

PeopleSoft®

EnterpriseOne B73.3.1
ActivEra Implementation Activators
PeopleBook

June 1999

J.D. Edwards World Source Company
One 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, 1997 - 1999

All Rights Reserved

SKU B7331CEACA

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

ActivEra Implementation Activators Overview	1-1
Benefits of Implementation Activators	1-1
Who Should Use the ActivEra Implementation Activators Guide	1-2
Customer Rough Cut	2-1
Creating a Rough Cut Customer Configuration	2-3
Implementation Activators	3-1
Using Implementation Activators	3-3
Starting the Implementation Activators Application	3-3
Accessing CBT for an Implementation Activator Process	3-7
Adding an Attachment to an Implementation Activator Process	3-7
Adding Additional Information to an Implementation Activator Process	3-9
Reviewing a Reason for an Implementation Activator Process	3-10
Implementation Activator Process Setup	4-1
Naming Conventions Used in Implementation Activator Processes	4-1
Data Items	4-2
Activity Names	4-2
Prototyping Implementation Activator Processes	4-3
Creating an Implementation Activator Process	4-5
Adding Activities to an Implementation Activator Process	4-9
Adding an Interactive Application Activity	4-9
Adding a Process Activity	4-19
Defining the Flow of Implementation Activator Processes	4-23
Activity Conditions	4-23
Understanding the Flow of Activities	4-25
Special Cases	4-25
Example: Process Flow	4-26
Example: Flow of Activities	4-27
Modifying Implementation Activator Processes	4-33
Activating or Deactivating an Implementation Activator Process	4-33
Adding Supplemental Information to an Implementation Activator Process	4-34
Creating Dependencies for Implementation Activator Processes	4-37
Modifying Text within an Activator Question Form	4-41
Appendix	
Appendix A - Example of an Implementation Activator Process Prototype ..	A-1
Example of an Implementation Activator Process Prototype	A-3

Glossary



ActivEra Implementation Activators Overview

ActivEra™ is a broad product and technology suite that enables companies to easily adapt their ERP software after implementation to meet changing business conditions. ActivEra gives businesses control over their application and technical infrastructure to deliver quick adaptation of the ERP environment to meet ever-changing needs. ActivEra comprises three components. The ActivEra console provides a “window” into adaptation or change; ActivEra Business and Technology Activators are discrete applications or utilities that shape the system to the business or technology need; and the ActivEra Extension Architecture allows companies to create and “snap in” third-party business processes or technology activators.

ActivEra Implementation Activators automate and simplify OneWorld system configuration from early demonstrations through implementation. Implementation activators step you through system configuration by asking a series of questions about the enterprise, such as which business processes or OneWorld application suites will be used, the industries the enterprise deals with, and the geographic regions in which the enterprise does business. As you answer the questions, ActivEra assists you in tailoring the configuration to fit those requirements. Then, using Implementation Activators, the system follows the flow of the now-tailored set of specifications and leads you through the actual configuration of OneWorld, which includes further defining requirements by process or application suite, such as adding companies, business units, and user defined codes.

The Implementation Activator application, like most OneWorld applications, can be tailored to fit the requirements of a particular prospect or client. You use the applications within Activator Setup to modify an activator. For example, a presales consultant might define an activator for the prospect in a demonstration environment. Later, the postsales consultant might modify that activator to include more details from newly gained information.

Benefits of Implementation Activators

Implementation Activators offer you:

- A streamlined bridge between the preliminary prospect findings and the actual implementation
- A visual audit trail that allows you to track the status of in-process or completed activities during implementation configuration
- A directed flow of implementation setup activities that streamline configuration and provide a checklist of what needs to be done



- Change management directors to assist you in performing infrequent activities, such as incorporating a newly acquired company or new distribution facility into your existing configuration

Who Should Use the ActivEra Implementation Activators Guide

The *ActivEra Implementation Activators Guide* is designed for use by presales and postsales personnel, system administrators, implementors, business partners, customers, and other participants in the implementation process.

Before You Begin

- You should have a working knowledge of common OneWorld concepts, which can be found in the *OneWorld Foundation Guide*. You should also have some knowledge of OneWorld tools, which can be found in the *OneWorld Development Tools Guide*.



Customer Rough Cut

You use the Customer Rough Cut application to limit the scope of the Implementation Activators process. In Rough Cut, you choose the business processes or application suites, geographic regions, and industries to include in the OneWorld configuration.

After you have made your choices in Rough Cut and generate the data, the system populates the Rough Cut Selections table (F98882) with applicable and non-applicable specifications. When you invoke Implementation Activators, only the applicable specifications you defined during the rough cut process will be available, making it easier to configure your system.

The following topic is described:

- Creating a Rough Cut Customer Configuration



Creating a Rough Cut Customer Configuration

When you define your application suites using the Customer Rough Cut application, you can choose from business processes, such as Order to Cash, Design to Market, Procure to Pay, or Production; or you can choose individual application suites that you want to include in the configuration.

If you choose a business process, the system chooses all application suites associated with that business process. For example, if you choose Design to Market, the system automatically selects Foundation, Financials, AEC, Logistics/Distribution, and Manufacturing for you, because they are all required to support Design to Market. Similarly, if you choose a particular application suite, such as Manufacturing, the system chooses all other application suites that are required to use the Manufacturing suite, such as the Financials, AEC, Logistics/Distribution, and Foundation suites.

You further define your rough cut inclusions by industry and by geographic regions.

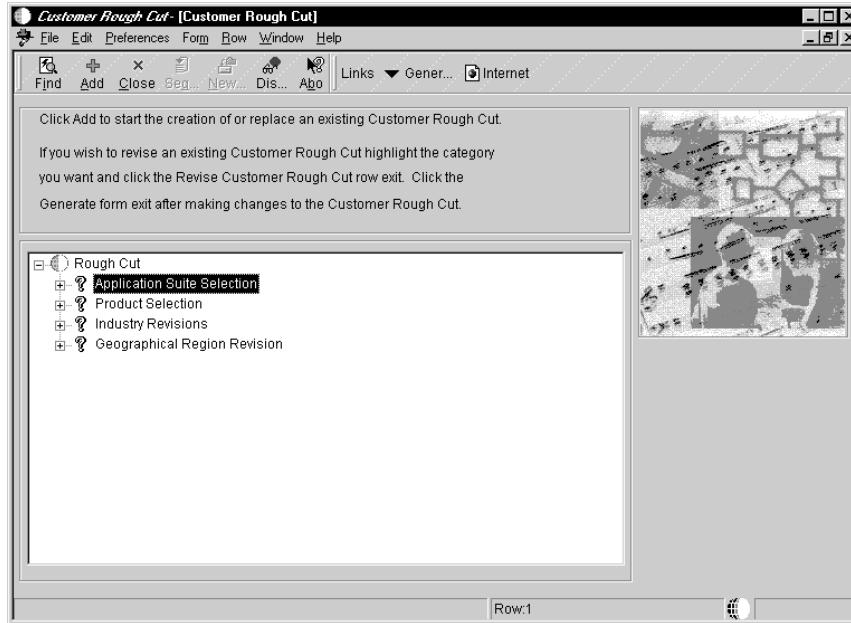
After you generate a rough cut you can revise it, if necessary. When you revise a rough cut, you can also save your revisions at any point in the process by clicking Save.

Complete the following tasks:

- Create a rough cut customer configuration
- Revise a rough cut customer configuration

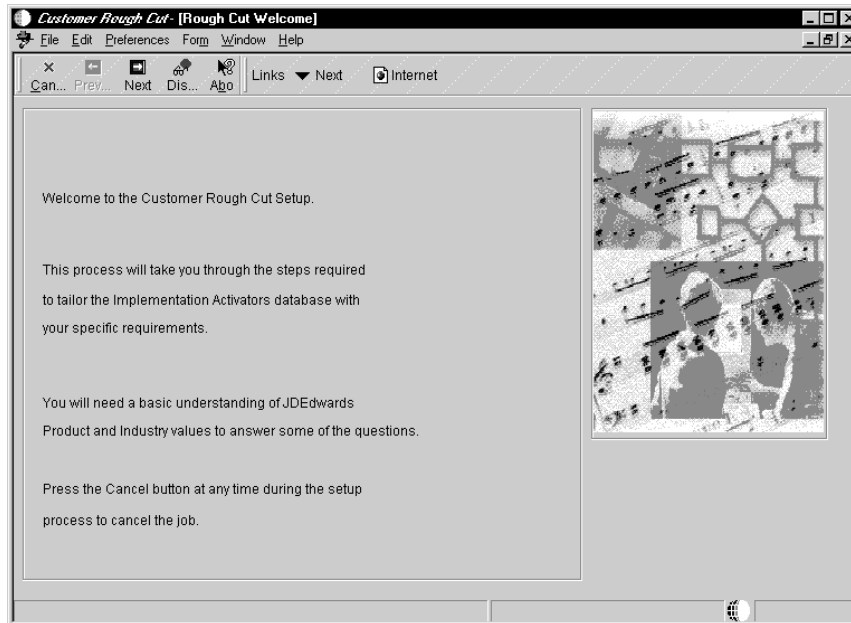
► **To create a rough cut customer configuration**

From the ActivEra menu (G021), choose Customer Rough Cut.

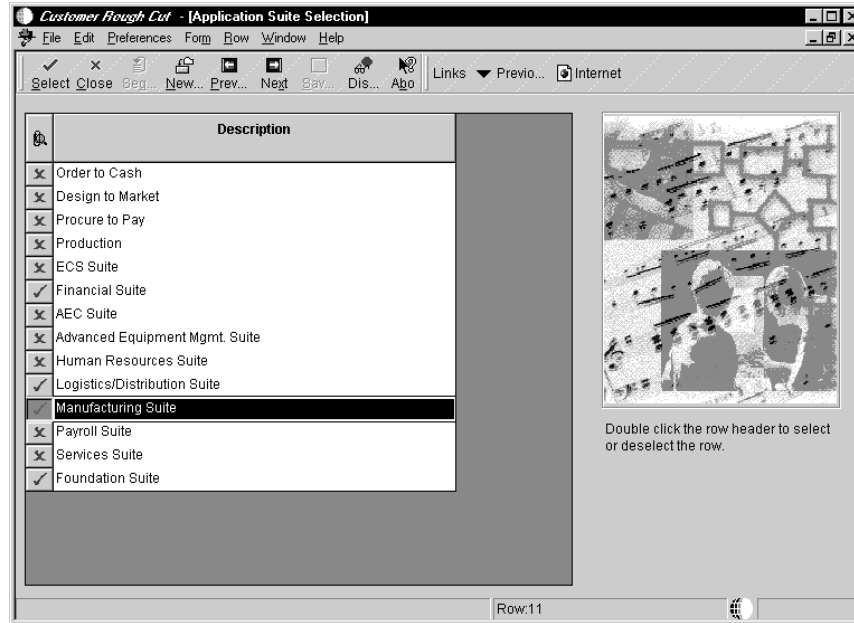


1. Click Add.

The system displays the Welcome form of the Director.



2. Click Next.



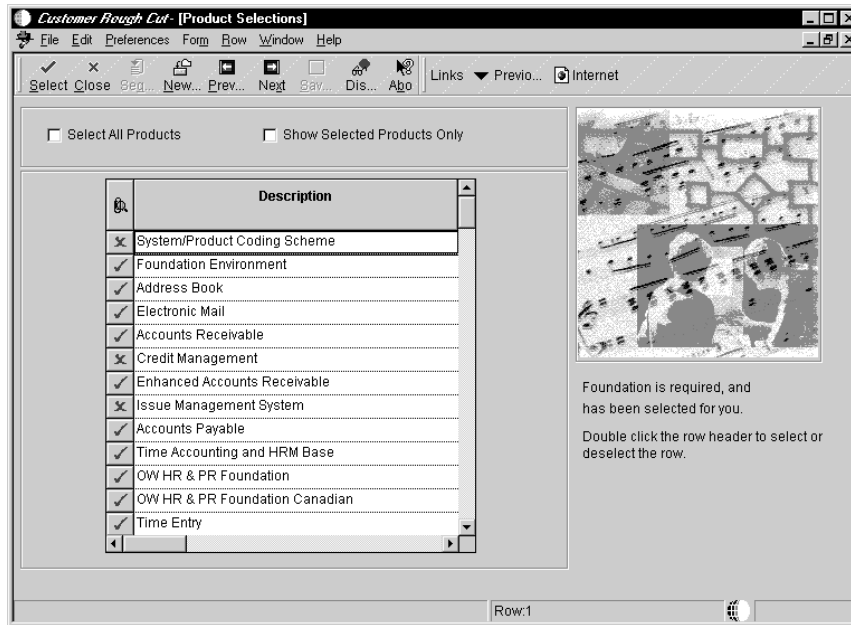
3. On Application Suite Selection, choose the business process or application suite that you want to include in the implementation and click Select.

A check mark appears next to each selection you make. To deselect an application suite, choose the item and click Select again. An X appears next to the application suite that you do not want to include in the configuration.

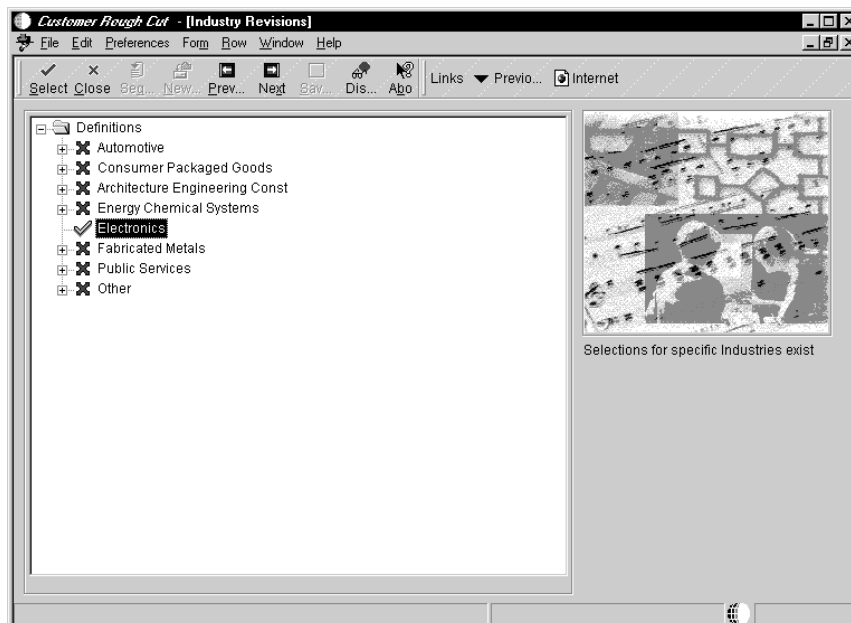
Alternatively, you can double-click each choice to select or deselect it.

