first line of the description appears as the text for the build step. This line of text appears in the tree view on the left side of the form. You may want to enter a short descriptive name in the first line of the Description field, such as Check Mastering Items, then press the Enter key before typing in a more robust description for the step.

- Target Folder

OneWorld may populate this field based upon the build step's definition in the software master template. To change this field, click the Target Folder button and use the Select CD Directory form to choose the target folder for this build step.

- Executable

OneWorld may populate this field based upon the build step's definition in the software master template. To change this field, click the Executable button and use the Select a Windows Executable form to choose an executable for this build step.

- Value/Parameter

Verify the value or parameter. To change this information, click the Value/Parameter button, then enter a value or parameter appropriate to the step you chose. This field is dynamic. The button name is either Value or Parameter, based upon the step that you chose. The form that appears when you click the button also depends upon which step you chose. For example, the Package Build step calls the Select Package Build form, from which you can choose a package; and the Build Change Tables step calls the Work With Change Table Definitions form, from which you can modify the change table configuration.

9. To add a step, click Add.

OneWorld creates a new step labelled Undefined and places it at the end of the steps.

10. Choose the Undefined step and complete the following fields:

- Sequence
- Type
- Status
- Description
- Value

11. To run one or more steps, choose the build step you want to run, and from the View menu either choose Execute Step to run just that step or choose Run From Step to run that step plus any steps that follow it.

This option runs the step and updates its status to 60 (Built). You might use this option for manual build steps. Manual steps are those that you need to perform yourself, such as an additional mastering step that is not part of the Product Packaging Tools. After you complete the manual step, use the Execute Step option to update the step to a status of 60 so that you can proceed to the next step in the process.

12. To add an attachment to a build step, choose the step and from the View menu, click Attachments.

The Media Objects form appears. See “Media Objects and Imaging” in the *System Administration Guide* for information about attaching media objects.

After attaching a media object, a paper clip appears in the upper-right corner of the Build Step tab for that build step.

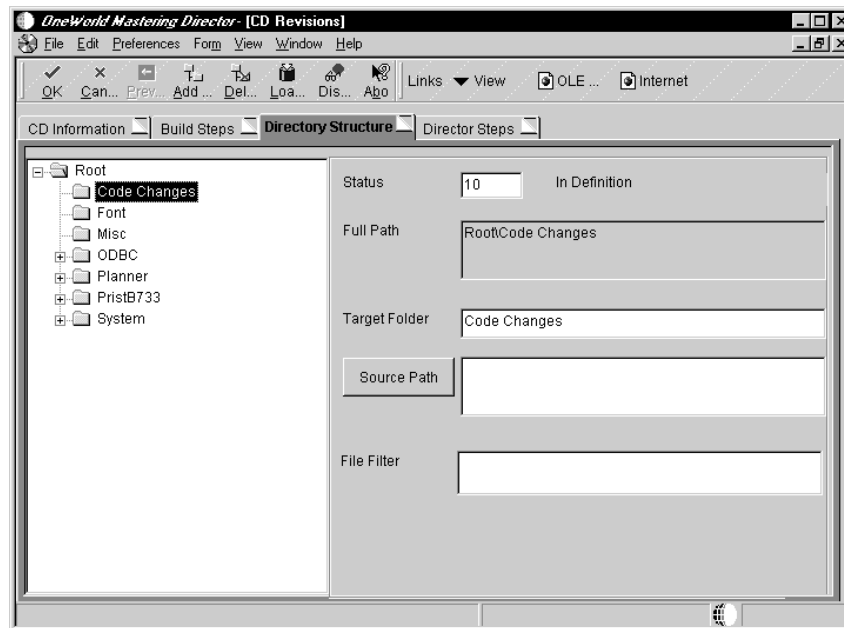
13. To renumber your build steps, from the Form menu, click Renumber.

OneWorld renumbers your build steps, updating the Sequence field. The renumbering begins with 10 and increments each step by 10 (10, 20, 30, etc.). OneWorld keeps the build steps in the same sequence that you or the software master template established.

14. To delete a step, choose the step, and click Delete.



15. Click the Directory Structure tab.



16. To revise a directory, choose the directory and complete the following fields:

- Status

Verify the status of the directory.

- Target Folder

OneWorld populates this field based upon the directory's definition in the software master template. Do not change the directory name that was defined in the template.

- Source Path

Verify the directory path of the source that will populate the component. This is the source path name of the data that you want to retrieve for the master and that you placed into the target folder. You can specify this path using either a relative or absolute path. See "Appendix A" for information about relative and absolute paths.

- File Filter

Verify the file filter (\*.\*) is the default). The file filter allows you to filter any data files you retrieve from the source directory. For example, you can use \*.\* to retrieve all files in the source directory.

- To add a subdirectory, click on a directory for which you want to add a subdirectory and click Add. You can choose any directory, including Root.

OneWorld creates a new directory labeled Undefined and places it at the bottom of the structure.

- Choose the Undefined directory and complete the following fields:
  - Status
  - Target Folder

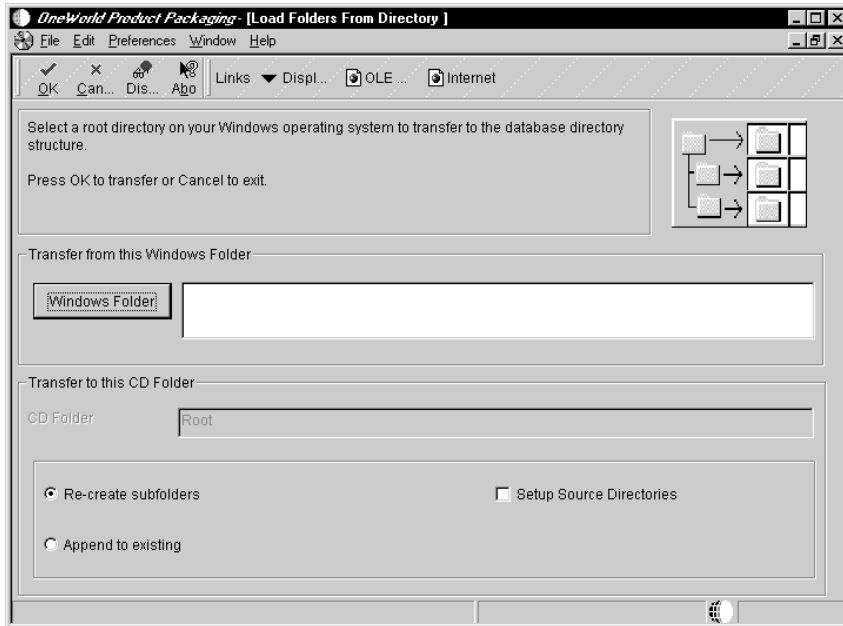
OneWorld populates this field for new directories with Undefined. Change this field to represent the new directory's name.

- Source Path
  - File Filter
- To load a subdirectory from an enterprise network directory, click on the directory that will be the root of the subdirectory that you want to load, which must already exist on your enterprise.

This feature copies any enterprise directory structure into the software master definition. Complete the following:

- Click Load.

The Load Folders From Directory form appears.



- Click Windows Folder.

The Select Directory form appears.



22. Find and choose a directory you want to load into your software master, then click OK.
23. On the Load Folders From Directory form, choose one of the following:
  - Re-create subfolders
 

This option replaces (deletes) the existing subdirectories and replaces them with the subdirectories you want to load into the software master.
  - Append to existing
 

This option appends to the existing subdirectories the subdirectories you want to load into the software master.
24. If needed, click the Setup Source Directories checkbox. When creating the software master, having this checked will automatically create the structure of the source directories and it will also copy all files within the source directories to the software master.
 

If you do not check Setup Source Directories, OneWorld will create the source directory structure within the software master and the directories will be empty.
25. Click OK.
26. To delete a directory, choose the directory, then click Delete.
27. When finished revising the software master definition, click OK.

Field	Explanation
Status	The status of a build step or directory structure within the mastering process. For example, the build step or directory structure could be at a status of defined, validated, or built. Use the visual-assist button to view a complete list of values for this field.
Comment	Short informational comments provided by OneWorld about each step of the mastering process. You can enter your own comments into this field, but OneWorld will overwrite any text currently in the field during the next mastering process.
Sequence	The order in which OneWorld processes build steps, usually in increments of 10. For example, a sequence of 10, 20, 30, etc.
Type	The process you want the build step to perform. For example, you can enter a code of 02 to indicate that the build step should perform a package build during the software mastering process. Use the visual-assist button to view a complete list of values for this field.
Full Path	<p>This field is dynamic and its name and function depend upon the build step highlighted. The possible names and functions are as follows:</p> <ul style="list-style-type: none"> <li>• Value - The value needed to complete the highlighted build step. For example, if the build step is a batch process, the value field contains the name of the batch process and its version, such as R95012\XJDE0001.</li> <li>• Parameter - The parameters needed for the highlighted build steps that process an executable program.</li> </ul> <p>..... <i>Form-specific information</i> .....</p> <p>The full (absolute) target path name of the highlighted directory, such as Root/planner/updatepkg.</p>
Target Folder	The name of the target folder, such as 'planner' or 'updatepkg.' OneWorld populates this field based upon the template used to define the software master. Change the name of the target folder only if you added the folder yourself.
Source Path	The directory on your enterprise that you want to use to populate the highlighted target folder. You can use an absolute (full) directory path name, or you can use a relative path name with a token. OneWorld copies the data contained in your source path, depending upon how you filter those files (see the File Filter field description for this information), into the highlighted target folder of your software master.

<b>Field</b>	<b>Explanation</b>
File Filter	The filter, which you can designate, for the files contained in the source path. For example, you can use *.* to retrieve all files in the source path, or *.exe to retrieve only executable programs.

## Copying a Software Master Definition

This task explains how to copy a software master definition that you already created. This allows you to copy the information that you entered into the CD Configuration Director, the build steps, and the directory structure.

### **To copy a software master definition**

---

From Product Packaging (GH962)

1. Choose OneWorld Product Packaging (P9640).

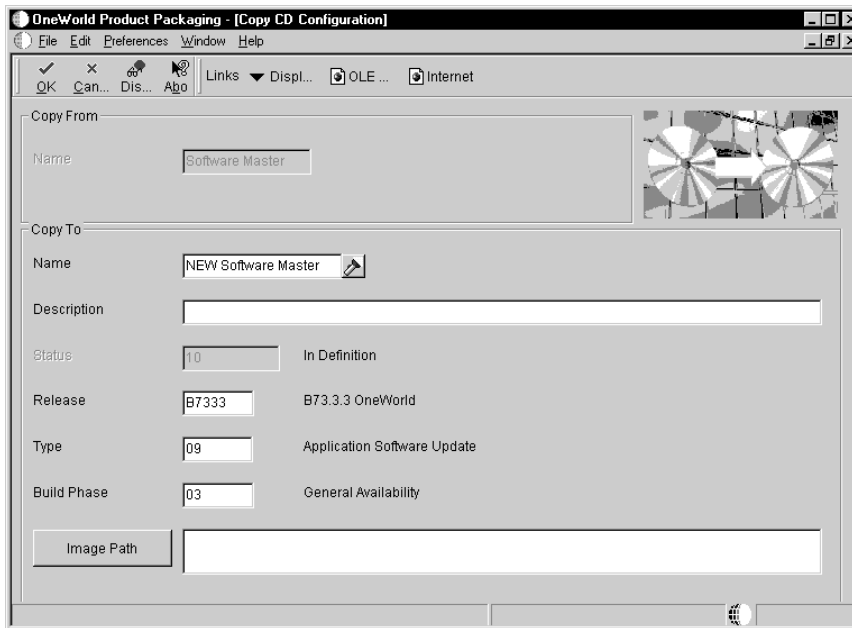
The Work With Software Mastering form appears.

2. Click Find.

Existing software master definitions appear.

3. Choose the master you want to copy, then click Copy.

The Copy CD Configuration form appears.



4. Complete the following fields, then click OK:

- Name

Enter a unique name for the software master definition you are copying, such as B7333 Update.

- Description

Optional. Enter a description of the software master definition.

- Release

Enter the OneWorld release for the copy of the software master definition, such as B7333.

- Type

In most cases you will not need to change the default value. If you need to, however, you can change the type of your software master when making a copy. For example, you can change the type from a Setup CD to an ASU CD. This does not change build steps or the directory structure. When you validate your software master, you will receive warning messages that your master does not match the default settings of the ASU CD template, because your master is actually based upon the Setup CD template. This will not prevent you from creating your software master.

- Build Phase

Enter which phase of development your software master definition represents, such as an alpha, beta, or general availability phase.

- Image Path

Enter the directory path where you want your software master to reside on your enterprise. If the directory does not exist, OneWorld will create it for you when you run the software mastering process. The directory you indicate will be the root directory for the software master that you create. See the “Software Master Creation” section in this guide for information about running the software master process.

## Proofing a Software Master Definition

This task explains how to proof software master definitions. This creates a report that details each build step in the definition, showing the sequence of the steps and what processes, if any, each step would run when creating the actual master.

Proofing a software master definition does not change any part of a software master’s status.

### **To proof a software master definition**

---

From Product Packaging (GH962)

1. Choose OneWorld Product Packaging (P9640).

The Work With Software Mastering form appears.

2. Click Find.

Existing software master definitions appear.

3. Complete one of the following:

- To proof one software master definition, choose the definition you want to create, then from the View menu, choose Proof CD.
- To proof all software master definitions for a specific release, from the View menu, choose Proof Release. A form appears in which you enter the release number of the definitions you want to proof.

With whichever proof method you chose, the Report Output Destination form appears.

4. Specify whether to send the report to a printer or to view an online version of the report using Acrobat Reader, and click OK.

A batch process runs a report that shows each build step in sequence and what processes, if any, each step runs when creating the actual master.

## Validating a Software Master Definition

This task explains how to run a report that validates the software master definition that you created. This task runs the report in final mode and the report will change the status of those build steps that do not have errors.

### **To validate a software master definition**

---

From Product Packaging (GH962)

1. Choose OneWorld Product Packaging (P9640).

The Work With Software Mastering form appears.

2. Click Find.

Existing software master definitions appear.

3. Choose the master you want to validate, then click Validate.

The Report Output Destination form appears.

4. Specify whether to send the report to a printer or to view an online version of the report using Acrobat Reader, and click OK.

The report runs. All steps without errors whose status was In Definition or Defined are promoted to the status of Validated. The report lists each build step that will run when you create your master. The report lists whether the build step is validated or not. If it is validated, the build step will work properly when creating the software master. Otherwise, the build step will have either a warning or error message:

- Warning messages will not stop the creation of a software master. The message specifies what the warning is. If a build step has a warning message, you should verify that the build step is how you want it to be before proceeding.
- Error messages will stop the creation of a software master. You must fix any build steps that contain error messages. The message specifies the error.



## Deleting a Software Master Definition

This task explains how to delete a software master definition. Everything associated with the definition will also be deleted, such as the change table configurations.

### **To delete a software master definition**

---

From Product Packaging (GH962)

1. Choose OneWorld Product Packaging (P9640).

The Work With Software Mastering form appears.

2. Click Find.

Existing software master definitions appear.

3. Choose the master you want to delete, then click Delete.

A message box appears asking if you are sure you want to delete the software master definition.

4. Click OK.



# Software Master Templates





## Software Master Templates

This section explains software master templates. It describes each build step that you can use in a template. Build steps are the processes used to create a software master. This section also explains how to add, revise, copy, and delete software master templates.

### See Also

- “Defining a Software Master” in this guide for information about the OneWorld Mastering Director and how to indicate the template you want to use when creating your software master

This section contains the following:

- Understanding software master templates
- Working with software master templates





# Understanding Software Master Templates

---

This chapter explains the components of the software master templates used by the Product Packaging Tool. Templates provide default information when creating a new software master. For example, to distribute objects to a disconnected enterprise, you would chose to build a master based upon the ASU (Application Software Update) CD template. The OneWorld Mastering Director would use the ASU CD template to determine what director steps, build steps, and directory structure to use when creating the software master. You can, of course, customize the default information provided by the templates.

This chapter contains the following:

- Details of all build step processes
- ASU CD Template
- Custom CD

## Details of all Build Step Processes

This topic explains all of the software mastering build step processes that a template can offer. Not all of these processes will appear on every template.

### Undefined

Type = 00

Performed Automatically = Not Applicable

This build step is the default process when adding a new build step to a software master or template. Once you add this build step, you need to define it for the master or template.

### Create CD File Structure

Type = 01

Performed Automatically = Yes

This build step creates the physical directory structure at the image-path location that you specified when creating the software master definition.

### Package Build

Type = 02

Performed Automatically = Yes

This build step builds the assigned package based upon the package assembly and build definitions.

### Manual Build Step

Type = 03

Performed Automatically = No, see the specific template definition in this chapter for complete information about this build step's performance.

This build step allows the administrator to instruct the user on how to perform a process that has not yet been automated. The software mastering batch process stops when it encounters this build step. Once you complete this build step, change its status to Succeeded (status code of 60), and restart the software mastering batch process.

### Pre-Build Package

Type = 04

Performed Automatically = Yes

For J.D. Edwards Internal Use Only.

### Pre-Build Pristine

Type = 05

Performed Automatically = Yes

For J.D. Edwards Internal Use Only.

### Build Cross-Reference

Type = 06

Performed Automatically = Yes

For J.D. Edwards Internal Use Only.



### **Build Data Dictionary**

Type = 07

Performed Automatically = Yes

For J.D. Edwards Internal Use Only.

### **Build In-House Package Database**

Type = 08

Performed Automatically = Yes

For J.D. Edwards Internal Use Only.

### **Build Planner Database**

Type = 09

Performed Automatically = Yes

For J.D. Edwards Internal Use Only.

### **Build Pristine Database**

Type = 10

Performed Automatically = Partial, see the specific template definition in this chapter for complete information about this build step's performance.

This build step creates the pristine data tables in the final pristine jdeb7.mdb database that corresponds to the given set of modules.

### **Build Store & Forward Database**

Type = 11

Performed Automatically = Yes

For J.D. Edwards Internal Use Only.

## Build Standalone Database

Type = 12

Performed Automatically = Yes

For J.D. Edwards Internal Use Only.

## Build Change Tables

Type = 13

Performed Automatically = Yes

This build step builds the change tables based upon the change table configuration that was set up as part of the software master definition.

## Master Central Objects

Type = 14

Performed Automatically = No

For J.D. Edwards Internal Use Only.

## Compress

Type = 15

Performed Automatically = Partial, see the specific template definition in this chapter for complete information about this build step's performance.

This build step compresses the final pristine jdeb7.mdb database into the data.cab file that OneWorld places within the final image path.

## Create INF

Type = 16

Performed Automatically = Yes

Please DO NOT modify this file by hand. This build step updates the deployment.inf file that the installation applications use during the update installation.

## Copy CD Directory Files

Type = 17

Performed Automatically = Yes

This build step copies the built package to the target image path. This step also copies to the target image path any additional files defined in the software master directory structure.

If you have already built the package and do not want the mastering process to build the package again, set the status of the Package Build step to 60. When you create your software master, the process will skip the package build and when it runs this Copy CD Directory Files step, it will copy the package.

## Virus Check

Type = 18

Performed Automatically = Partial

For J.D. Edwards Internal Use Only.

## Windows Executable

Type = 19

Performed Automatically = Partial, see the specific template definition in this chapter for complete information about this build step's performance.

This build step allows you to specify a Microsoft Windows executable or batch file as part of the mastering process.

## Data Cleanup

Type = 20

Performed Automatically = No

For J.D. Edwards Internal Use Only.

## Build Package Database

Type = 21

Performed Automatically = Yes

For J.D. Edwards Internal Use Only.