Note that when you make one selection, the system automatically makes additional selections on which your selection depends. For example, the system automatically selects the Foundations suite for you. You cannot deselect the Foundation suite because it is required.

4. When you have finished selecting application suites, click Next.



5. On Product Selections, all products that are associated with the application suites or business products that you selected appear with check marks next to them. You can select additional products or deselect products that you do not want to include in the implementation by choosing them and clicking Select.
6. If you want to select all products, regardless of whether they are associated with the application suites you chose, click the “Select All Products” option.
7. To show only the products that are associated with the application suites you chose, click the “Show Selected Products Only” option.
8. Click Next.

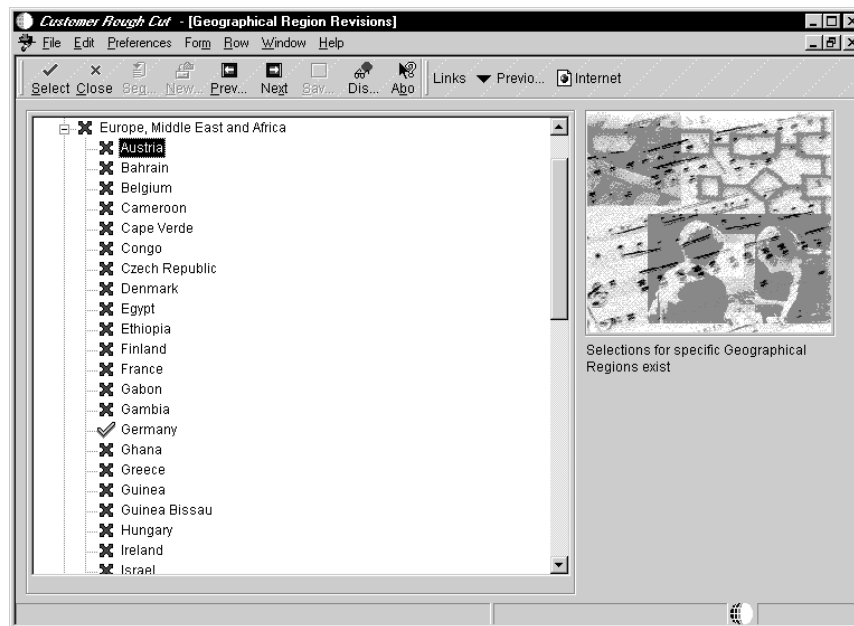


9. On the Industry Revisions form, click the plus sign next to the Definitions folder to view all industry types.
10. Choose the industry that applies to your business and click Select. You can choose more than one industry.

Alternatively, you can double-click the industry names to select them.

11. Click Next.

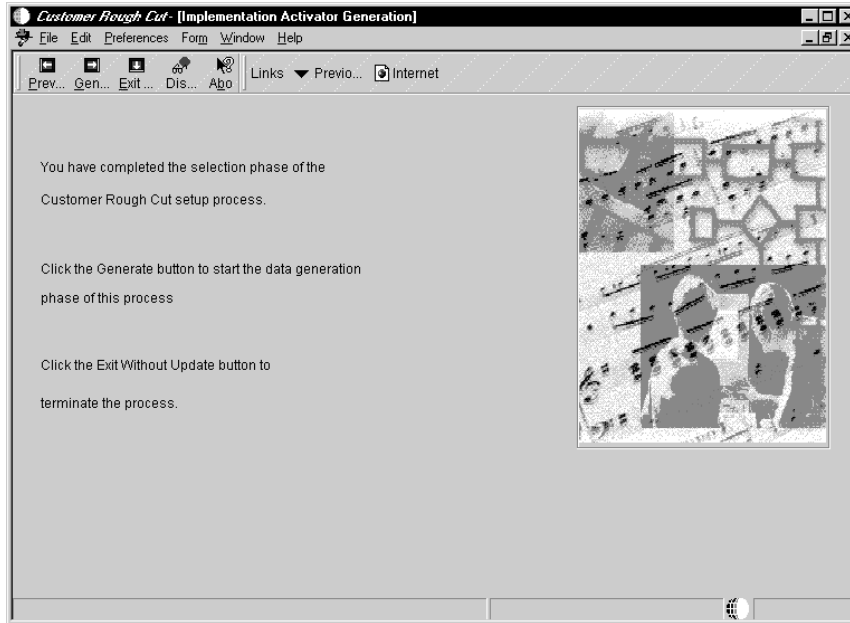
The system displays Geographical Region Revisions.



12. Click the plus sign next to the Definitions folder to view all geographic regions, then click the plus sign next to the region that you want to include to view all countries associated with that region.

For example, click the plus sign next to Europe, Middle East and Africa to view all counties associated with that region.

13. Select the geographical areas that apply to your configuration by double-clicking them. Your selections determine which international requirements to include in the selection for setup.
14. When you have finished choosing geographic regions, click Next.



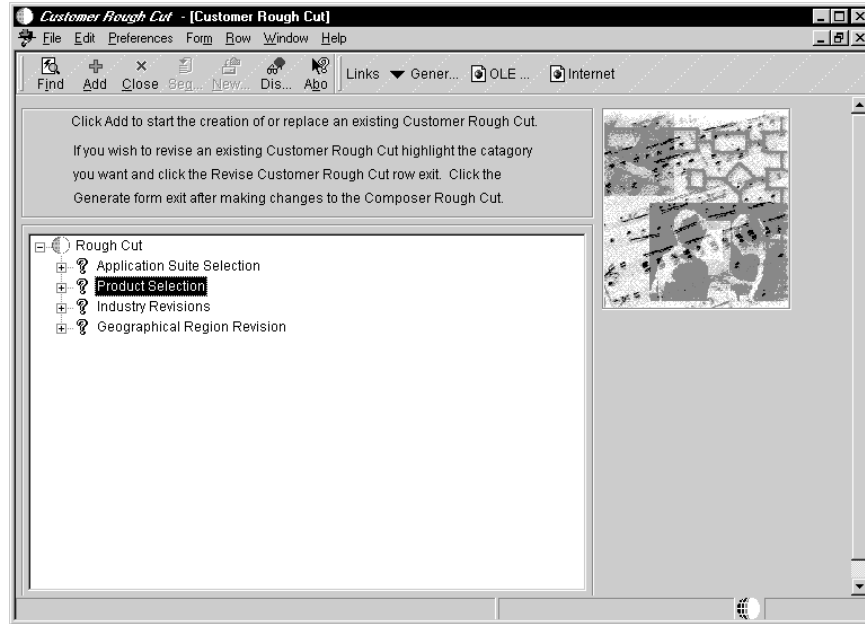
15. On Implementation Activator Generation, click the Generate button to create the Implementation Activator database. The generation process runs on your desktop and might take several minutes to complete.

When you run the generation process, the system marks all system codes, industries, and geographic regions that you chose as applicable for implementation. Therefore, only those items that are applicable to your business will appear in the next part of the process, Implementation Activators.

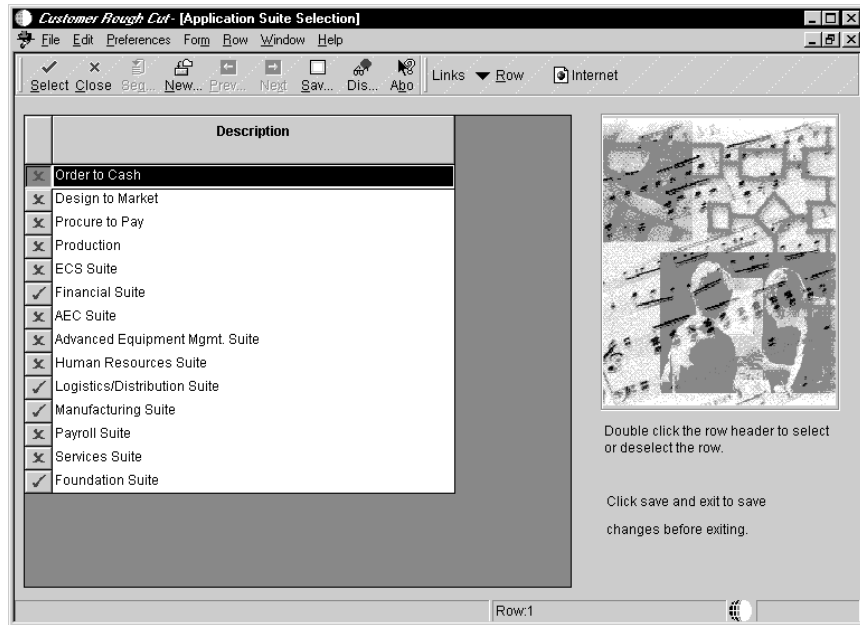
If you click Exit without generating the process, your choices will be lost.

► To revise a generated rough cut customer configuration

From Customer Rough Cut



1. Choose the category that you want to revise and choose Revise Category from the Row menu.



The system displays the revision form for the category you selected.

2. Revise the application suites, product selections, industry revisions, and geographical region revisions as necessary. Click Next after you have revised each category.

You can click Save at any point during the revision process to save the changes you made.

3. When you have finished revising the rough cut configuration, click the Generate button to update the database.



Implementation Activators

You use Implementation Activators to configure application suites or business processes after you specify the high-level requirements using the Customer Rough Cut application.

The following topic is described:

- Using Implementation Activators

Implementation Activators are divided into four areas:

- Implementation Approach, which consists of determining the approach you will take to implement the system. This area acts as a central repository for work papers and attachments for training plans; installation and environment setup documents; business process modeling workshop results, configuration information; information regarding going live; and information for refining the system after going live.
- Technology Environment, in which you define network environments, disk requirements, deployment strategies, language requirements, and security requirements.
- Organization Configuration, in which you configure foundation definitions, such as companies, addresses, cost centers, chart of accounts, inventory items, and storage locations.
- Business Process Configuration, in which you determine business processes by function, such as Financial, Distribution, or Manufacturing; and by process, such as Order Fulfillment, Design to Market, and Procure to Pay. You can also refine your business processes by defining menus for the organization, setting up user-defined address book functionality, and reclassifying accounts.

After you have completed all activities within an implementation activator, or when you click Exit within an implementation activator, the system displays a completion form. This form allows you to update the status icon for an activator, such as unopened, in progress, pending, or complete. The icons give you a visual cue of the activator's status.



Before You Begin

- Depending on your needs, you can activate two processing options prior to using the Implementation Activators application. The first processing option, Prompt Levels, allows you to show or hide the application level question and answer forms that the system displays while you use Implementation Activators. If you activate this processing option, the system displays one question and answer form that describes the implementation activator. If you answer Yes to this question and answer form, the system invokes all applications, one at a time, without invoking the other question and answer forms that normally appear.

The second processing option, Display Lower Level Activities, shows or hides the lowest level of activities within all implementation activators.

Using Implementation Activators

The Implementation Activators application simplifies the process of configuring your system by stepping you through a series of questions related to activities in your organization, your key business processes, your technology environment, and your project. Based on the information you enter, the Implementation Activators application assists you in configuring your OneWorld system.

When you use the Implementation Activators application to configure your system, you access processes and activities related to the application suites you specified during Customer Rough Cut. Within the activities you add the data, such as user defined codes, that pertain to that particular process.

Some of the data that the implementation activator processes need to complete a configuration relates to the OneWorld environment. The implementation activator processes derive this information from the environment you are signed onto during the configuration. For example, a prototype environment, a test environment, or a production environment.

The following topics are described:

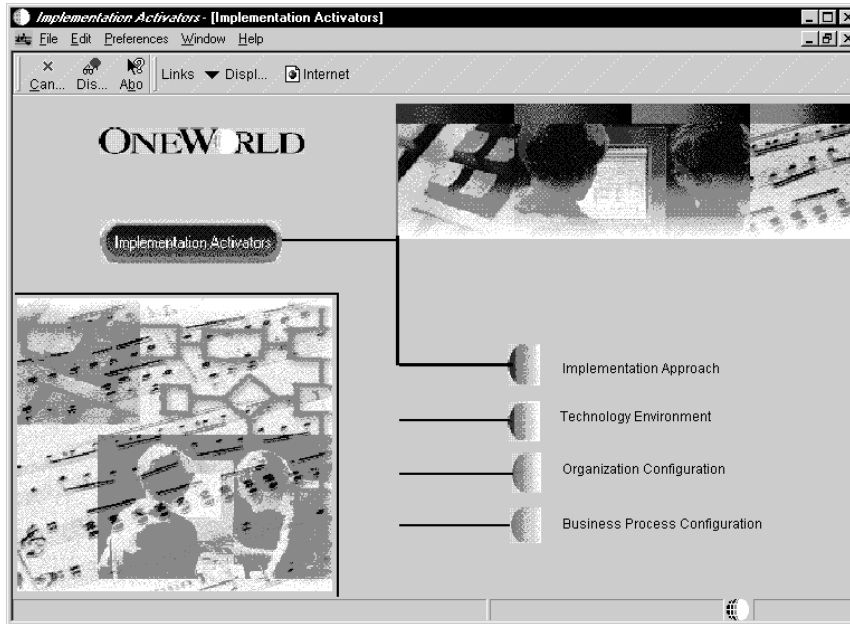
- Starting the Implementation Activators application
- Accessing CBT for an implementation activator process
- Adding an attachment to an implementation activator process
- Adding additional information to an implementation activator process
- Reviewing a reason for an implementation activator process

Starting the Implementation Activators Application

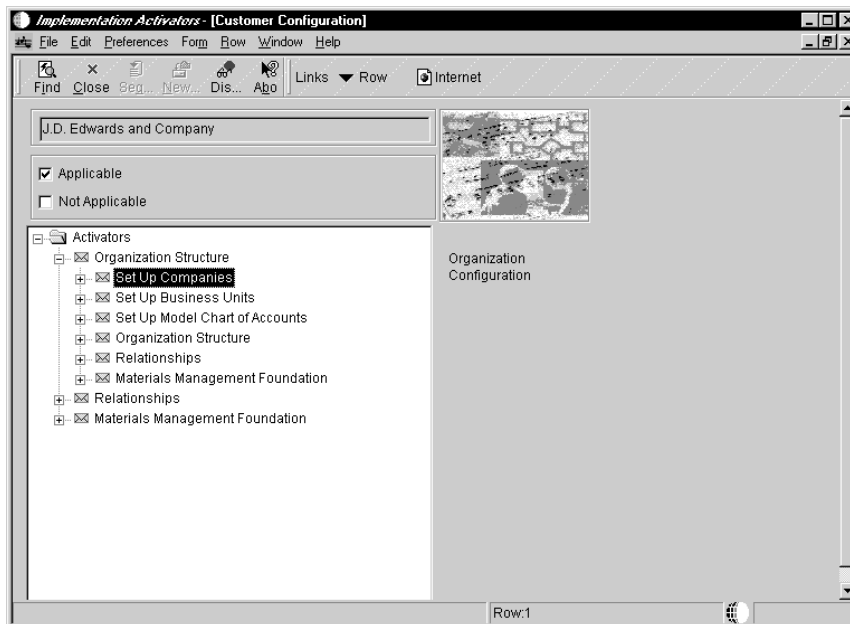
When you configure a process, the system launches programs within OneWorld in which you can add or modify data.

► **To start the Implementation Activators application**

From ActivEra (G021), choose Implementation Activators.



1. On the Implementation Activators form, click the topic that you want to work with. For example, click Organization Configuration.



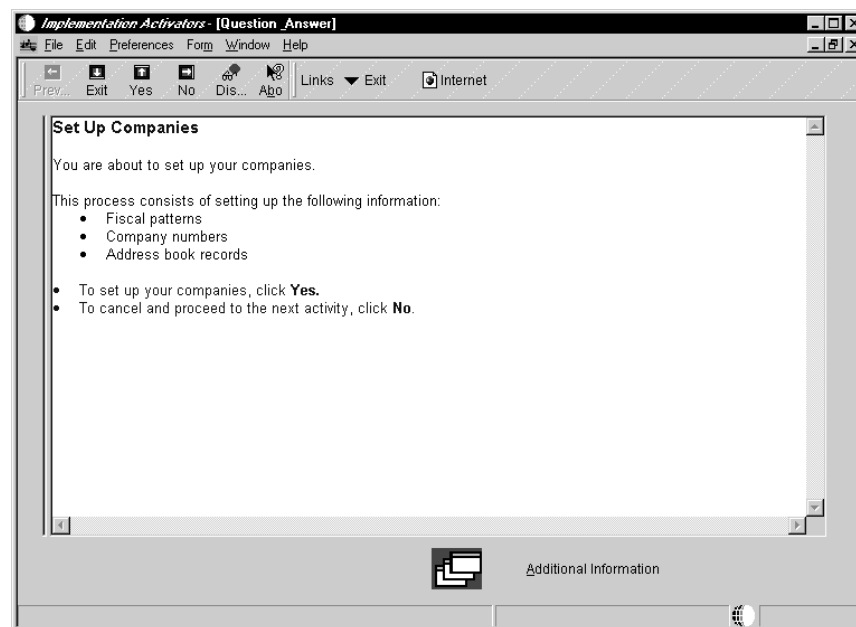
2. On Customer Configuration, click the plus sign next to the Activators folder to view all implementation activator processes or activities associated with the topic you chose.

The tree structure shows a hierarchy of implementation activator processes for the topic. Implementation activator processes are indicated by an envelope icon; the activities associated with them are indicated by a lightning bolt icon.

3. Choose the implementation activator process you want to work with, for example, “Set Up Companies,” and choose Start Config from the Row menu.

NOTE: You cannot choose an activity within an implementation activator process and start configuration on that activity. You must choose the *implementation activator process* in which the activity resides and then choose Start Config from the Row menu to invoke the activity.

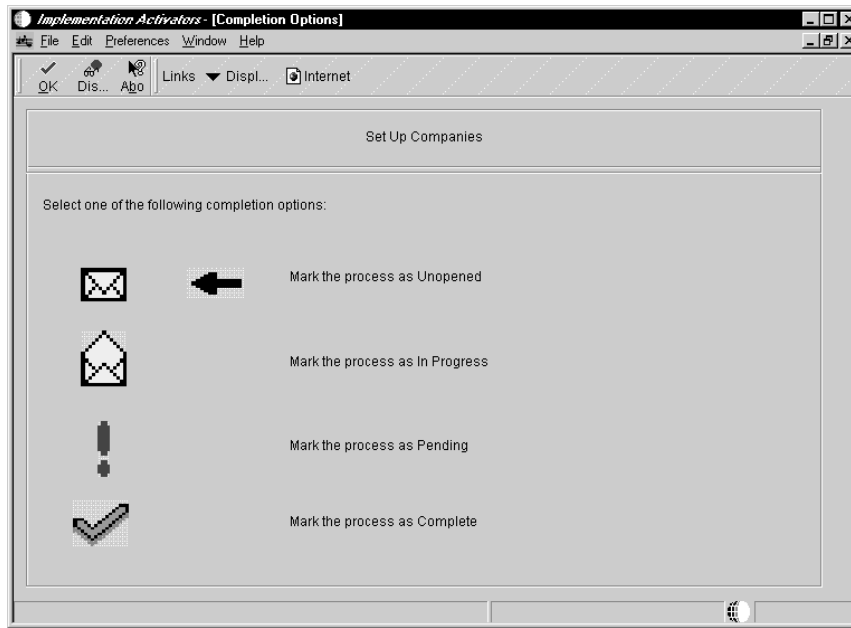
The system displays a Question and Answer form. This form summarizes the activities within the process you chose.



4. Do one of the following:
 - Click the Yes button to continue the process and set up the activity, or click the No button to skip this particular activity and move to the next activity.
 - Click Exit to end the implementation activator process.
 - Click Previous to return to the previous activity in the process.
5. When you click Yes, the system either invokes the particular application associated with the topic you chose, or displays another question and answer form and then displays an application, such as Work With Fiscal Date Patterns.

- When you are finished with an activity, click OK or click Close. The system invokes the next activity, or the next question and answer form.

If you click Exit on a question and answer form, the system displays the Completion Options form. The system will also display the Completion Options form if you are finished with the last activity in a process.



Choose one of the following options:

- Choose Mark the process as Unopened if you have not answered any of the questions within the implementation activator process.
- Choose Mark the process as In Progress if you have completed some, but not all of the questions within the implementation activator process. You can then go back at another time and complete the remainder of the questions within the process.
- Choose Mark the process as Pending to indicate that the activities within an implementation activator process are complete but need to be reviewed before they can be considered complete.
- Choose Mark the process as Complete if you have answered all of the questions within an implementation activator process, and the activities within the process have been reviewed.

If you have marked an implementation activator process as complete, it will not stop you from making changes to that particular process later.

When you choose a completion option, it might not immediately appear in the tree structure within the Implementation Activators application. It will appear the next time that particular set of processes is displayed or when you click the Find button.

7. Repeat Steps 1 through 6 until you are finished configuring the implementation activator process you are working with.

Accessing CBT for an Implementation Activator Process

You might want to review a Computer Based Training (CBT) module to understand how to perform a task, such as setting up user defined codes.

▶ **To access CBT for an implementation activator process**

1. Choose the implementation activator process for which you want to review CBT.
2. From the Row menu, choose CBT.

If a CBT module exists for the process, the CBT choice on the Row menu will be active. If no CBT module exists, the CBT choice on the Row menu will be disabled.

Adding an Attachment to an Implementation Activator Process

You can use the Attachments option in Customer Configuration to add text, graphics, and other OLE objects to an implementation activator process. For example, you can attach a note to explain special circumstances relating to an implementation activator process you completed. Or you can attach drawings, multimedia (AVI) files, and other types of objects to implementation activator processes.

▶ **To add an attachment to an implementation activator process**

1. On Customer Configuration, choose an implementation activator process, then from the Row menu, choose Attachments.

The system displays the Media Objects workspace.



The Media Objects workspace is split into two panels. The left panel is the icon panel and the right panel is the viewer panel. Icons for any files previously attached to the record appear in the icon panel.

2. From the File menu, choose Add, then choose one of the following:
 - Text
 - Image
 - OLE
 - Shortcut

Alternatively, in the icon panel, click the right mouse button. From the drop-down menu, choose one of the options from the list above.

See Also

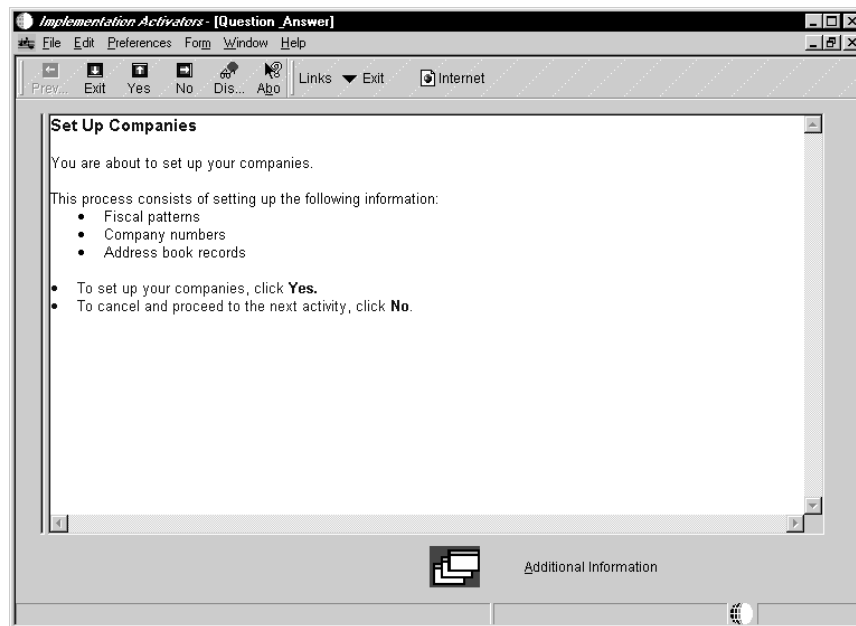
- *Working with Media Objects* in the *OneWorld Foundation Guide* for more detailed information about adding text, images, OLE objects, and shortcuts.

Adding Additional Information to an Implementation Activator Process

You can add additional information to a Question and Answer form within an implementation activator process, if necessary.

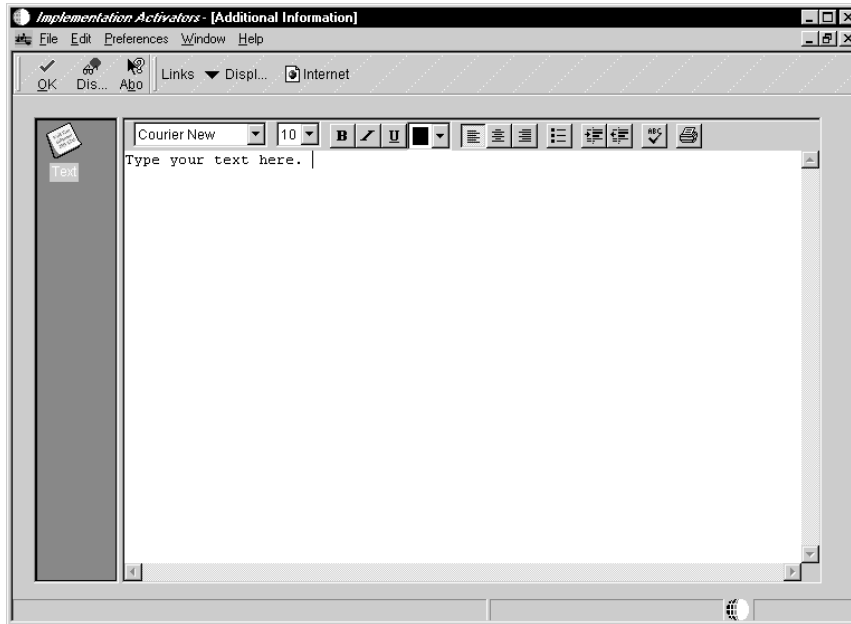
► To add additional information to an implementation activator

On a Question and Answer form



1. Click the Additional Information icon at the bottom of the form.

The system displays Additional Information.



2. Type the text you want to include.
3. Click OK.

The system returns to the Question and Answer form. A plus sign appears next to the Additional Information icon, indicating that more information is available for that form.

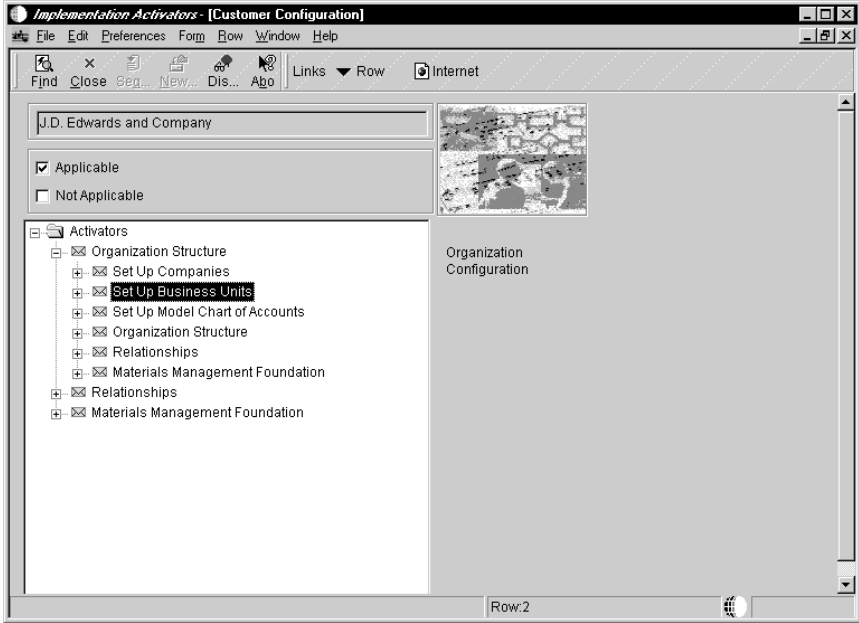
Reviewing a Reason for an Implementation Activator Process

You can review reasons for implementation activator processes to see why they were included in the configuration. Reasons are another form of online help, and merely tell you whether the implementation activator process was included because it was chosen during Customer Rough Cut, or because it was a required selection. Reasons are read only.