## Batch Process

Type = 22

Performed Automatically = Yes

This build step allows you to define a batch process to run as part of the software master definition.

## ASU CD Template

The ASU CD is the main template used with the Product Packaging Tools. This template is the basis for delivering OneWorld objects to distributed enterprises. The following table shows the build steps and their sequence for an ASU CD:

Sequence Number	Build Step (Type)
10	Check Mastering Items (03 - Manual Build Step)
20	Cleanup Temporary Files (22 - Batch Process) *
30	Create CD Structure (01 - Create CD File Structure)
40	Build (02 - Package Build)
50	Create Database (22 - Batch Process) *
60	Build Change Tables (13 - Build Change Tables)
70	Create/Load ASU Change Tables (22 - Batch Process) *
80	Create/Load ASU Control Tables (22 - Batch Process) *
90	Create/Load ASU Misc Tables (22 - Batch Process) *
100	Create/Load New & Regen Tables (22 - Batch Process) *
110	Create/Load ASU OL & VL tables (22 - Batch Process) *
120	ASU v/s Package Validation (22 - Batch Process) *
130	Copy Database (22 - Batch Process) *
140	Compact Database (22 - Batch Process) *
150	Rename Database (22 - Batch Process) *
160	Compress/Copy Database (22 - Batch Process) *
170	Copy Package and Feature INFs (22 - Batch Process) *
180	Copy CD Directory Files (17 - Copy CD Directory Files)
190	Create INF (16 - Create INF)
200	Create self-extracting exe (22 - Batch Process) *
210	Validate the Mastered CD (22 - Batch Process)

\* Additional information provided, see the following.

### Additional Build Step Information for the ASU CD Template

The following table provides additional information pertaining to a particular build step within the ASU CD template.

Sequence Number	Build Step (Type) with Additional Information
20	Cleanup Temporary Files (22 - Batch Process) - this build step launches the R9640L batch process using versions XJDE0001 through XJDE0005. This process deletes temporary files from the previous software master.
50	Create Database (22 - Batch Process) - this build step launches the R9640F batch process using version ZJDE0002 and the R9640K batch process using version XJDE0001. This process either creates an empty Microsoft Access database, if the database does not exist, or it clears the tables from the existing database. This process uses the following directory location and database name: \$TEMP/master/jdeb7.mdb
70	Create/Load ASU Change Tables (22 - Batch Process) - this build step launches the R9670 batch process using version XJDE0003. This process creates change tables in the database (\$TEMP/master/jdeb7.mdb), populating the tables from the Change Table data source.
80	Create/Load ASU Control Tables (22 - Batch Process) - this build step launches the R9670 batch process using version XJDE0001. This process creates control tables in the database (\$TEMP/master/jdeb7.mdb), populating the tables from the Control Table data source.
90	Create/Load ASU Misc Tables (22 - Batch Process) - this build step launches the R9670 batch process using version XJDE0002. This process creates miscellaneous tables in the database (\$TEMP/master/jdeb7.mdb), populating the tables based upon the selections you made for the objects in your ASU package, your change tables, and the software master definition.
100	Create/Load New & Regen Tables (22 - Batch Process) - this build step launches the R96700 batch process using version XJDE0001. This process creates tables in the database (\$TEMP/master/jdeb7.mdb) that are marked as new or regenerate in the table change table or the table conversion scheduler.
110	Create/Load ASU OL & VL tables (22 - Batch Process) - this build step launches the R9600400C batch process using version XJDE0001. This process copies to the jdeb7.mdb database the Object Librarian, Versions List, package assembly information for your master. Verify that the data source, which you can set in the processing options of this version, is set correctly.

120	ASU v/s Package Validation (22 - Batch Process) - this build step launches the R9671 batch process using version XJDE0001. This process cross validates, ensuring that objects, tables, and specifications in the ASU package definition match with the object change table, table change table, index change table, and table conversion scheduler in the software master.
130	Copy Database (22 - Batch Process) - this build step launches the R9640E batch process using version XJDE0002. This process copies the ASU database from \$TEMP/master/jdeb7.mdb to \$TEMP/data/jdeb7.mdb.
140	Compact Database (22 - Batch Process) - this build step launches the R9640G batch process using version ZJDE0001. This process uses Microsoft Access to compact the ASU database.
150	Rename Database (22 - Batch Process) - this build step launches the R9640H batch process using version XJDE0001. This process renames the jdeb7.mdb database to <i>packagename.mdb</i> , where <i>packagename</i> is the name of the package associated with the database.
160	Compress/Copy Database (22 - Batch Process) - this build step launches R9640E using version XJDE0003. This process copies the ASU database ( <i>packagename.mdb</i> ) from the \$TEMP/data directory to the final software master image path.
170	Copy Package and Feature INFs (22 - Batch Process) - this build step launches the R9600400E batch process using version XJDE0001 and the R9640E batch process using versions XJDE0032 through XJDE0034. This process copies deployment features INFs, the package INFs, and the feature INFs from the deployment server to \$TEMP directories. The INFs are then compressed in \$TEMP then copied to the final software master image path.
200	Create self-extracting exe (22 - Batch Process) - this build step launches the R9640J batch process using version XJDE0001. This process takes the entire software master image and creates a CAB file and a self-extracting executable file in the \$TEMP/final directory. You can use either of these files to deliver the software master. Uncompress the CAB file or run the executable file on the deployment server, then use the <i>Application Software Update Guide</i> to update OneWorld with the software master.

## Custom CD

The Custom CD is for general use. This CD type has no template. It is primarily used by a OneWorld administrator to create backups of source code, documents, and so forth, while still using the Product Packaging Tools. This CD type allows you to enter a directory structure that contains all of the files that you want loaded onto a custom CD image. Once completed, the administrator can move that image to a CD burner to create the actual CD.

# Working with Software Master Templates

---

This chapter explains how to add, revise, copy, proof, and delete software master templates. Once you add a template, you can use the OneWorld Mastering Director to create a software master definition using that template. See “Defining a Software Master” in this guide for information.

This chapter contains the following:

- Adding a software master template
- Revising a software master template
- Copying a software master template
- Proofing a software master template
- Deleting a software master template

## Adding a Software Master Template

This task explains how to add a software master template. If you have already set up the template and need to revise it, see “Revising a Software Master Template” in this chapter.

### **To add a software master template**

---

From Advanced Operations (GH9622)

1. Choose OneWorld Mastering Templates (P9640).

The Work With CD Templates form appears.

2. Click Add.

The Template Director form appears.

3. Click Next.

The Template Information form appears.

4. Complete the following fields:

- Name

Read-only. OneWorld populates this field with the information you enter into the following fields. For example, if you enter release B7333 with an 11 type and a 02 build phase, the name of the template will be B7333-11-02. OneWorld will not allow you to add a template with the same name as an already existing template.

- Release

Enter the OneWorld release for the software master template, such as B7333.

- Type

Enter which type of CD that you want the template to define, such as an ASU CD.

- Build Phase

Enter which phase of development your software master template represents, such as an alpha, beta, or general availability phase.

- Description

Optional. Enter a description of the software master template.

5. Click Next.

The CD Revisions form appears.

6. Set up the build steps and directory structure for the template; see the “Revising a Software Mater Template” topic for an explanation.

## Revising a Software Master Template

This task explains how to revise a software master template. If you need to add a new template, see “Adding a Software Master Template” in this chapter.

### **To revise a software master template**

---

From Advanced Operations (GH9622)

1. Choose OneWorld Mastering Templates (P9640).

The Work With CD Templates form appears.



2. Click Find.

Existing templates appear.

3. Choose the template you want to revise, then click Select.

The CD Revisions form appears.

4. To refresh the information shown on any tabs on the CD Revisions form from the View menu, choose Refresh.
5. To validate, in proof mode only, the information shown on any tabs, click Validate. This runs a report that validates the definition.
6. Click the CD Information tab and, if needed, revise the following fields:

- Status

Enter the status of the template, which should be “10 In Definition.”

- Description

Optional. Enter a description of the software master template.

- Comment

Optional. The mastering process populates this field with status information about the process. You can also enter any additional information about the software master template, but OneWorld will overwrite that information when a subsequent process completes as you define and create your software master.

7. Click the Build Steps tab. If needed, choose a step and revise the following fields:

- Sequence

Enter the order in which you want the step performed during the creation of the software master definition. You can use the same sequence number for multiple build steps.

- Type

Enter which component type you want the build step to perform. For example, you can define steps to create file structures, build a package, or create INF files.

- Status

Enter the starting status of the build step, which should be “10 In Definition.” The Product Packaging process changes this field during the subsequent stages of defining and creating a software master.

The following fields are dynamic and appear depending upon the build step you chose:

- Description

Enter a description about the build step. The first line of the description appears as the text for the build step. This line of text appears in the tree view on the left side of the form. You may want to enter a short descriptive name in the first line of the Description field, such as Check Mastering Items, then press the Enter key before typing in a more robust description for the step.

- Target Folder

Click the Target Folder button. Use the Select CD Directory form to choose the target folder for the build step you chose.

- Executable

Click the Executable button. Use the Select a Windows Executable form to choose an executable for this build step.

- Value/Parameter

Click the Value/Parameter button, then enter a value or parameter appropriate to the step you chose. This field is dynamic. The button name is either Value or Parameter, based upon the build step. The form that appears when you click the button also depends upon which step you chose. For example, the Package Build step calls the Select Package Build form from which you can choose a package, and the Build Change Tables step calls the Work With Change Table Definitions form from which you can modify the change table configuration.

8. If you need to add a step, click Add.

OneWorld creates a new step labelled Undefined and places it at the bottom of the steps.

9. Choose the Undefined step and complete the following fields:

- Sequence
- Type
- Status
- Description
- Value

10. To add an attachment to a build step, choose the step and from the View menu, click Attachments.

The Media Objects form appears. See “Media Objects and Imaging” in the *System Administration Guide* for information about attaching media objects.

After attaching a media object, a paper clip appears in the upper-right corner of the Build Step tab for that build step.