► To review a reason for an implementation activator process

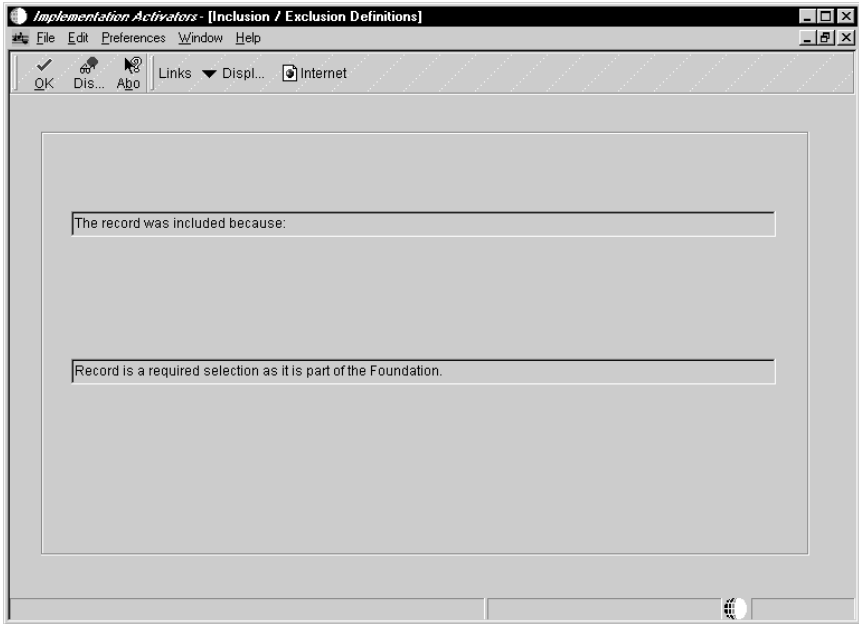
On Customer Configuration

1. Choose the implementation activator to which you want to review a reason.



2. From the Row menu, choose Reason.

The system displays the Inclusion/Exclusion Definitions form, which tells you why the implementation activator was included or excluded from the configuration.





Implementation Activator Process Setup

You can modify implementation activator processes to reflect the needs of a particular customer for which you will configure OneWorld. For example, you might want to modify an implementation activator process that reflects the needs of an international customer. Or, you might want to add more information to an implementation activator process, such as the address book numbers of the people who will be responsible for certain processes. You can also create new implementation activator processes or change the text within an activator question form.

The following topics are described:

- Creating an implementation activator process
- Adding activities to an implementation activator process
- Defining the flow of implementation activator processes
- Modifying implementation activator processes
- Creating dependencies for implementation activator processes
- Modifying text within an Activator Question form

Naming Conventions Used in Implementation Activator Processes

Implementation activator processes use specific naming conventions for process names, data item, and activities. These naming conventions are as follows:

Implementation Activator Process Names

You name implementation activator processes using the following naming convention:

*xx*yyyy

Where *xx* identifies the system code (use codes 55 through 59 for customer-specific processes)

yyyy identifies a sequential number



Data Items

Activators use data items to populate question and answer forms with text. These data items use the following naming convention:

Cxxxxyyy

where C identifies the data item as an activator data item

xxxx = the system code

yyy = a sequential number

Activity Names

Implementation activator processes use four naming conventions for activities, depending on their purpose. Following is an explanation of each of these activity types:

Question activity

Qxx, where Q indicates that it is a question, and xx is a sequential number. For example, Q10, Q20, Q30.

A question activity will typically contain a description naming convention that further identifies it as a specific question activity within the activator.

For example, ?re: Set Up Companies. When you add a description, make sure you add a space between the colon and the description.

The description naming convention shown above is used for translation purposes only. Customer-specific implementation activators do not need to follow the same naming convention.

Process confirm

Process confirm is always the first question within an activator and is always identified by Q00.

Action activity

Action activities are identified by Axx, where A identifies the activity as one that calls an application (for example, User Defined Codes), and xx is a sequential number (for example, A10, A20, A30).

Completion activity form

Completion activity forms are always identified with the naming convention Q99, which indicates that it is the last question within an implementation activator.

The system displays a completion form when a user has either completed an implementation activator or if the user exits out of the implementation activator without completing it.

Prototyping Implementation Activator Processes

Before you start creating an implementation activator process, you should create a prototype of the process flow on paper or on an Excel spreadsheet. This prototype should include the description or name of each implementation activator process; the application and the form it will be associated with; version (if applicable); data dictionary item names (for question and answer activities); and comments.

After you create the prototype, you should create prototypes of the glossary items (the text that appears within an Activator Question form). These glossary items should describe the activities.

See Also

- *Appendix A* for an example of an implementation activator process prototype.
- *Modifying Text within an Activator Question Form* for an example of an Activator Question form, and for information about modifying or adding glossary items.

Creating an Implementation Activator Process

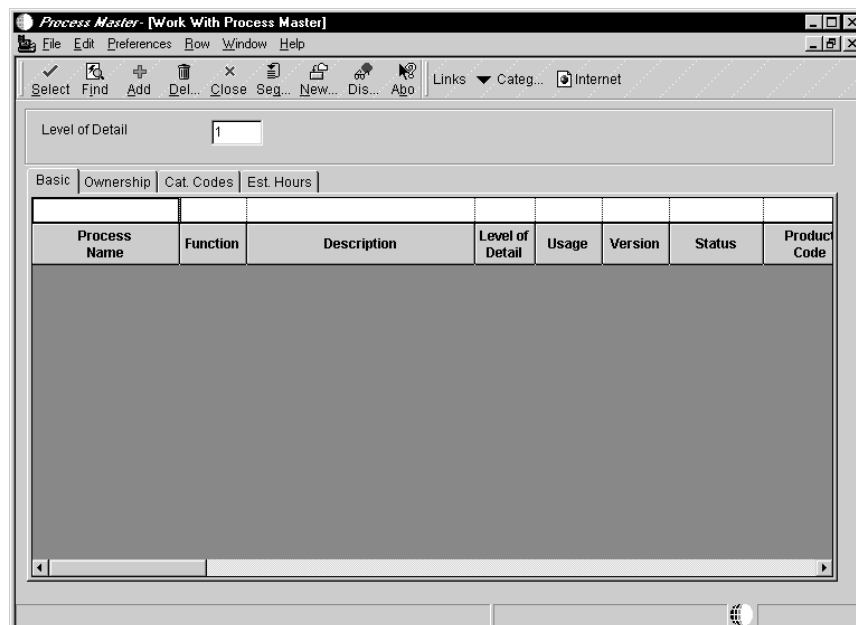
Implementation activator processes typically contain interactive application activities that call a question and answer form, a completion form (where you can mark an implementation activator as complete, pending, waiting for approval, and so forth), or a User Defined Codes form.

An implementation activator process can also call another implementation activator process (or subprocess), through a process activity. You attach subprocesses when you have certain subsets of activities common to multiple implementation activator processes or that get used more than once.

When you create an implementation activator process, you add a level of detail to it that specifies the indent level on the Process Master application.

► To create an implementation activator process

From the Activator Setup menu (G02141), choose Process Master.



1. On Work With Process Master, click Add.

The system displays a dialog box that asks you if you want assistance in creating processes. Click No.

The system displays Process Revisions.

The screenshot shows a software window titled "Process Master - [Process Revisions]". The window has a menu bar with "File", "Edit", "Preferences", "Form", "Window", and "Help". Below the menu bar is a toolbar with icons for "OK", "Cancel", "Dismiss", and "Apply", along with a "Links" dropdown and a "Categorize" button. The main area of the window is a form with the following fields and values:

Process	55001
Version	1
Parent	
Process Type	02 ActivEra - Activator
Description	Set Up Companies
Product Code	55 Reserved for Clients
Level Of Detail	9 LOD 9
Category Code 1	
Category Code 2	
Category Code 3	
Usage	3 Title/Customer
Function	
CBT Path	

There is also a checkbox labeled "Create Instances" which is currently unchecked.

2. On Process Revisions, complete the following fields:

- Process

Enter a unique name for the process using the appropriate naming conventions.

See Naming Conventions Used in Implementation Activator Processes.

- Version

You would normally enter 1 in this field.

- Parent
- Process Type

Enter 02 to identify this process as an activator process.

- Description
- Product Code

Enter a system code for the implementation activator process you are creating. If you are creating a customer-specific implementation activator, enter a system code between 55 and 59.

- Level of Detail

Enter a number from 1 to 9 in this field to specify the level at which this implementation activator process will appear within the hierarchy in the Process Master, 1 being the highest level and 9 being the lowest level.

- Category Codes
- Usage
- Function
- CBT Path

Enter the location at which a particular CBT module exists.

3. Do not choose the Create Instances option.
4. Click OK.

See Also

- *Adding Activities to an Implementation Activator Process*

Adding Activities to an Implementation Activator Process

The activities within implementation activator processes initiate interactive applications that display Activator Question forms or applications. Activities can also invoke other processes (subprocesses). You use Activity Revisions to add interactive application and process activities to an implementation activator process

You can also add activity conditions to activities. These activity conditions determine when an activity will be invoked within an implementation activator process.

The following topics are described:

- Adding an interactive application activity
- Adding a process activity

Adding an Interactive Application Activity

An interactive application activity invokes a OneWorld interactive application. In Implementation Activators, you can choose from three interactive application types through Activator Question Forms (P988825) when you create an interactive application. They are:

Glossary (W988825A)

This form calls a media object that is populated by a data item containing question and answer text.

Completion Options (W988825F)

This form displays completion options such as:

- Mark the process as unopened
- Mark the process as in progress
- Mark the process as pending
- Mark the process as complete

Work With User Defined Codes (W988825K)

This form displays user defined codes for a particular system code.

Note that even though this guide describes how to set up interactive activities that are specific to implementation activator processes, you can set up other interactive activities that invoke other forms within an application, depending on your needs. For example, you might want to set up an interactive activity that invokes a form within the Business Units application (P0006).

Complete the following tasks:

- Add a glossary activity
- Add a completion activity
- Add an action activity

► To add a glossary activity

1. On Process Relationships, find the implementation activator process to which you want to add the glossary application activity and choose Insert Activity from the Row menu.

The system displays a dialog box that asks if you want assistance from the Director in setting up activities. Click No.

The system displays the Activity Revisions form with the Interactive Application option already chosen for you.

The screenshot shows a dialog box titled "Process Master - [Activity Revisions]". The window has a menu bar with "File", "Edit", "Preferences", "Form", "Window", and "Help". Below the menu bar is a toolbar with buttons for "OK", "Can...", "Dis...", "Ab...", and "Internet". The main area contains several input fields: "Process" with the value "55001", "Version" with "1", "Activity" with "Q01", and "Description" with "?re: Set Up Companies". Below these fields is a section with radio button options: "Start", "Batch Application", "Interactive Application" (which is selected), "Function", "Message", "Process", "Run Executable", and "Halt Process". At the bottom of the dialog is a "Definition" section with a large empty text area.

2. On Activity Revisions, complete the following fields:

- Activity

Use the appropriate naming convention for this activity type, such as Q00 or Q01. If you want this activity to call a process confirmation form, enter Q00.

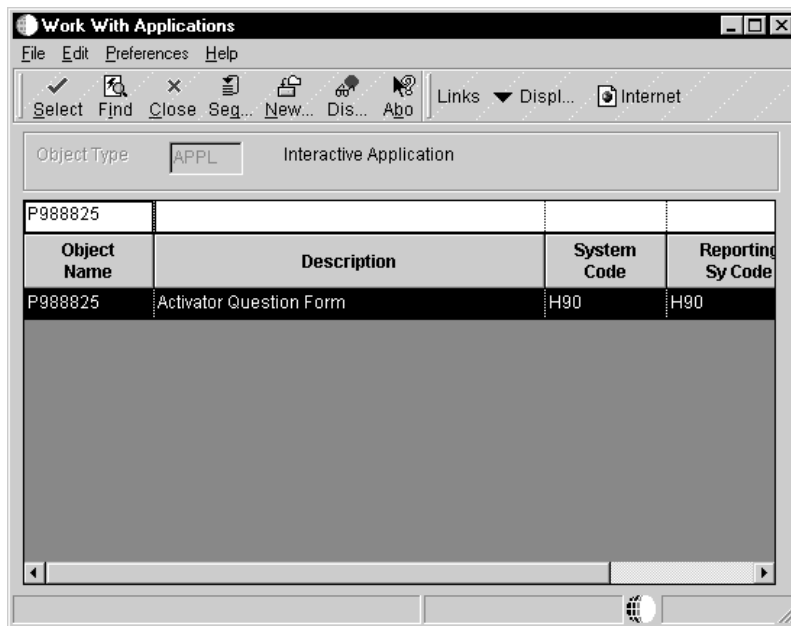
You should number your glossary activities using increments of 5 or 10. For example, Q5, Q10, Q15, Q20 or Q10, Q20, Q30, and so on.

- Description

Use the appropriate naming convention for the description. For example, ?re: Set Up Companies.