11. To renumber your build steps, from the Form menu, click Renumber.

OneWorld renumbers your build steps, which updates the Sequence field. The renumbering begins with 10 and increments each step by 10 (10, 20, 30, etc.). OneWorld keeps the build steps in the same sequence that you established.

12. To delete a step, choose the step, and then click Delete.
13. Click the Directory Structure tab and, if needed, revise each directory by choosing a directory and verifying the following fields:

- Status

Enter the status of the directory, which should be 10 In Definition.

- Target Folder

Enter the Target Folder path for the directory you chose.

- Source Path

Enter the directory path of the source that will populate the component. This is the source of the data that you want to retrieve for the master and placed into the target folder (directory). You can specify this path using either a relative or absolute path. See “Appendix A” for information about relative and absolute paths.

- File Filter

Enter the file filter (\*.\*) is the default) to filter data files that you retrieve from the source directory. For example, you can use \*.\* to retrieve all files in the source directory.

14. To add a subdirectory, click on a directory for which you want to add a subdirectory, and then click Add. You can choose any directory, including Root.

OneWorld creates a new directory labelled Undefined and places it beneath the directory that you chose.

15. Choose the Undefined directory and complete the following fields:

- Status

- Target Folder
  - Source Path
  - File Filter
16. To load a subdirectory from an enterprise network directory, click on the directory that will be the root of the subdirectory that you want to load. This directory must already exist on your enterprise.
  17. Click Load.

The Load Folders From Directory form appears.

18. Click Windows Folder.

The Select Directory form appears.

19. Find and choose a directory you want to load into your template, then click OK.
20. On the Load Folders From Directory form, choose one of the following:

- Re-create subfolders

This option replaces (deletes) the existing subdirectories and replaces them with the subdirectories you want to load into the template.

- Append to existing

This option appends to the existing subdirectories the subdirectories you want to load into the software template.

21. If needed, click the Setup Source Directories checkbox. When creating the software master, having this checked will automatically create the structure of the source directories and will copy all files within the source directories to the software master.

If you do not check Setup Source Directories, OneWorld will create the source directory structure within the software master and the directories will be empty.

22. Click OK.
23. When finished revising the template, click OK.

## Copying a Software Master Template

This task explains how to copy a software master template. This allows you to copy the information from the template, which includes the build steps and directory structure.

If you need to add a new software master template, and cannot copy and revise one that already exists, see “Adding a Software Master Template” in this section.

### ► To copy a software master template

From Advanced Operations (GH9622)

1. Choose OneWorld Mastering Templates (P9640).

The Work With CD Templates form appears.

2. Click Find.

Existing templates appear.

3. Choose the template you want to copy, then click Copy.

The Copy CD Configuration form appears.

4. Complete the following Copy To fields, then click OK:

- Description

Optional. Enter a description of the software master definition.

- Release

Enter the OneWorld release for the template, such as B7333.

- Type  
Enter which type of template you want to use when setting up your software master, such as the ASU CD template.
- Build Phase  
Enter which phase of development your template represents, such as an alpha, beta, or general availability phase.

## Proofing a Software Master Template

This task explains how to proof software master templates. This creates a report that details each build step in the definition, showing the sequence of the steps and what processes, if any, each step would run when creating the actual master.

Proofing a software master template does not change a build step's status.

### **To proof a software master template**

---

From Advanced Operations (GH9622)

1. Choose OneWorld Mastering Templates (P9640).

The Work With CD Templates form appears.

2. Click Find.

Existing templates appear.

3. Complete one of the following:
  - To proof one software master template, choose the template you want to create, and then from the View menu, choose Proof CD.
  - To proof all software masters templates for a specific release, from the View menu choose Proof Release. A form appears in which you enter the release number of the templates you want to proof.

The Report Output Destination form appears.

4. Specify whether to send the report to a printer or to view an online version of the report using Acrobat Reader, and click OK.

A batch process runs a report that shows each build step in sequence and what processes, if any, each step would run when creating the actual master.

## Deleting a Software Master Template

This task explains how to delete a software master template.

### **To delete a software master template**

---

From Advanced Operations (GH9622)

1. Choose OneWorld Mastering Templates (P9640).

The Work With CD Templates form appears.

2. Click Find.

Existing templates appear.

3. Choose the template you want to delete, then click Delete.

A message box appears asking if you are sure you want to delete the software master template.

4. Click OK.





# Change Table Configuration





## Change Table Configuration

This section explains how to work with a change table configuration after you have created one. Once you have completed the configuration of a change table you can revise, copy, or delete it.

### See Also

- “Configuring Change Tables” in this guide for information on how to create a change table configuration

This topic contains the following:

- Revising or submitting change tables
- Copying a change table configuration
- Deleting a change table configuration





# Revising or Submitting Change Tables

---

This task explains how to build and submit change table configurations to a software master after you have already created the software master and need to revise or add change tables. You can ignore this task if you have not yet created your software master. If the template you chose for your software master required change tables, the software master will build the change tables you specified during the setup of your software master definition.

## Revising or Submitting Change Tables

Complete the task below to work with change tables.

► **To revise or submit change tables**

---

From Product Packaging (GH962)

1. Choose Change Table Configurations (P9642).

The Work With Change Table Definitions form appears.

The screenshot shows a software window titled "Change Table Configurations - [Work With Change Table Definitions]". The window has a menu bar (File, Edit, Preferences, Form, Row, Window, Help) and a toolbar with icons for Select, Find, Add, Copy, Del..., Close, Seq..., New..., Dis..., and Ago. Below the toolbar are fields for "CD Name" (Software Master), "Release" (B7333), and "Description" (GA of a Software Master). A message states: "The Configuration Status must be Validated before the associated Change Table Batch Application can be submitted. The checked items show those that are ready to run or have already run successfully." Below this is a table with the following data:

	Source Release	Target Release	Source Environment	Change Table Type	Batch Application	Version
<input checked="" type="checkbox"/>	B7333	B7333SM	ADEVHPO2	Data Dictionary	R989200T	XJDE0001

At the bottom of the window, there is a "Find records" field and a "Row:1" indicator.

2. Complete the following field, and then click Find:

- CD Name

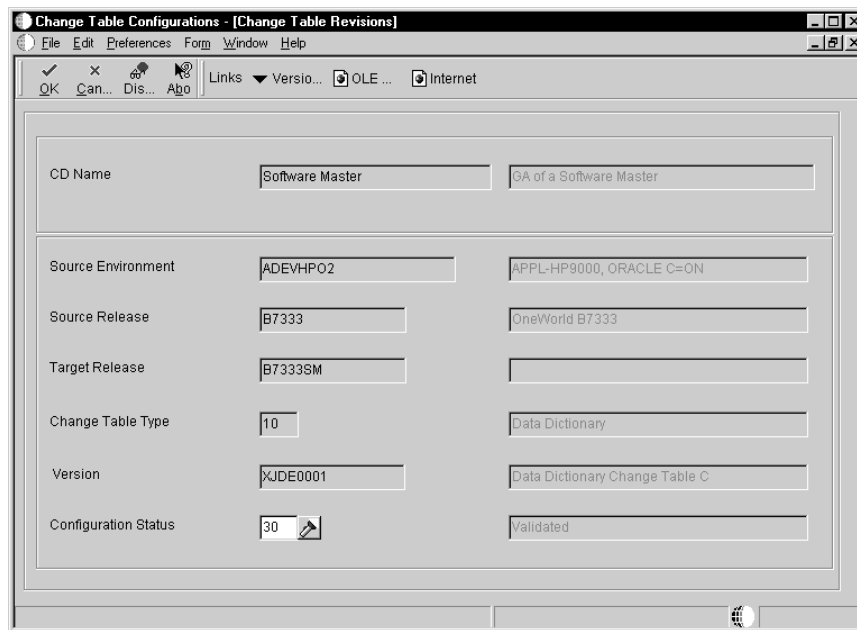
Enter the name of the software master for which you want to view change table configurations.

Existing change table configurations appear with the following:

- A check mark appears to the left of each table configuration row that either completed its build successfully or is ready to be built.
- A circle with a line through it appears to the left of any table configuration that failed its build or is not at a valid status for the configuration to be submitted.

3. If you need to rerun a particular change table batch process and the change table row does not have a checkmark to the left of it, correct any issue that may exist, choose the change table row, and click Select.

The Change Table Revisions form appears.



4. Complete the following field, then click OK:

- Configuration Status

Change to or verify that the status is 30 (Validated), which allows this change table to be built and submitted to the software master.

5. To submit change tables, complete one of the following:

- To submit an individual change table row, choose the row and from the Row menu, choose Submit.

- To submit multiple change table rows, choose the rows using the Control or Shift keys with your mouse, and from the Row menu, choose Submit.
- To submit all of the change table rows, from the Form menu, choose Submit.

OneWorld builds the change tables you chose and includes them with the software master. If you revised existing change tables, they will replace themselves in the software master. If you added change tables, they will be appended to the software master.





# Copying a Change Table Configuration

---

This task explains how to copy change table configurations from one master definition to another master definition. This process copies all change table configurations associated with the master you are copying from.

## Copying a Change Table Configuration

Complete the following task to copy a change table configuration.

**► To copy a change table configuration**

---

From Product Packaging (GH962)

1. Choose Change Table Configurations (P9642).

The Work With Change Table Definitions form appears.