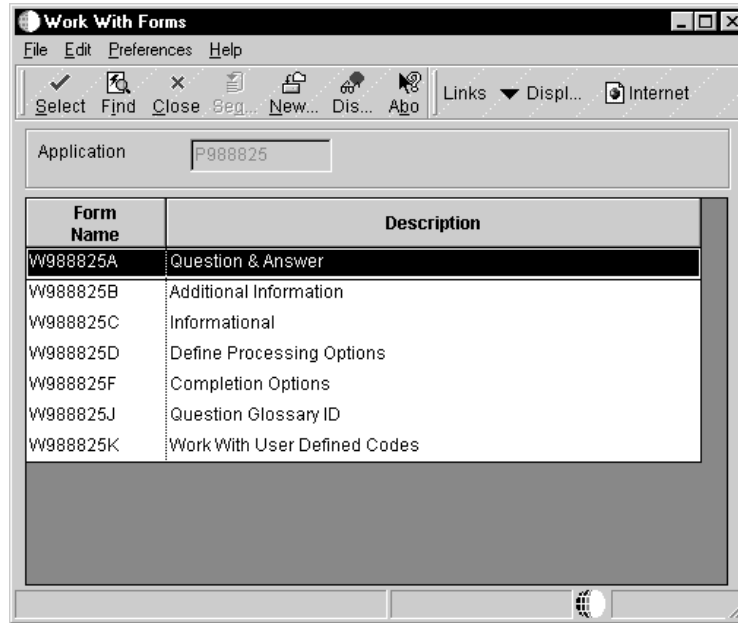
See *Naming Conventions Used in Implementation Activators* for more information regarding naming conventions for descriptions.

3. Choose Edit Definition from the Form menu.



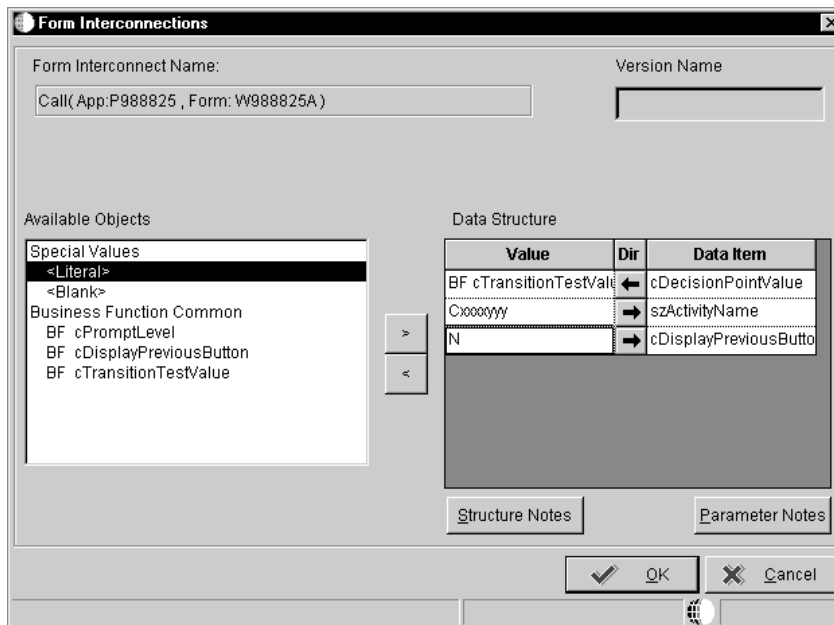
4. On Work With Applications, find and select P988825.

The system displays Work With Forms.



- On Work With Forms, choose form W988825A.

The system displays Form Interconnections.



- On Form Interconnections, pass data from the available objects to the values as follows:
 - cDecisionPointValue: map this data item to BF cTransitionTestValue. The directional arrow must point from the data item to the value.

- szActivityName: Double-click <Literal> and enter the appropriate data item value (for example, C09001010). The directional arrow must point from the value to the data item.
 - cDisplayPreviousButton: If you are creating a Q00 (Process Confirm) activity, always map this data item as a literal “N.” For all other Q activities, map this data item as a literal “Y.” The directional arrow must point from the value to the data item.
7. Click OK.
 8. On Activity Revisons, click OK.

► To add a completion activity

1. On Process Relationships, find the implementation activator process to which you want to add the completion activity and choose Insert Activity from the Row menu.

The system displays a dialog box that asks if you want assistance from the Director in setting up activities. Click No.

The system displays the Activity Revisions form with the Interactive Application option already chosen for you. Make note of the process ID for later use.

The screenshot shows the 'Process Master - [Activity Revisions]' dialog box. The 'Process' field is set to '1241' and the 'Organization Structure/1' is selected. The 'Version' field is '1'. The 'Activity' field is 'Q99' and the 'Description' is 'Completion Status'. The 'Interactive Application' radio button is selected. The 'Definition' field contains the text 'Call(App:P988825, Form: W988825F)'.

2. On Activity Revisions, complete the following fields:
 - Activity

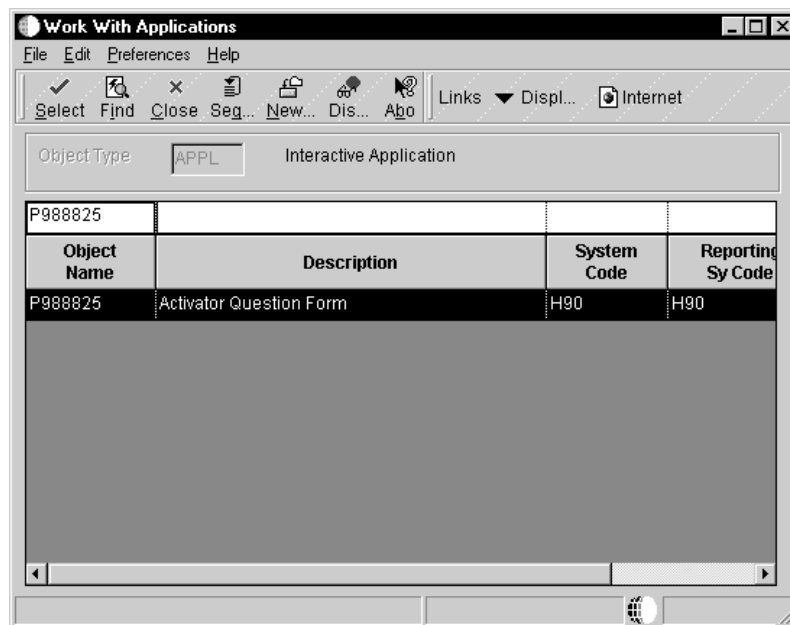
Enter Q99 for the activity name.

Note that completion activities should have a number greater than your last “Q” activity if your last “Q” (Glossary) activity contains a number greater than 100. For example, if your last “Q” activity is 150, you would name the completion activity Q155 (if you are numbering activities in an increment of 5), or Q160 (if you are numbering the activities in an increment of 10).

- Description

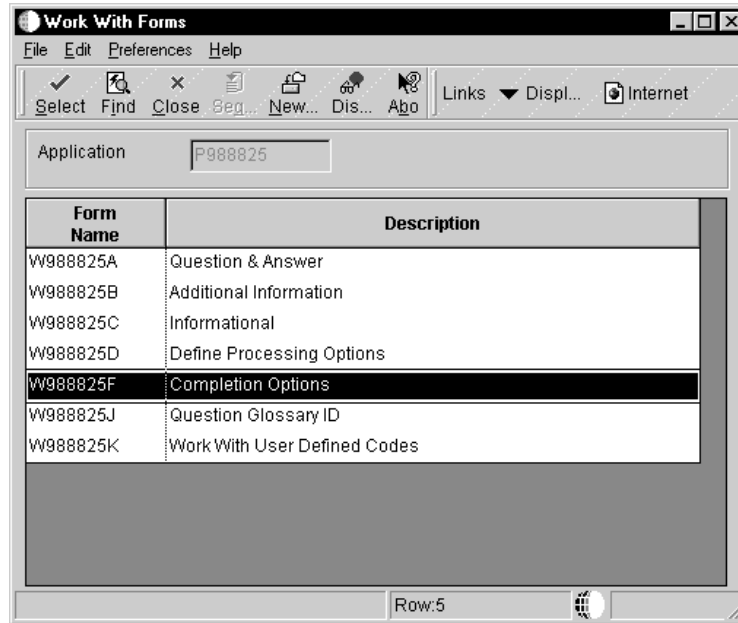
Because this activity calls the Completion form, you might want to enter a description such as Completion Status.

3. Choose Edit Definition from the Form menu.



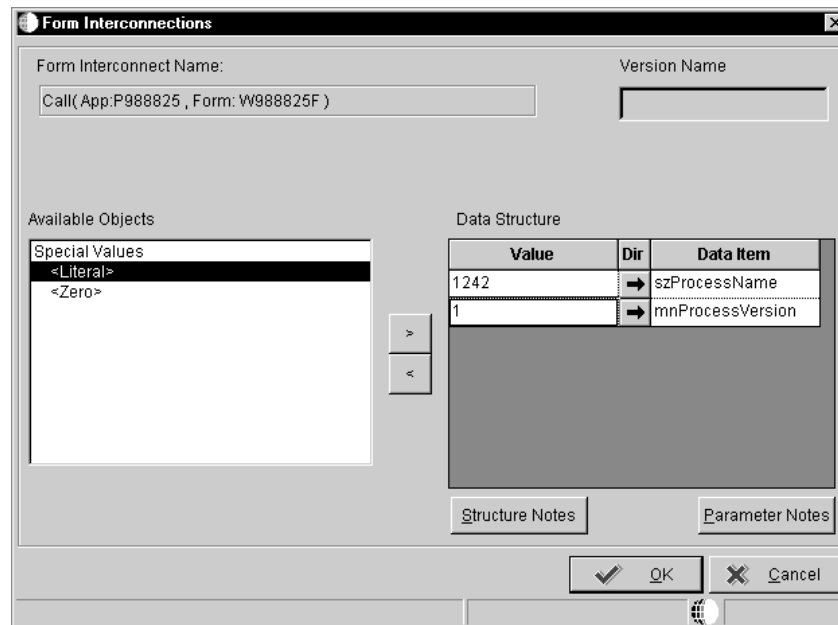
4. On Work With Applications, find and select P988825.

The system displays Work With Forms.



5. On Work With Forms, choose form W988825F.

The system displays Form Interconnections.



6. On Form Interconnections, pass data from the available objects to the values as follows:
 - szProcessName: Double-click <Literal> and enter the name of the implementation activator that will be completed. The directional arrow must point from the value to the data item.

- mnProcessVersion: Double-click <Literal> and enter a 1 (one), unless you want to use a different version from this one. The directional arrow must point from the value to the data item.
7. Click OK.
 8. On Activity Revisions, click OK.

▶ To add an action activity

An action activity can invoke a User Defined Codes form or any other interactive application form.

1. On Process Relationships, find the implementation activator process to which you want to add the action activity and choose Insert Activity from the Row menu.

The system displays a dialog box that asks if you want assistance from the Director in setting up activities. Click No.

The system displays the Activity Revisions form with the Interactive Application option already chosen for you.

The screenshot shows a dialog box titled "Process Relationships - [Activity Revisions]". It has a menu bar with "File", "Edit", "Preferences", "Form", "Window", and "Help". Below the menu bar is a toolbar with buttons for "OK", "Can...", "Dis...", and "Ab...", along with a "Links" dropdown and an "Internet" icon. The main area contains several input fields: "Process" with the value "55002", "Version" with "1", "Activity" with "A10", and "Description" with "Set Up Address Book UDCs". Below these fields is a section with radio buttons for different activity types: "Start", "Batch Application", "Interactive Application", "Function", "Message", "Process", "Run Executable", and "Halt Process". The "Interactive Application" radio button is selected. At the bottom of the dialog is a "Definition" text area.

2. On Activity Revisions, complete the following fields:

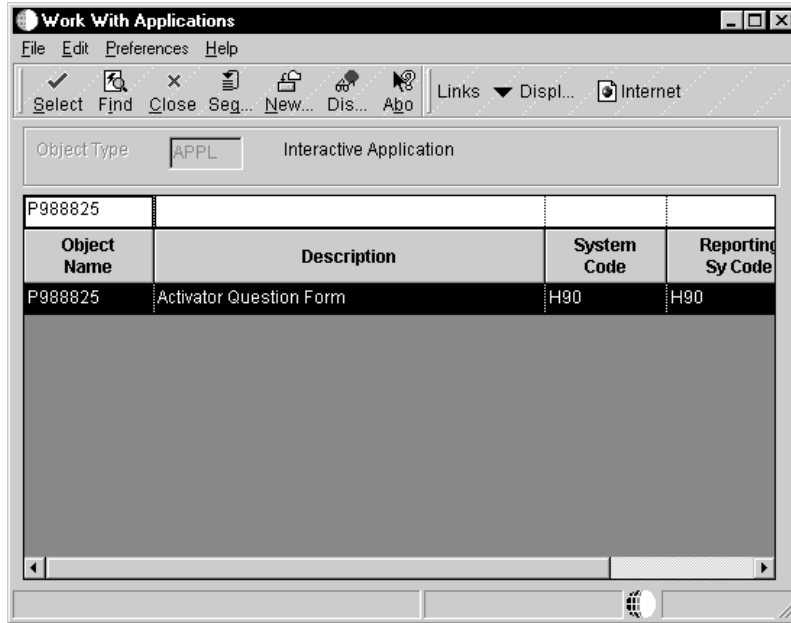
- Activity

Follow the appropriate naming conventions. For example, A10.

- Description

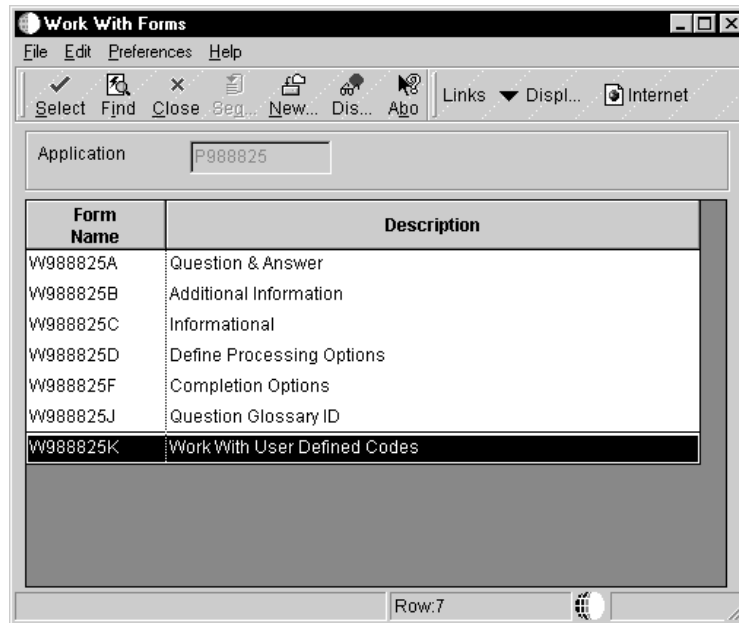
Enter a description for this activity. For example, Set Up Address Book UDCs.