2. Complete the following field, and then click Find:

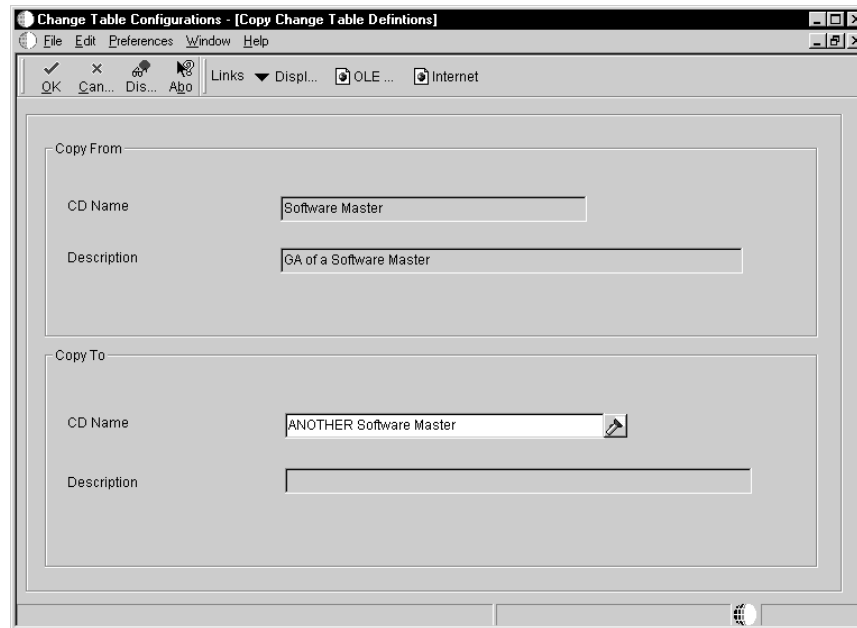
- CD Name

Enter the name of the software master from which you want to copy its change table configurations.

Existing change table configurations appear.

3. Choose a change table row and click Copy.

The Copy Change Table Definitions form appears.



4. Complete the following field, and then click OK:

- CD Name

Enter the name of the software master definition to which you want to copy the change table configurations. The master you are copying to must already exist.

# Deleting a Change Table Configuration

---

This task explains how to delete change table configurations you have added to your software master definition.

## **To delete a change table configuration**

---

From Product Packaging (GH962)

1. Choose Change Table Configurations (P9642).

The Work With Change Table Definitions form appears.

2. Complete the following field, and then click Find:
  - CD Name

Existing change table configurations appear.

3. Choose a change table row and click Delete.



# Data Cleanup





## Data Cleanup

This section explains how to use the data cleanup application to create procedures and templates to clean up your software master data. The data cleanup application can delete obsolete data, standardize existing data, and provide integrity, summary, and error reports about the data. You can clean up the following:

- Data dictionary tables and specifications
- Auto pilot tables
- User defined codes and types
- Menu tables
- Object librarian tables
- Central object tables

This section contains the following:

- Working with data cleanup procedures
- Working with data cleanup templates







# Working with Data Cleanup Procedures

---

This chapter explains how to work with data-cleanup procedures for your software master. This function of the Product Packaging Tools includes an application to add, revise, copy, and delete a set of data cleanup steps. You define how you want these steps to clean up your software master data.

This chapter contains the following:

- Adding a data cleanup procedure
- Revising a data cleanup procedure
- Copying a data cleanup procedure
- Executing a data cleanup procedure and build step
- Deleting a data cleanup procedure

## Adding a Data Cleanup Procedure

This task explains how to add a data cleanup procedure. This allows you to add procedure information and the steps necessary to clean up data.

**Note:** Even though this Product Packaging Tools function is primarily for creating data cleanup procedures, you can use it to create any type of procedure to run a batch process or Microsoft Windows executable.

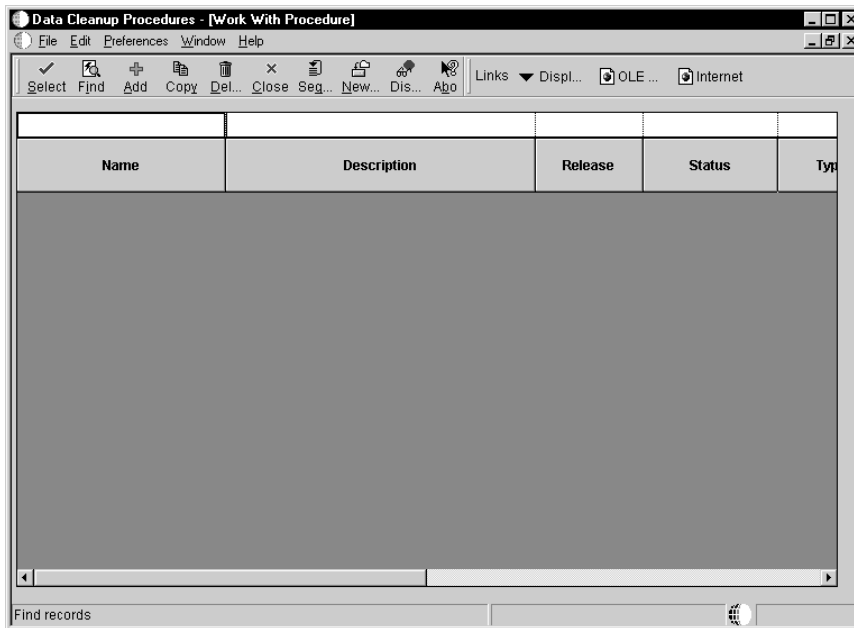
### **To add a data cleanup procedure**

---

From Product Packaging (GH962)

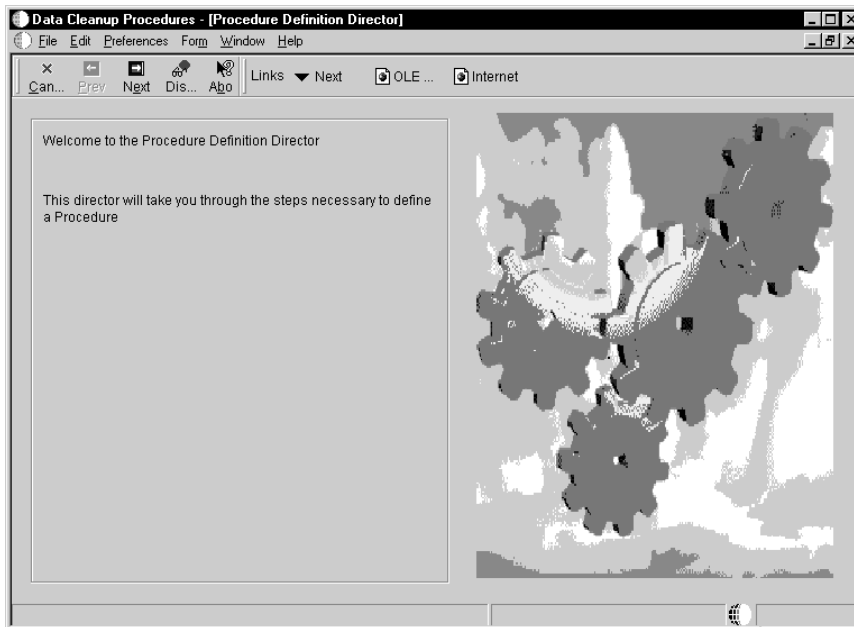
1. Choose Data Cleanup Procedure Application (P9646).

The Work With Procedure form appears.



2. Click Add.

The Procedure Definition Director form appears.



3. Click Next.

The Add Procedure form appears.

Enter information about your new procedure.

Enter the unique name and other information related to the procedure

Name

Description

Comment

Status  In Definition

Procedure Release

Template Name

4. Complete the following fields:

- Name

Enter a unique name for the data cleanup procedure you are adding, such as Object Librarian Cleanup.

- Description

Optional. Enter a description of the data cleanup procedure.

- Comment

Optional. Enter a comment about the data cleanup procedure.

- Status

Verify the status of the data cleanup procedure. Set the status to 30 (Validated) before you execute the procedure.

- Procedure Release

Enter the OneWorld release that pertains to the data cleanup procedure, such as B7333. This is the release currently installed on the client workstation you are using.

- Template Name

Optional. Enter the name of the data cleanup template that you want to work from for the procedure you are adding.

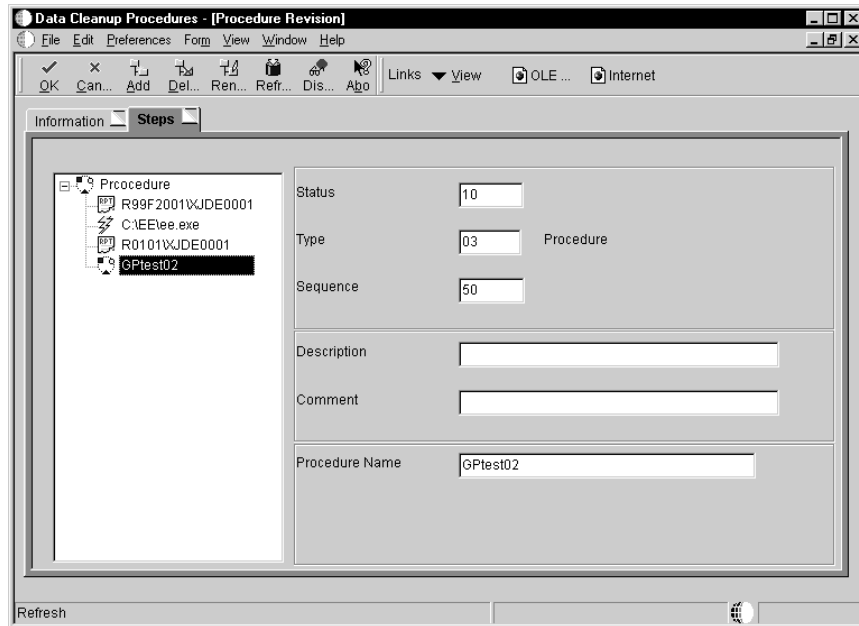
5. Click Next.

The Procedure Revision form appears.

The screenshot shows a software window titled "Data Cleanup Procedures: [Procedure Revision]". The window has a menu bar with "File", "Edit", "Preferences", "Form", "View", "Window", and "Help". Below the menu bar is a toolbar with icons for "OK", "Can...", "Add", "Del...", "Ren...", "Refr...", "Dis...", and "Abp". To the right of the toolbar are buttons for "Links", "Run M...", "OLE...", and "Internet". The main content area has two tabs: "Information" (selected) and "Steps". The "Information" tab contains a form with the following fields: "Name" (Gary), "Description" (empty), "Status" (10 In Definition), "Comment" (empty), and "Release" (B733). There is a graphic of gears on the right side of the form.

6. To refresh the information shown on either the Information tab or the Steps tab of the Procedure Revision form, from the View menu, choose Refresh.
7. Click the Information tab, and complete the following fields:
  - Description
  - Status
  - Comment
  - Release

8. Click the Steps tab.



9. To add a step, click Add.

OneWorld creates a new step labelled Undefined and places it at the end of the highlighted procedure displayed in the tree. If you add a step to a subprocedure, the Undefined step appears at the end of that subprocedure. A form appears asking you to verify your add to the subprocedure; the add changes that procedure, whether it is a subprocedure or not, wherever it appears.

10. Choose the Undefined step and complete the following fields:

- Status

Enter the status of the data cleanup build step. Set the status to 30 (Validated) before you execute the step.

- Type

Enter which type of build step you want to add. For example, you can add steps that are batch processes, a Microsoft Windows executable program, or another data cleanup procedure.

- Sequence

OneWorld automatically displays a sequence number based upon where you add the build step. You can change this sequence number to move the build step up or down in the procedure. You can use the same sequence number for multiple build steps.

- Description  
Optional. Enter a description of the data cleanup build step.
  - Comment  
Optional. Enter a comment about the data cleanup build step.
11. Complete the following fields, which appear if you are adding a batch process:
- Object Name  
  
Enter a OneWorld name of a batch process. For example, enter R9647R to have the step execute the Object Librarian - Object Relationship Cleanup report.
  - Version  
  
Enter the name of the report version for the batch process you entered. For example, XJDE0002, the Final Mode version.
12. Complete the following fields, which appear if you are adding a Microsoft Windows executable program:
- Executable  
  
Enter the name of a Microsoft Windows-compatible executable program. For example, the `xcopy.exe` program.
  - Parameters  
  
Enter the parameters for the executable program. You can enter command-line switches, directories, and files. For example, for the `xcopy.exe` program, you could enter a copy-from parameter of `c:\temp\*.*`, and you can enter a command-line switch of `\s`.
13. Complete the following field, which appears if you are adding a data cleanup procedure:
- Procedure Name  
  
Enter the name of an existing data cleanup procedure. The data cleanup procedure that you enter must already exist. If the procedure appears on the visual-assist form, then it exists.
14. When you tab out of the last field or click on a step in the tree structure on the left side of the form, the data cleanup application saves your step.
15. When finished adding steps, click OK.

## Revising a Data Cleanup Procedure

This task explains how to revise a data cleanup procedure that you already created. This allows you to revise the information and steps you created within the procedure. Be aware that if you revise a data cleanup procedure, those revisions are reflected wherever that procedure is used, such as when the revised procedure is nested within another procedure.

If you need to add a new data cleanup procedure, see “Adding a Data Cleanup Procedure” in this chapter.

### **To revise a data cleanup procedure**

---

From Product Packaging (GH962)

1. Choose Data Cleanup Procedure Application (P9646).

The Work With Procedure form appears.

2. Click Find.

Existing data cleanup procedures appear.

3. Choose the procedure you want to revise, then click Select.

The Procedure Revision form appears.

4. To refresh the information shown on either the Information tab or the Steps tab of the Procedure Revision form, from the View menu, choose Refresh.

5. Click the Information tab, and revise the following fields:

- Description

Optional. Enter a description of the data cleanup procedure.

- Status

Verify the status of the data cleanup procedure. Set the status to 30 (validated) before you execute the procedure.

- Comment

Optional. Enter a comment about the data cleanup procedure.

- Release

Verify the OneWorld release that pertains to the data cleanup procedure, such as B7333. This is the release currently installed on the client workstation you are using.

6. Click the Steps tab.

7. To revise a build step, choose the step, and complete the following fields:

- Status

Verify the status of the data cleanup build step. Set the status to 30 (Validated) before you execute the step.

- Type

Verify which component type you want the build step to perform. For example, you can define steps as a batch process, a Microsoft Windows executable, or another data cleanup procedure.

- Sequence

Verify the order in which you want the step performed during the execution of the procedure. You can use the same sequence number for multiple build steps.

- Description

Optional. Enter a description of the data cleanup build step.

- Comment

Optional. Enter a comment about the data cleanup build step.

The following fields appear if you are revising a batch process:

- Object Name

Verify the OneWorld name of the batch process.

- Version

Verify the name of the report version for the batch process.

The following fields appear if you are revising a Microsoft Windows executable program:

- Executable

Verify the Microsoft Windows executable program.



- Parameters

Verify the parameters of the executable.

The following field appears if you are revising a data cleanup procedure:

- Procedure Name

Verify the name of the data cleanup procedure.

8. To renumber your build steps, from the View menu, click Renumber.

OneWorld renumbers your build steps, updating the Sequence field. The renumbering begins with 10 and increments each step by 10 (10, 20, 30, etc.). OneWorld keeps the build steps in the same sequence that you or the data cleanup template established.

9. To delete a step, choose the step, and click Delete.
10. When finished revising the data cleanup procedure, click OK.

## Copying a Data Cleanup Procedure

This task explains how to copy a data cleanup procedure that you already created.

### **To copy a data cleanup procedure**

---

From Product Packaging (GH962)

1. Choose Data Cleanup Procedure Application (P9646).

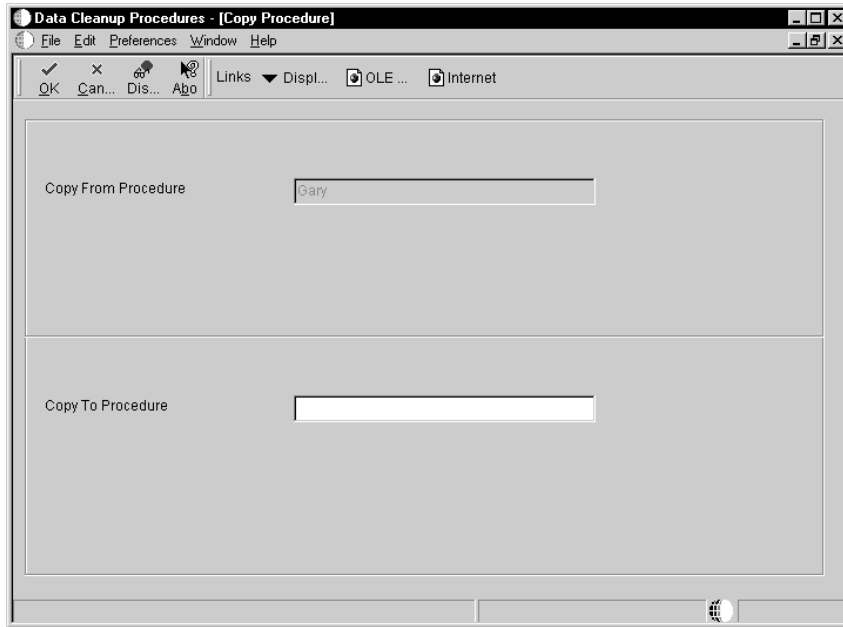
The Work With Procedure form appears.

2. Click Find.

Existing data cleanup procedures appear.

3. Choose the procedure you want to copy, then click Copy.

The Copy Procedure form appears.



4. Complete the following fields, and then click OK:
  - Copy From Procedure  
Enter the name of an existing data cleanup procedure.
  - Copy To Procedure  
Enter the name that you want to call the new data cleanup procedure.

## Executing a Data Cleanup Procedure and Build Step

This task explains how to execute a data cleanup procedure and build step from the data cleanup application.

**Note:** While this task explains how to execute procedures from the data cleanup application, you can also use the software-master build steps to define a step that executes your data cleanup. See “Creating a Software Master” in this guide for information about adding build steps to the software master definition.

### To execute a data cleanup procedure and build step

---

From Product Packaging (GH962)

1. Choose Data Cleanup Procedure Application (P9646).

The Work With Procedure form appears.

2. Click Find.

Existing data cleanup procedures appear.

3. To execute the entire procedure, complete one of the following:
  - On the Work With Procedures form, choose the procedure and from the Row menu, choose Execute.
  - On the Procedure Revision form, choose the main data cleanup procedure and from the View menu, choose Execute Step.
  - On the Procedure Revisions form and from the Form menu, choose Run Main Procedure.

These options execute the entire procedure. If it executes successfully, OneWorld updates the procedure and the build steps' statuses to 60 (Built).

4. To execute individual build steps, on the Work With Procedures form, choose a procedure, and then click Select.

The Procedure Revision form appears.

5. Click on the Steps tab.
6. Choose the build step you want to execute and from the View menu, choose Execute Step to process only that step.

This option executes the step. If the step executes successfully, OneWorld updates the step status to 60 (Built). If you execute a data cleanup procedure step, all of its subordinate steps also execute, including any subprocedures.

## Deleting a Data Cleanup Procedure

This task explains how to delete a data cleanup procedure. The application deletes everything associated with the procedure, including deleting where it existed as a subprocedure under other data cleanup procedures.

### **To delete a data cleanup procedure**

---

From Product Packaging (GH962)

1. Choose Data Cleanup Procedure Application (P9646).

The Work With Procedure form appears.

2. Click Find.

Existing data cleanup procedures appear.

3. Choose the procedure you want to delete, then click Delete.

A message box appears to verify that you want to delete the procedure.

4. Click OK.

# Working with Data Cleanup Templates

---

This chapter explains how to work with data cleanup templates. Templates are optional for data cleanup. You can use them as a guide to building your own data cleanup procedures. You can add, revise, copy, and delete templates just as you would procedures.

## See Also

- “Working with Data Cleanup Procedures” for information about procedures.

This chapter contains the following:

- Adding a data cleanup template
- Revising a data cleanup template
- Copying a data cleanup template
- Deleting a data cleanup template

## Adding a Data Cleanup Template

This task explains how to add a data cleanup template. This allows you to add template information and the steps you want for cleaning up data.