3. Choose Edit Definition from the Form menu.



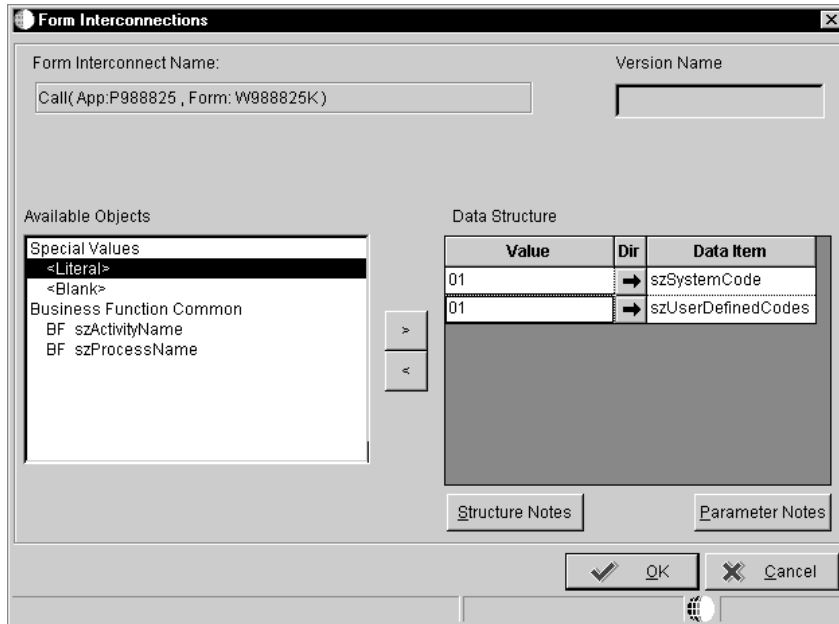
4. On Work With Applications, find and select P988825.

The system displays Work With Forms.



5. On Work With Forms, choose form W988825K.

The system displays Form Interconnections.



6. On Form Interconnections, pass data from the available objects to the values as follows:
 - szSystemCode: Double-click <Literal> and enter the appropriate system code for the UDC form you want to display. The directional arrow must point from the value to the data item.
 - szUserDefinedCode: Double-click <Literal> and enter the appropriate user defined code you want to display. The directional arrow must point from the value to the data item.
7. Click OK.
8. On Activity Revisions, click OK.

See Also

- *Understanding Form Interconnections* in the *OneWorld Development Tools Guide* for more information about Form Interconnections.

Adding a Process Activity

A process activity starts another implementation activator process, also referred to as a subprocess. A subprocess includes its own set of activities. When you add a process activity, you enter the name of an implementation activator process that already exists in the system, then define the parameters under which that subprocess will be invoked within the implementation activator process.

► To add a Process Activity

1. On Process Relationships, find the implementation activator process to which you want to add a process activity and choose Insert Activity from the Row menu.

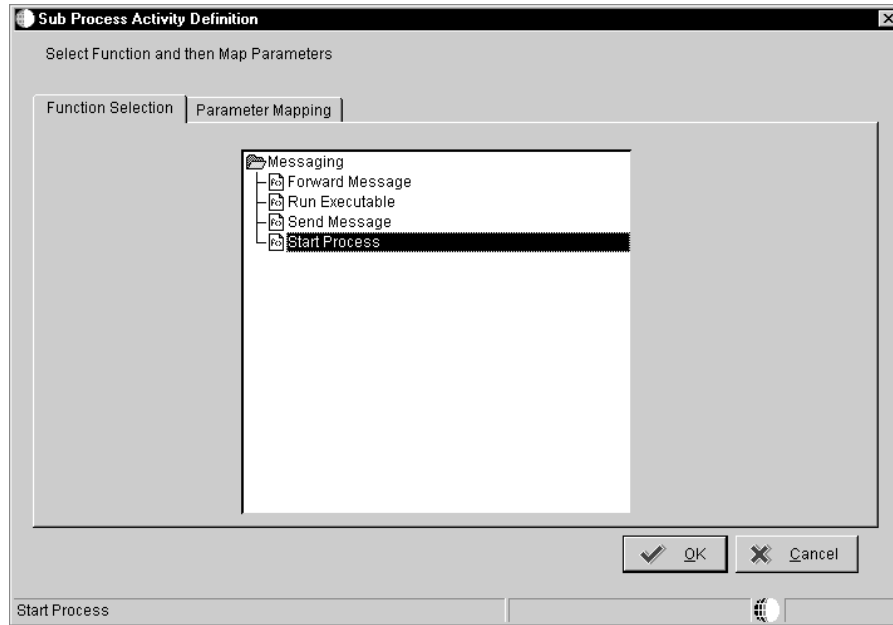
The system displays a dialog box that asks if you want assistance from the Director in setting up activities. Click No.

The system displays the Activity Revisions form.

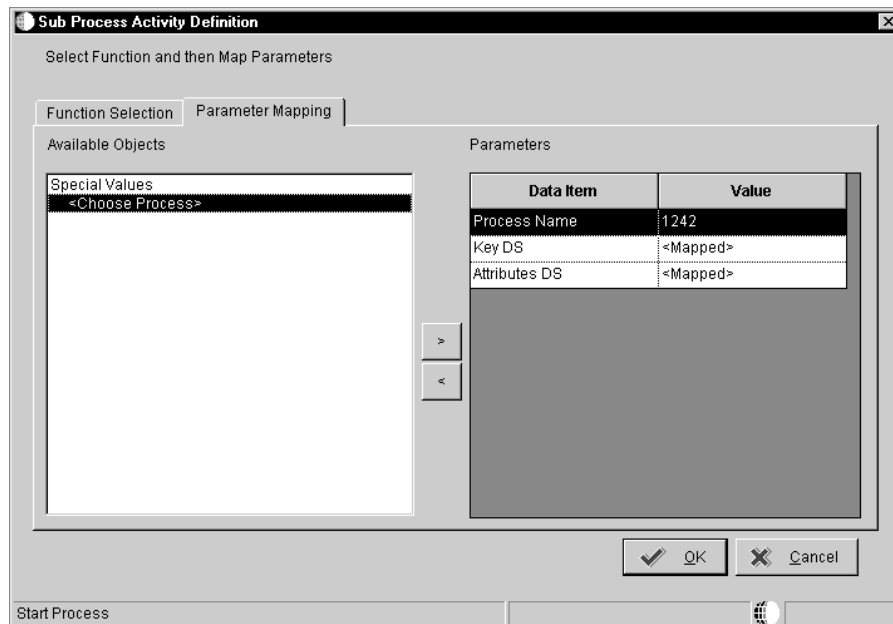
2. On Activity Revisions, enter the name of an existing implementation activator process you want to attach as a subprocess in the Activity field.
3. Choose the Process option.

The description of the existing process appears in the Description field.

4. Choose Edit Definition from the Form menu.



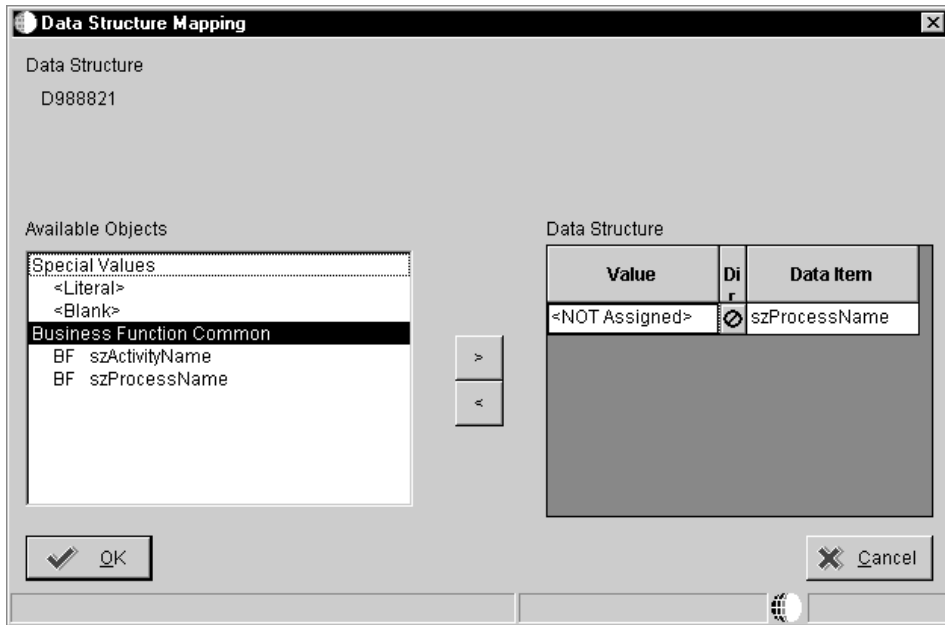
5. On Sub Process Activity Definition, open the Messaging folder and choose the Start Process option.
6. Click the Parameter Mapping tab.



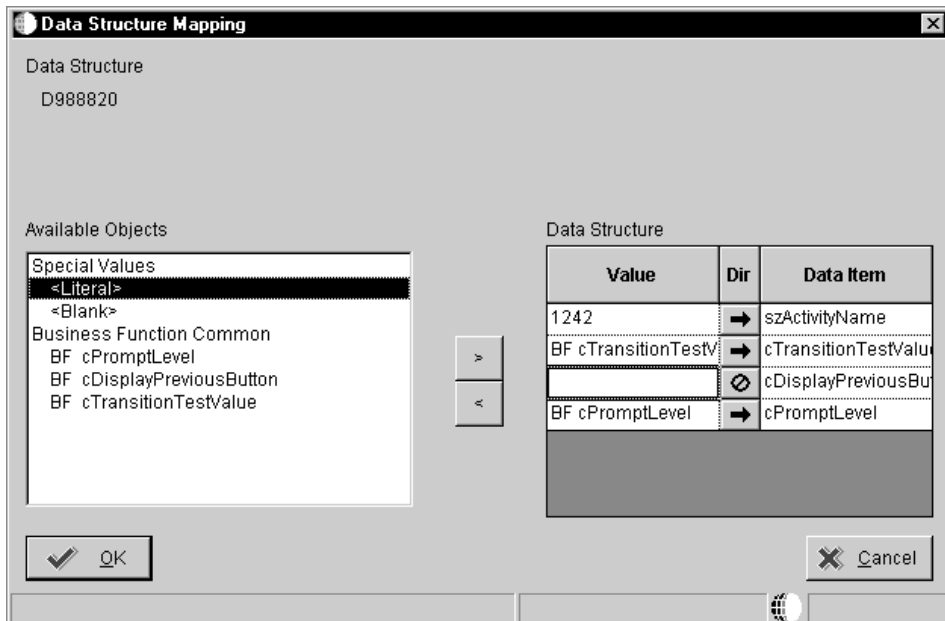
7. Map the data items as follows:
 - Process Name: Choose Process Name under Parameters and double-click <Choose Process> under Available Objects. On Process Search and Select, find and select the subprocess that will be launched by the implementation activator process.

Adding Activities to an Implementation Activator Process

- Key DS: Choose Key DS and double-click <Define Mapping> under Available Objects.



8. On Data Structure Mapping, click OK without mapping a Key Data Structure.
9. Choose Attribute DS and double-click <Define Mapping> under Available Objects



10. On Data Structure Mapping, map the Attribute Data Structure as follows:
 - `szActivityName`: Double-click <Literal> under Available Objects. Enter the process name value and click OK. The directional arrow must point from the value to the data item.
 - `cTransitionTestValue`: Choose BF `cTransitionTestValue`. The directional arrow must point from the value to the data item.
 - `cDisplayPreviousButton`: Do not map this value.
 - `cPromptLevel`: Choose BF `cPromptLevel`. The directional arrow must point from the value to the data item.
11. Click OK. To save and close each form.
12. On Activity Revisions, click OK.

See Also

- *Defining the Flow of Implementation Activator Processes* for more information about sequencing the order of activities and subprocesses within an implementation activator process and adding activity conditions to activities.

Defining the Flow of Implementation Activator Processes

After you create the activities for your implementation activator process, you must define the flow so that the system will invoke the activities in the order in which you want them to execute.

You can also add activity conditions to activities at this point. Activity conditions act like event rules and specify what happens when a particular activity executes. For example, the activity condition called “NEXT” might trigger the system to invoke a UDC activity, or an activity that calls a UDC form. Or, the activity condition called “PREVIOUS” might trigger the system to invoke a previous question and answer form.

In the Implementation Activators application activity conditions are predefined; you do not have to create your own. You use these predefined conditions to define the flow of activities within activators.

After you define the flow of the activities, you should review the flow using Advanced Setup.

Complete the following tasks:

- Define the flow of activities
- Review the flow of activities

Activity Conditions

The predefined activity conditions that implementation activator processes use are as follows:

EXIT

Use this condition for question (or “Q”) activities.

You use EXIT in association with Q99 as the Next Activity. When the user clicks the Exit button, the system displays Q99 (the completion form).

Do not use this condition with a Start activity or with Q99 (completion form).

NEXT	<p>Use this condition for action (“A”) and START activities. You can use NEXT conditions for subprocesses as well. The last “A” activity’s NEXT condition should point to END.</p> <p>NEXT conditions specify which subprocesses and activities will appear on the Implementation Activators tree structure.</p>
NO	<p>Use this condition only for question activities. For example, if a user clicks No on a question and answer form, the system might invoke the next question activity, depending on how the process flow is defined. If a user clicks No on a confirmation form (Q00) the system should invoke Q99 (the completion activity).</p> <p>Do not use the NO condition with Q99.</p>
NOPROMPT	<p>Use this condition only for “A” activities. It points to the next “A” activity if the user has turned off prompting through the processing options in Implementation Activators. This condition shows the system where to go when the user has activated the processing option that hides the yes/no prompts. If the user has turned off prompting, the question and answer forms do not appear in the system.</p> <p>Do not use NOPROMPT with the last “A” activity in the flow.</p>
PREVIOUS	<p>Use this condition only for questions. It normally points to the previous question (“Q”) activity.</p> <p>Do not use the PREVIOUS condition with Q99.</p>
PROMPT	<p>Use this condition only for “A” activities. It points to the next question activity. This condition tells the system what it should invoke when the user has turned on prompting through the processing options.</p>
YES	<p>Use this condition only for question activities. It normally points to the “A” activity, which runs when the user clicks Yes, except for Q00, where YESNOPROMPT and YESPROMPT are used.</p> <p>Do not use the YES condition with Q99.</p>

YESNOPROMP Use this condition for Q00 (confirmation form) only. It points to the first “A” activity and is valid only if the user turns off prompting through the processing options.

Do not enter YESNOPROMPT when you add this condition. You must enter YESNOPROMP instead. If you add the T to the end of the word, it could cause a system error.

YESPROMPT Use this condition for Q00 only. It points to the first question activity and is valid only if the user turns on prompting through the processing options.

Understanding the Flow of Activities

The following guidelines describe how to structure the flow of activities within an implementation activator process.

Activity conditions used for question (“Q”) Activities You can use the following conditions with question activities: EXIT, NO, PREVIOUS, NEXT.

Do not use EXIT, NO, PREVIOUS, or NEXT conditions with Q00 (process confirmation form) or Q99 (completion form).

Action (“A”) Activities (also called UDC activities) You can use the following conditions with action activities: NEXT, NOPROMPT, PROMPT.

Special Cases

Leave the condition blank for the first START activity that points to the Next Activity Q00 (confirmation form). Leaving the condition blank indicates that the Next Activity will automatically execute unconditionally. However, you would enter NEXT as the condition for a second Start activity that points to the first “A” activity as the Next Activity, such as A10. For example, in the first row in Advanced Setup you would first enter START as the Activity, then Q00 as the Next Activity with no activity condition. On the next row, you would enter START, then NEXT as the condition, and A10 as the Next Activity.

START could also point to a subprocess, not just an activity.

Activity	Activity Condition	Next Activity
START		Q00
START	NEXT	A10

Do not enter a condition or a Next Activity for an END activity, as shown in the example below.

Activity	Activity Condition	Next Activity
END		

The last activity (“A”) in a process flow must include a NEXT condition that points to an END Next Activity. In addition, it must not contain a condition that points to Q99, as shown below.

Activity	Activity Condition	Next Activity
A10	NEXT	END

and

Activity	Activity Condition	Next Activity
A10		Q99

Do not add a condition for a Q99 activity. In addition, make sure that Q99 always points to END, as shown in the following example:

Activity	Activity Condition	Next Activity
Q99		END

Example: Process Flow

The following example shows the process flow for implementation activator process 5291 (Supplier Information). In this example, the Start activity invokes the Q00 question and answer form, which tells the user that they are about to set up supplier information, such as processing options for supplier master entry, supplier master data, and supplier supplemental data. If the user clicks Yes to continue, the system invokes the subprocess for setting up processing options. Otherwise, if the user clicks No or Exit, the system invokes the Q99 activity, which is the completion form.

Activity	Activity Condition	Next Activity
START		Q00
START	NEXT	2134
Q00	EXIT	Q99
Q00	NO	Q99
Q00	YES	2134
2134		1245
2134	NEXT	1245
Q99		END

1245		Q99
1245	NEXT	END
END		

Example: Flow of Activities

The following example shows the flow of question and action activities Q00 through A20 for the Set Up Stock Items process. In this example, if a user clicks No in the Q00 confirmation form, the system invokes the next activity, which is the completion form (Q99) that allows the user to mark the implementation activator process as completed, not completed, pending, and so forth.

The YESNOPROMP between the Q00 and A10 activities means that if the user answers Yes to the Q00 question and answer form, the system invokes the A10 activity and will not invoke other question and answer forms throughout the rest of the implementation activator process. The YESNOPROMP activity condition specifies that if the user has turned off prompting through the processing options in Customer Configuration, then the system will not invoke any question activities.

The YESPROMPT activity condition means that if the user did not turn off the prompts through the processing options, the system invokes the question activities for the next activity specified (usually Q10 or the first question after Q00). If the user clicks Yes on a question and answer form, such as Q10, the system invokes the specified “A” activity, such as A10, which normally launches an application. If the user cancels, closes an application, or clicks OK, the system invokes the next question and answer form, such as Q20, as specified by the prompt conditions when prompts are on. Otherwise, the Next Activity for NOPROMPT is invoked. After the user works with the A10 activity and clicks OK or Close, the system prompts them with the next question and answer form, such as Q20. If the user clicks Yes in that question and answer form, the system invokes the activity associated with it, such as A20.

Any activities that include a PREVIOUS activity condition will invoke the previous form. For example, if a user clicks Previous on the Q20 question and answer form, the previous question and answer form (Q10) will be invoked.

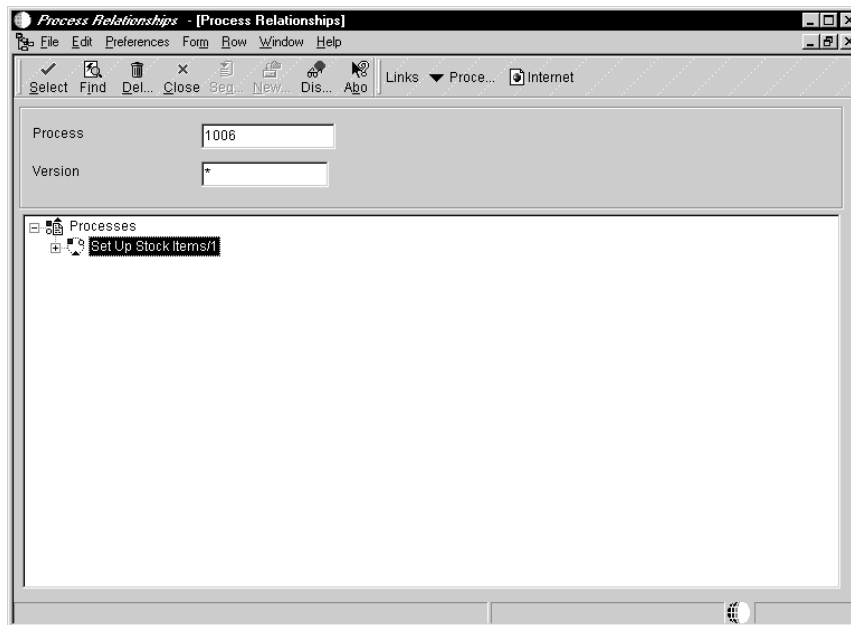
Activity	Activity Condition	Next Activity
START		Q00
START	NEXT	A10
Q00	EXIT	Q99
Q00	NO	Q99
Q00	YESNOPROMP	A10
Q00	YESPROMPT	Q10
A10	NEXT	A20
A10	NOPROMPT	A20

A10	PROMPT	Q20
Q99		END
Q10	EXIT	Q99
Q10	NO	Q20
Q10	PREVIOUS	Q00
Q10	YES	A10
A20		Q99
A20	NEXT	END
Q20	EXIT	Q99
Q20	NO	Q99
Q20	PREVIOUS	Q10
Q20	YES	A20
END		

The activity conditions cover all combinations a user might take, whether they click Next, Previous, Yes, or No; and whether they turns prompting on or off through the processing options for the Implementation Activators application.

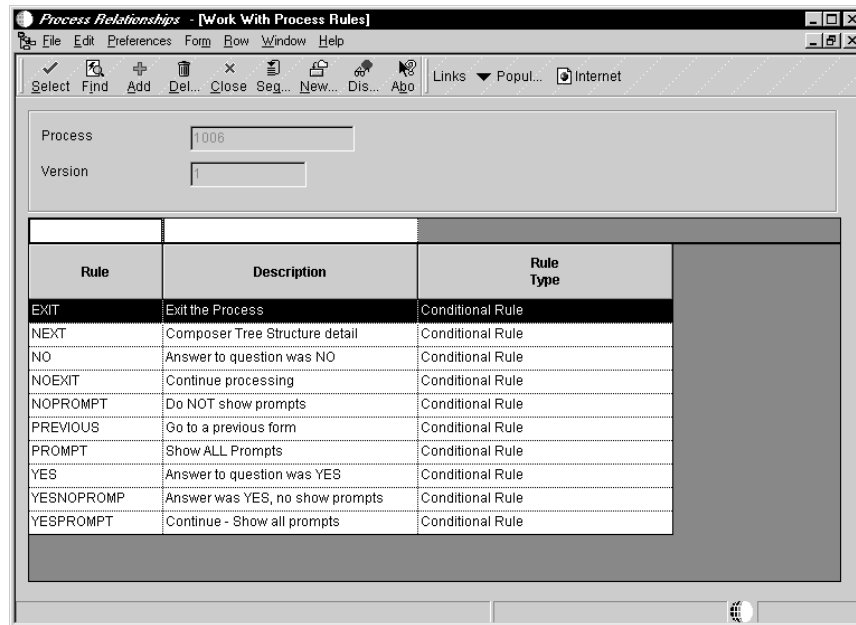
► **To define the flow of activities**

From Activator Setup (G02141), choose Process Relationships.

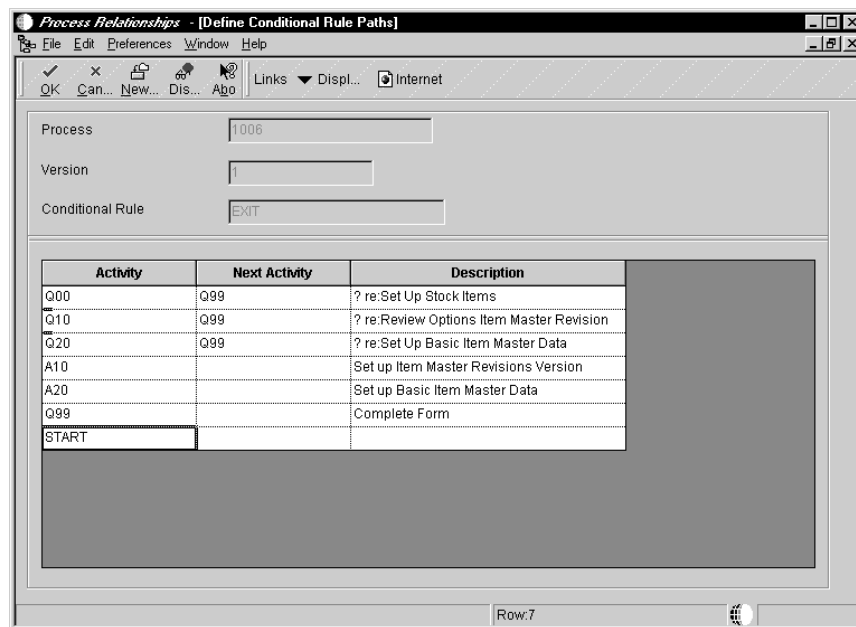


1. On Process Relationships, find the implementation activator process for which you want to define the flow of activities.

2. From the Row menu, choose Process Rules.
3. On Work With Process Rules, choose Populate Rules from the Form menu.



4. On Define Conditional Rule Paths, choose the activity condition that you want to use.
For example, choose EXIT.
5. Choose Define Flow from the Row menu.



6. On Define Conditional Rule Paths, notice that the system displays all activities associated with the particular activity condition that you chose.
7. Enter the next activity that you want the system to invoke in the Next Activity field.

You add a next activity only to those activities for which you want to associate the activity condition that you chose. For example, for the EXIT activity condition, you would enter Q99 as the Next Activity for all the question activities (Q00 through Q40). Therefore, when the user clicks Exit at any of these points, the system invokes the next activity, which is the completion form defined for Q99.

See the guidelines in *Understanding the Flow of Activities* for more information about the activity conditions that should be associated with an activity.

8. Continue to add next activities as necessary.
9. When you are finished, click OK.

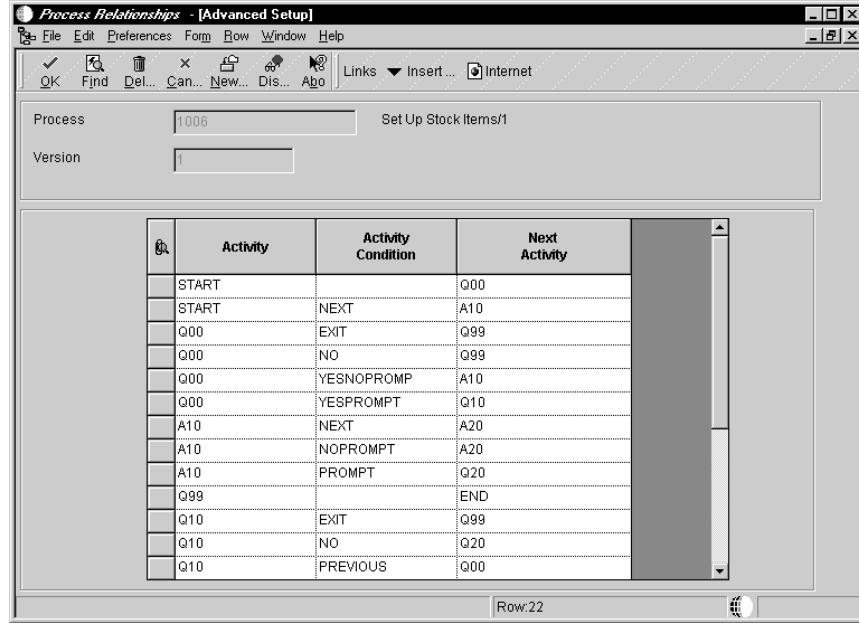
The system returns to Work With Process Rules.

10. Choose the next activity condition you want to work with (for example, NEXT), and repeat the steps above.

To review the flow of activities

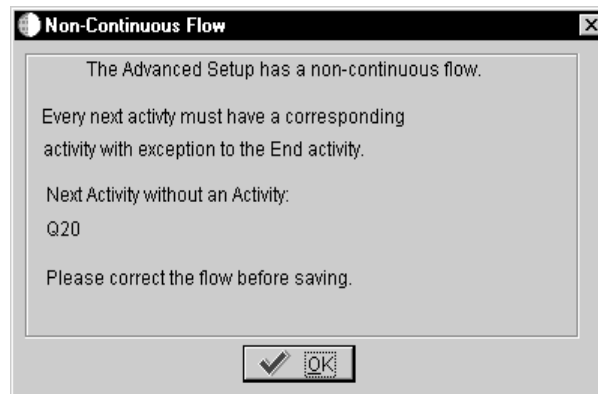
After you define the flow of activities, you should review them in Advanced Setup to ensure that they follow the correct progression.