### **To add a data cleanup template**

---

From Advanced Operations (GH9622)

1. Choose Data Cleanup Procedure Templates (P9646).

The Work With Templates form appears.

2. Click Add.

The Template Definition Director form appears.

3. Click Next.

The Add Template form appears.

4. Complete the following fields:

- Name

Enter a unique name for the data cleanup template you are adding, such as Object Librarian Cleanup.

- Description

Optional. Enter a description of the data cleanup template.

- Comment

Optional. Enter a comment about the data cleanup template.

- Status

Enter a status of 10 (In Definition).

- Template Release

Enter the OneWorld release that pertains to the data cleanup template, such as B7333. This is the release currently installed on the client workstation you are using.

- Template Name

Enter the name of another data cleanup template that you want to work from for the template you are adding.

5. Click Next.

The Template Revision form appears.

6. To refresh the information shown on either tab on the Template Revision form, from the View menu, choose Refresh.

7. Click the Information tab, and complete the following fields:

- Description
- Status
- Comment
- Release

8. Click the Steps tab.
9. To add a step, click Add.

OneWorld creates a new step labelled Undefined and places it at the end of the highlighted procedure displayed in the tree. If you add a step to a subprocedure, the Undefined step appears at the end of that subprocedure. A form appears asking you to verify your add to the subprocedure; the add changes that procedure, whether it is a subprocedure or not, wherever it appears.

10. Choose the Undefined step and complete the following fields:
  - Status  

Enter the status of the data cleanup build step. Set the status to 10 (In Definition).
  - Type  

Enter which type of build step you want to add. For example, you can add steps that are batch processes, a Microsoft Windows executable program, or another data cleanup procedure or template.
  - Sequence  

OneWorld automatically displays a sequence number based upon where you add the build step. You can change this sequence number to move the build step up or down in the template. You can use the same sequence number for multiple build steps.
  - Description  

Optional. Enter a description of the data cleanup build step.
  - Comment  

Optional. Enter a comment about the data cleanup build step.
11. Complete the following fields, which appear if you are adding a batch process:
  - Object Name  

Enter a OneWorld name of a batch process. For example, enter R9647R to have the step execute the Object Librarian - Object Relationship Cleanup report.
  - Version  

Enter the name of the report version for the batch process you entered. For example, XJDE0002, the Final Mode version.

12. Complete the following fields, which appear if you are adding a Microsoft Windows executable program:
  - Executable  

Enter the name of a Microsoft Windows-compatible executable program. For example, the `xcopy.exe` program.
  - Parameters  

Enter the parameters for the executable program. You can enter command-line switches, directories, and files. For example, for the `xcopy.exe` program, you could enter a copy-from parameter of `c:\temp\*.*`, and you can enter a command-line switch of `\s`.
13. Complete the following field, which appears if you are adding a data cleanup procedure:
  - Procedure Name  

Enter the name of an existing data cleanup procedure. The data cleanup procedure that you enter must already exist. If the procedure appears on the visual-assist form, then it exists.
14. When you tab out of the last field or click on a step in the tree structure on the left side of the form, the data cleanup application saves your step.
15. When finished adding steps, click OK.

## Revising a Data Cleanup Template

This task explains how to revise a data cleanup template that you already created. This allows you to revise the information and steps you created within the template.

If you need to add a new data cleanup template, see “Adding a Data Cleanup Template” in this chapter.

### **To revise a data cleanup template**

---

From Advanced Operations (GH9622)

1. Choose Data Cleanup Procedure Templates (P9646).

The Work With Templates form appears.

2. Click Find.

Existing data cleanup templates appear.



3. Choose the template you want to revise, then click Select.

The Template Revisions form appears.

4. To refresh the information shown on either tab on the Template Revisions form, from the View menu, choose Refresh.
5. Click the Information tab, and revise the following fields:

- Description

Optional. Enter a description of the data cleanup template.

- Status

Verify the status of the data cleanup template. Set the status to 10 (In Definition) before you execute the template.

- Comment

Optional. Enter a comment about the data cleanup template.

- Release

Verify the OneWorld release that pertains to the data cleanup template, such as B7333. This is the release currently installed on the client workstation you are using.

6. Click the Steps tab.
7. To revise a build step, choose the step, and complete the following fields:

- Status

Verify the status of the data cleanup build step. Set the status to 30 (validate) before you execute the step.

- Type

Verify which component type you want the build step to perform. For example, you can define steps as a batch process, a Microsoft Windows executable, or another data cleanup template.

- Sequence

Verify the order in which you want the step performed during the execution of the procedure. You can use the same sequence number for multiple build steps.

- Description

Optional. Enter a description of the data cleanup build step.

- Comment

Optional. Enter a comment about the data cleanup build step.

The following fields appear if you are revising a batch process:

- Object Name

Verify the OneWorld name of the batch process.

- Version

Verify the name of the report version for the batch process.

The following fields appear if you are revising a Microsoft Windows executable program:

- Executable

Verify the Microsoft Windows executable program.

- Parameters

Verify the parameters of the executable.

The following field appears if you are revising a data cleanup procedure:

- Procedure Name

Verify the name of the data cleanup procedure.

8. To renumber your build steps, from the View menu, click Renumber.

OneWorld renumbers your build steps, updating the Sequence field. The renumbering begins with 10 and increments each step by 10 (10, 20, 30, etc.). OneWorld keeps the build steps in the same sequence that you established.

9. To delete a step, choose the step, and then click Delete.
10. When finished revising the data cleanup template, click OK.

## Copying a Data Cleanup Template

This task explains how to copy a data cleanup template that you already created.

### **To copy a data cleanup template**

---

From Advanced Operations (GH9622)

1. Choose Data Cleanup Procedure Templates (P9646).

The Work With Template form appears.

2. Click Find.

Existing data cleanup templates appear.

3. Choose the template you want to copy, then click Copy.

The Copy Template form appears.

4. Complete the following fields, then click OK:

- Copy From Template

Enter the name of an existing data cleanup template.

- Copy To Template

Enter the name for the new data cleanup template.

## Deleting a Data Cleanup Template

This task explains how to delete a data cleanup procedure. The application deletes everything associated with the procedure, including deleting it where it existed as a subprocedure under other data cleanup procedures.

### **To delete a data cleanup template**

---

From Advanced Operations (GH9622)

1. Choose Data Cleanup Procedure Templates (P9646).

The Work With Template form appears.

2. Click Find.

Existing data cleanup templates appear.

3. Choose the master you want to delete, then click Delete.

A message box appears to verify that you want to delete the template.

4. Click OK.

# Appendices





## Appendix A - Relative and Absolute Paths

An absolute path name begins with the root directory and details the entire path, such as `Z:\DepServer\CD Templates\Partial Update`. A relative path name begins with a token, such as `$DEP`. The token represents at least the root directory and possible contiguous directory paths. For example, `$DEP` may represent `Z:\DepServer`. This relative path might appear within an applicable field as `$DEP\CD Templates\Partial Update`. The following list shows the relative path tokens you can use:

- `$DEP` - Represents the root directory where OneWorld is installed on your deployment server.
- `$IMAGE` - Represents the image path of your software master.
- `$CLIENT` - Represents the root directory where OneWorld is installed on the workstation you are currently using.
- `$TEMP` - Represents the Microsoft Windows temporary directory of the workstation you are currently using.

You do not need to define the tokens. The Product Packaging Tool automatically resolves the token values based upon your OneWorld installation and other information available to it.

This chapter contains the following:

- Using relative path tokens
- Converting relative and absolute paths

### Using Relative Path Tokens

This task explains how to use a relative path token within a field.

#### **To use relative path tokens**

---

From a field that allows relative paths, enter a token followed by the remainder of the path needed for that field. For example:

```
$DEP\CD Templates\Partial Update
```



## Converting Relative and Absolute Paths

OneWorld allows you to convert relative path tokens and absolute path names for those fields that accept them within the Product Packaging Tools. If any fields use a token, you can convert the token into an absolute path or you can convert the absolute path into a token. For example, you can convert any instance of \$DEP into Z:\DepServer or you can convert Z:\DepServer into \$DEP.

### ► **To convert relative and absolute paths**

---

From Product Packaging (GH962)

- 1 Choose OneWorld Product Packaging (P9640).

The Work With Software Master form appears.

- 2 Click Find.

Existing software masters appear.

- 3 Choose the master for which you want to convert relative and absolute paths, and from the Row menu, choose Convert Paths.

The Work With Batch Versions - Available Versions form appears.

- 4 Click Find.

Available versions appear.

- 5 Choose a report version, and then from the Row menu, choose Processing Options.

The Processing Options form appears.

- 6 Complete the following fields, and then click OK:

- FromPath

Enter the root path or relative path token that you want to change. For example, enter Z:\DepServer.

- CD ToPath

Enter the root path or relative path token that you want the FromPath to become. For example, enter \$DEP.



- 
- 7 On the Work With Batch Versions - Available Versions form, choose the version and click Select.

The Version Prompting form appears.

- 8 Click Submit.

The Report Output Destination form appears.

- 9 Choose to send the report to a printer or to your screen, then click OK.

OneWorld converts the absolute and relative paths for any fields within the Product Packaging Tools to which this applies.



# Glossary



# 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.

**change tables.** Tables that store control table changes (adds, changes, and deletes) from previous releases of OneWorld to the current release. The OneWorld installation process uses change tables to update an enterprise's control tables.

**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 tables.** Tables that organize and sequence system data. Users can add, change, or delete this data (see the *change tables* glossary item) which is used by J.D. Edwards systems. Examples of control tables include data dictionary, menu, user defined code, user override, workflow, central object, object librarian, and versions list.

**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 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 receive all or parts of the data and procedure definitions

from a parent class. Inheritance enhances development 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 first 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 its 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 application 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 delete 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, workflow 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



# Index

---

## Symbols

\$CLIENT, B-1  
\$DEP, B-1  
\$IMAGE, B-1  
\$TEMP, B-1

## A

Absolute and relative paths, B-1  
Absolute paths, B-1  
    converting to relative path, B-2  
Adding a build step, 3-17, 6-7, 6-17  
Adding a change table configuration, 2-17  
Adding a data-cleanup procedure, 6-3, 6-15  
Adding a repair to a software master, 3-3  
Adding a software master definition, 2-10  
Adding a software master template, 4-12  
Adding a subdirectory, 3-20  
Adding ODBC data sources, 2-5  
Attach to a build step, 3-18

## B

Batch Application Selection, 2-19  
Batch process (build step process), 4-8  
Breaking News web site, 1-4  
Build change tables (build step process), 4-6  
Build cross-reference (build step process), 4-5  
Build data dictionary (build step process), 4-5  
Build in-house package database (build step process), 4-5  
Build package database (build step process), 4-8

Build planner database (build step process), 4-5  
Build pristine database (build step process), 4-5  
Build standalone database (build step process), 4-6  
Build step processes, 4-3  
    batch process, 4-8  
    build change tables, 4-6  
    build cross-reference, 4-5  
    build data dictionary, 4-5  
    build in-house package database, 4-5  
    build package database, 4-8  
    build planner database, 4-5  
    build pristine database, 4-5  
    build standalone database, 4-6  
    build store & forward database, 4-6  
    compress, 4-6  
    copy CD directory files, 4-7  
    create CD file structure, 4-4  
    create INF, 4-7  
    data cleanup, 4-8  
    manual build step, 4-4  
    master central objects, 4-6  
    package build, 4-4  
    pre-build package, 4-4  
    pre-build pristine, 4-4  
    undefined, 4-3  
    virus check, 4-7  
    Windows executable, 4-7  
Build steps  
    adding a step, 3-17, 6-7, 6-17  
    adding an attachment, 3-18  
    Build Steps tab on CD Revisions form, 3-15, 6-7, 6-10, 6-17, 6-19  
    deleting, 3-18, 6-11, 6-20  
    Execute Step command, 3-18, 6-13  
    list of step icons, 3-16  
    renumbering steps, 3-18, 6-11, 6-20  
    revising a step, 3-16, 6-10, 6-19  
    revising templates, 4-14  
    Run From Step command, 3-18

Build store & forward database (build step process), 4–6

### C

CD Configuration Director, 2–11, 6–4, 6–15

CD Information, 2–11, 6–4, 6–16

CD Information (templates), 4–14

CD Information tab, 3–5, 3–15, 6–6, 6–9, 6–19

CD Packages, 2–13, 6–16

CD Revisions, 3–14, 6–19

Change table configuration, 5–1

Change Table Configuration Director, 1–3

Change table configurations

adding, 2–17

copying, 5–7

deleting, 5–9

revising/submitting change tables, 5–3

setting up, 2–17

Change Table Director, 2–14

Change Table Revisions, 5–4

Change tables, configuring, 2–17

Cleaning software master data, 6–3

Compress (build step process), 4–6

Configuring change tables, 2–17

Conventions for this guide, 1–6

fonts, 1–7

version numbering, 1–6

Converting relative and absolute paths, B–2

Copy CD Configuration, 3–23, 6–11, 6–21

Copy CD directory files (build step process), 4–7

Copy Change Table Definitions, 5–7

Copying a change table configuration, 5–7

Copying a data-cleanup procedure, 6–11, 6–21

Copying a software master definition, 3–23

Copying a software master template, 4–18

Create CD file structure (build step process), 4–4

Create INF (build step process), 4–7

Creating a data dictionary in the pristine data source, 2–7

Creating a software master, 2–10

Creating the final software master, 2–25

Custom CD, 4–11

Customer support, Breaking News web site, 1–4

### D

Data cleanup, 6–3

Data cleanup (build step process), 4–8

Data sources

adding ODBC data sources, 2–5

creating a data dictionary in the pristine data source, 2–7

Data-cleanup build steps, executing, 6–12

Data-cleanup procedures

adding, 6–3, 6–15

copying, 6–11, 6–21

deleting, 6–13, 6–21

executing, 6–12

revising, 6–9, 6–18

Data-cleanup templates, 6–15

Defining a software master, 2–10

Definitions, product packaging, 1–1

Delete a directory, 3–21

Delete build steps, 3–18, 6–11, 6–20

Deleting a change table configuration, 5–9

Deleting a data-cleanup procedure, 6–13, 6–21

Deleting a repair to a software master, 3–10

Deleting a software master template, 4–20

Deleting a software master, 3–11

Deleting a software master definition, 3–27

Details of all build step processes, 4–3

Directory structures, 3–19

adding a directory, 3–20

deleting a directory, 3–21

loading a directory, 3–20

revising a directory, 3–19

Distributed development, 1–3

### E

Execute Step command, 3–18, 6–13

Executing a data-cleanup procedure, 6–12

Executing data-cleanup build steps, 6–12

**F**

Fonts, conventions for this guide, 1-7

## Forms

- Batch Application Selection, 2-19
- CD Configuration Director, 2-11, 6-4, 6-15
- CD Information, 2-11, 6-4, 6-16
- CD Information tab (templates), 4-14
- CD Packages, 2-13, 6-16
- CD Revisions, 3-14, 6-19
  - Build Steps tab, 3-15, 6-7, 6-10, 6-17, 6-19
  - CD Information tab, 3-15, 6-6, 6-9, 6-19
  - Directory Structure tab, 3-19
- CD Revisions (templates), 4-13
- Change Table Director, 2-14
- Change Table Revisions, 5-4
- Copy CD Configuration, 3-23, 6-11, 6-21
- Copy CD Configuration (templates), 4-18
- Copy Change Table Definitions, 5-7
- Master Data Locations, 3-5
  - CD Information tab, 3-5
  - Package Information tab, 3-6
- Master Repair Director, 3-4
- OneWorld Data Selection, 3-7
- OneWorld Object Selection, 3-6
- Procedure Revision, 6-6, 6-9
- Remaining Steps, 3-9
- Select Package Build, 2-13
- Source Environment Selection, 2-18
- Target Release Selection, 2-17
- Work With Batch Versions - Available Versions, B-2
- Work With CD Templates, 4-12
- Work With Change Table Definitions, 5-3
- Work with Master Repair Objects, 3-4
- Work With Software Mastering, 3-13

**I**

Icons (for build steps), 3-16

Internet address, Breaking News, 1-4

**L**

Loading a subdirectory, 3-20

**M**

- Manual build step (build step process), 4-4
- Master central objects (build step process), 4-6
- Master Data Locations, 3-5
- Master Repair Director, 3-4
- Minimum technical requirements, 1-4

**O**

- ODBC data sources, adding, 2-5
- OneWorld Data Selection, 3-7
- OneWorld Object Selection, 3-6

**P**

- Package build (build step process), 4-4
- Package Information tab, 3-6
- Pre-build package (build step process), 4-4
- Pre-build pristine (build step process), 4-4
- Prerequisites for product packaging, 1-3
- Pristine data source, creating a data dictionary, 2-7
- Procedure Revision, 6-6, 6-9
- Process flow graphic, product packaging, 1-2
- Product packaging
  - definitions, 1-1
  - prerequisites, 1-3
  - process flow graphic, 1-2
- Product Packaging Tools, 1-1
- Proofing a software master definition, 3-25, 4-19

### R

- Refresh the CD Revisions form, 3-14, 4-14, 6-6, 6-9, 6-19
- Relative and absolute paths, B-1
- Relative paths, B-1
  - converting to absolute path, B-2
  - using, B-1
- Remaining Steps, 3-9
- Renumber build steps, 3-18, 6-11, 6-20
- Repairing a software master, 3-3
- Requirements (minimum technical), 1-4
- Revising a build step, 3-16, 6-10, 6-19
- Revising a data-cleanup procedure, 6-9, 6-18
- Revising a directory, 3-19
- Revising a software master definition, 3-13
- Revising a software master template, 4-13
- Revising change tables, 5-3
- Run From Step command, 3-18

### S

- Select Package Build, 2-13
- Setting up a software master definition, 2-10
- Setting up change table configurations, 2-17
- Setting up software master templates, 4-12
- Software master, 3-1
  - adding a repair, 3-3
  - cleaning up data, 6-3
  - creating, 2-10
  - creating the final, 2-25
  - defining, 2-10
  - deleting, 3-11
  - deleting a repair, 3-10
  - repairing, 3-3
  - updating OneWorld, 2-27
- Software master checklist, creating, 2-3
- Software master definition, 3-13
  - adding, 2-10
  - copying, 3-23
  - deleting, 3-27
  - proofing, 3-25, 4-19
  - revising, 3-13
  - setting up, 2-10
  - validating, 2-24, 3-26

- Software master templates, 4-1
  - adding, 4-12
  - build step processes, 4-3
  - copying, 4-18
  - custom CD, 4-11
  - deleting, 4-20
  - revising, 4-13
  - setting up, 4-12
  - update CD template, 4-8
- Software Mastering Director, 1-2
- Source Environment Selection, 2-18
- Submitting change tables, 5-3

### T

- Target Release Selection, 2-17
- Telephone numbers, Worldwide Customer Support, 1-5

### U

- Undefined (build step process), 4-3
- Update CD template, 4-8
- Using relative-path tokens, B-1
- Using the software master to update OneWorld, 2-27

### V

- Validate the CD Revisions form, 3-15, 4-14
- Validating a software master definition, 2-24, 3-26
- Version numbering, conventions for this guide, 1-6
- Virus check (build step process), 4-7

### W

- Web site, Breaking News, 1-4
- Welcome, 1-1
- Windows executable (build step process), 4-7
- Work With Batch Versions - Available Versions form, B-2

- Work With CD Templates, 4-12
- Work With Change Table Definitions, 5-3
- Work with Master Repair Objects, 3-4
- Work With Software Mastering, 3-13
- Worldwide Customer Support, 1-4
  - telephone numbers, 1-5