1. On Process Relationships, choose Advanced Setup from the Row menu.



2. On Advanced Setup, review and resequence Activities and Next Activities as necessary.

If any activity does not have a corresponding next activity associated with it, an error message appears. For example, if the A10 activity is not associated with a next activity, such as A20. The system will not save the updated information in Advanced Setup until you ensure that there are relationships between all activities in the implementation activator process.



NOTE: Activity information is stored in the F98810 file and activity relationship information is stored in the F98830 file. If you click OK on Advanced Setup after resequencing the activities and conditions and there is an activity that is not represented in the F98830 file, the system will prompt you that the flow is noncontinuous.

If you delete an activity, the system asks for confirmation that you want to delete that particular activity from Advanced Setup.

Modifying Implementation Activator Processes

You might want to add to or change information in an implementation activator process if, for example, you want to specify who will “own” the implementation activator, or if you want to change the description of an implementation activator. You can also add category codes to the process or change the process name and the level of detail. Note that when you change the level of detail it only affects the appearance of the flow of implementation activators. It does not impact the level of operation of the implementation activators.

If you want to make changes to activities within an implementation activator, you must deactivate it first, then reactivate it after you have made the change.

Complete the following tasks:

- Activating or deactivating an implementation activator process
- Adding supplemental information to an implementation activator process

Activating or Deactivating an Implementation Activator Process

You deactivate an implementation activator process when you want to modify the activator or any activities within it, or when you do not want the activator to be available to a particular client. You activate an implementation activator process after you create or modify it.

When you activate an implementation activator process, the system automatically verifies the activities and subprocesses contained within it. The system ensures that all activities and subprocesses have a start and an end point, contain event rules, and include relationships between activities and activity conditions. If the implementation activator process cannot be verified, it will not be activated.

If errors exist, the system notifies you where in the implementation activator process the errors appear.



To activate or deactivate an implementation activator process

From Activator Setup (G02141), choose Process Master.

1. On Work With Process Master, find and choose the implementation activator process you want to activate or deactivate.

2. To activate an implementation activator process, choose Activate from the Row menu.

To deactivate an implementation activator process, choose Deactivate from the Row menu.

Adding Supplemental Information to an Implementation Activator Process

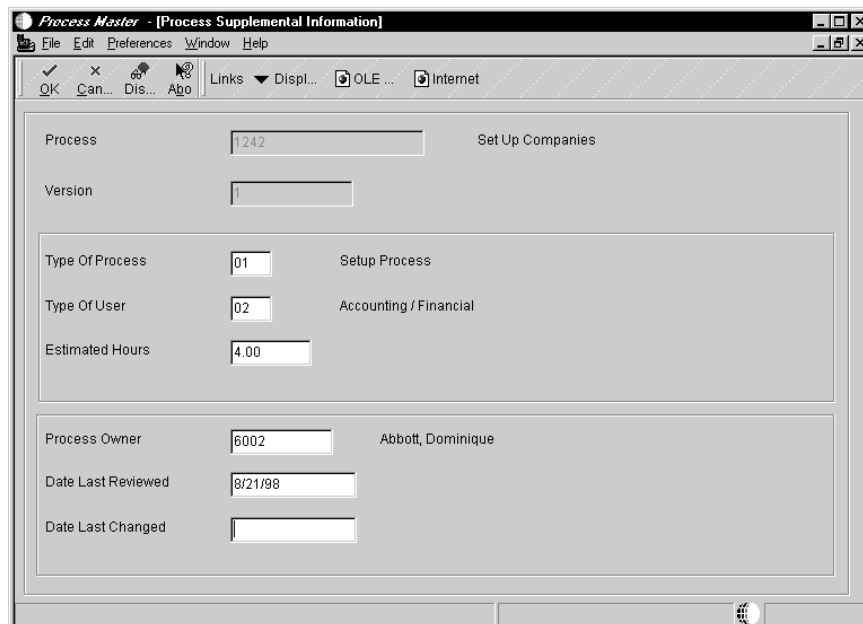
You can add supplemental information, such as specifying if an implementation activator process is a setup process, or a daily, periodic, or technical process. You can also specify which type of user will be involved with a process (such as accounting, technical, or operations); the number of hours you estimate it will take to complete a process; and the process owner, date last reviewed, and date last changed.

You do not have to deactivate an implementation activator process if you are only adding supplemental information to it.

► To add supplemental information to an implementation activator process

From Activator Setup (G02141), choose Process Master.

1. On Work With Process Master, find the implementation activator process to which you want to add supplemental information and choose it.
2. Choose Supplemental Info from the Row menu.



The screenshot shows a dialog box titled "Process Master - [Process Supplemental Information]". The dialog box has a menu bar with "File", "Edit", "Preferences", "Window", and "Help". Below the menu bar is a toolbar with icons for "OK", "Cancel", "Dismiss", and "Apply", along with "Links", "Display...", "OLE...", and "Internet". The main area of the dialog box contains several input fields and labels:

- Process: 1242 (with "Set Up Companies" text to the right)
- Version: 1
- Type Of Process: 01 (with "Setup Process" text to the right)
- Type Of User: 02 (with "Accounting / Financial" text to the right)
- Estimated Hours: 4.00
- Process Owner: 6002 (with "Abbott, Dominique" text to the right)
- Date Last Reviewed: 8/21/98
- Date Last Changed: (empty field)

3. On Process Supplemental Information, complete the following fields as necessary and click OK:

- Type Of Process
- Type Of User
- Estimated Hours
- Process Owner
- Date Last Reviewed
- Date Last Changed

Field	Explanation
Type Of Process	Classifies the process type. The valid values are: <ul style="list-style-type: none"> • 01 = Setup • 02 = Daily • 03 = Periodic • 04 = Technical
Type Of User	A code that indicates the type of knowledge a user needs to understand the process. Valid values are: <ul style="list-style-type: none"> 01 Technical 02 Accounting 03 Operations
Estimated Hours	The estimated number of hours to complete the process.
Process Owner	The user who creates and reviews the process setup.
Date Last Reviewed	A date that indicates when the process was last reviewed.
Date Last Changed	The last date that a change was made to either the process's master data or to the activity setup for the process.

Creating Dependencies for Implementation Activator Processes

You create dependencies for application suites, industries, and geographic regions to narrow the processes displayed to the user when they run Implementation Activators. You can assign dependencies to a specific process.

You can create dependencies for system codes. The dependencies indicate that an implementation activator process is valid for a specific system code. For example, you might want to suppress implementation activator processes related to Manufacturing and Distribution for those users who purchase only the OneWorld Financial system. System codes drive the rough cut process because they are closely related to the product suites.

You can also create a dependency based on industry codes and geographic regions. These dependencies indicate that an implementation activator process is valid only for a particular industry or geographic region.

You must assign a system code to an implementation activator process; assigning an industry or geographic code is optional. If you do not assign an industry or geographic region code, it implies that the implementation activator process is valid for all industries or all geographic regions.

When you create dependencies, you basically create a hierarchy of process selection and application suites, product dependencies, industry codes, and geographic regions.

Before You Begin

- Before you create dependencies with Activators Cross Reference, you must enter the following information into the user defined codes table (F0005):
 - System codes - UDC value: 98/SY
 - Industry codes - UDC value: 98/IN
 - Application lists - UDC value: 98/AL
 - Geographic regions - UDC value: H98/GR

The default values already exist, as well as the mappings. For example, suppose you want to add a new geographic region, such as Monaco. You would open the Geographic Regions UDC application and add Monaco to the H98/GR code. Monaco is now available for use in the Implementation Activator application.

NOTE: The following task describes how to create dependencies for process selection and application suites. You would follow the same steps to create dependencies for industry codes, geographic regions, and product dependencies.

► To create dependencies for an implementation activator process

From Activator Setup (G02141), choose Industry, Geographic & Process X-Ref.

1. On Activators Cross Reference, on the right side of the form, click the icon that is associated with the dependencies that you want to create. For example, click the Process Selection and Application Suites icon.

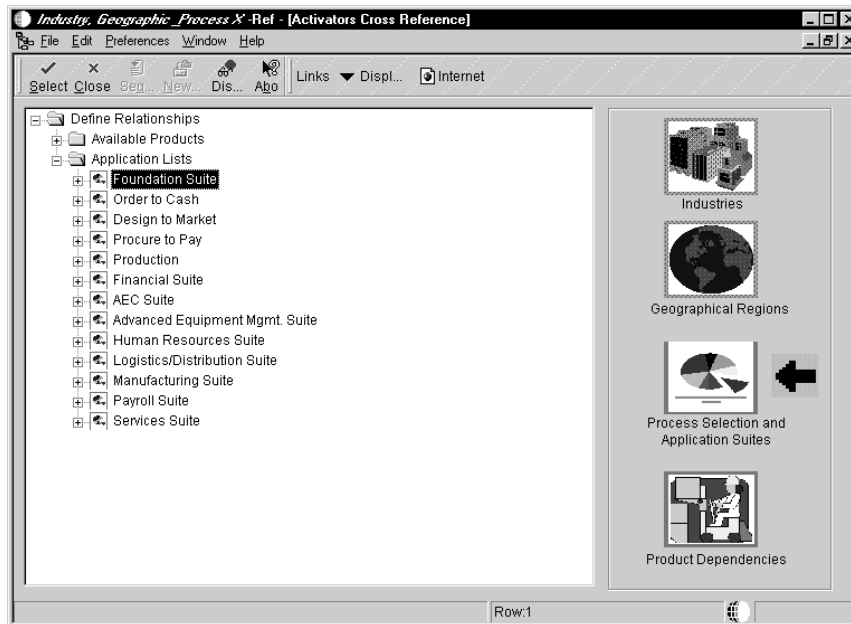
The arrow on the right moves next to the choice you made.

2. Click the plus sign next to the Define Relationships folder to expand it.

The system displays the Available Products and Application Lists folders.

3. Expand the Available Products folder and Application Lists folders by clicking the plus sign next to each one.

Any items that have not been placed under the Application Lists node appear in the Available products folder.



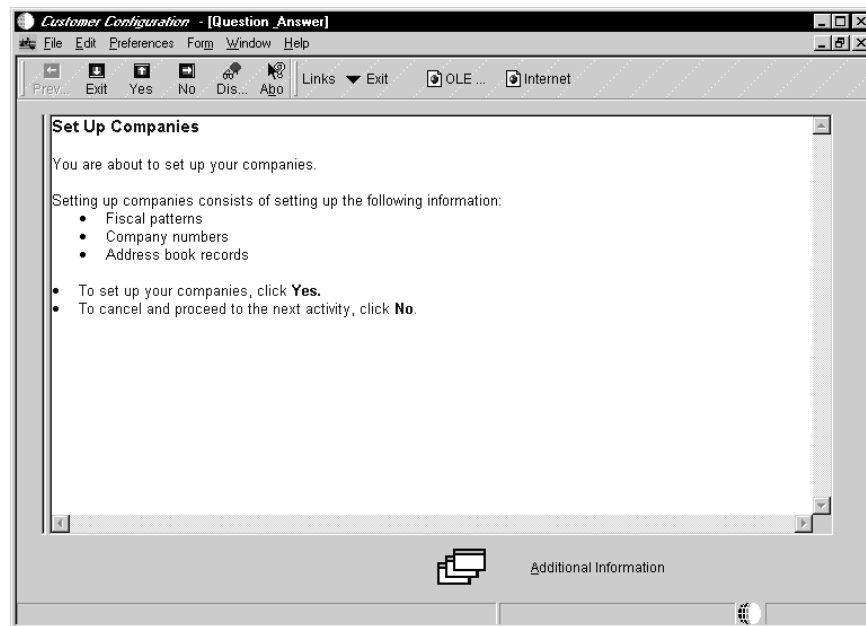
4. Drag the appropriate items from the Available Products folder to the Application Lists folder and click OK.

For example, you might want to move H08 Human Resources to the Human Resources Suite application list. To do this, drag the item from Available Products to a node below the Human Resources Suite application list.

When you drag an item from the Available Products folder to the Application Lists folder, the system records the move in the F98887 table.

Modifying Text within an Activator Question Form

During the configuration process, the system displays a question and answer form when the user chooses an activity in a process. Question and answer forms typically explain to the user what he is about to do, then ask him if he is sure he wants to do it. The Activator Question Forms are stored in OneWorld as standard application forms. The data within these forms is generic text, which is viewed through a media object control on the form.



The text within these forms is stored as a data dictionary item. You modify the text for a question and answer form through Workflow Messages in the data dictionary. After you make the change, the system displays the new text in the Implementation Activators application.

If you want to add a new data dictionary item to a form, you will have to modify the system function that calls the data dictionary item within the activity. You pass in the required data dictionary item as a parameter to the form, which displays the text to the user.



You might not have authority to modify text or add new data dictionary items. Check with your system administrator.

► **To modify text within an Activator question form**

You modify activator data item text the way you would modify the text within a workflow message data item.

From Data Dictionary Design (GH951), choose Workflow Messages.

1. On Work With Glossary Items, find the data item you want to modify and click Select.
2. Click the Glossary button.
3. On Data Item Glossary, modify the text as necessary.
4. Click OK when you have finished modifying the text.

See Also

- *Data Dictionary* in the *OneWorld Development Tools Guide*
- *Creating an Implementation Activator Process*
- *Adding Activities to an Implementation Activator Process*

Appendix

Appendix A - Example of an Implementation Activator Process Prototype

Example of an Implementation Activator Process Prototype

The following example illustrates the items you should define prior to creating implementation activator processes. Notice that the question, “Set Up Order Flows” is the first question a user will answer within this particular process, and its associated data item appears in the Data Dictionary column. If a user clicks Yes to this question, the system will launch the next question, which is “Set Up Document Types.” The system also launches form W0004A, version ZJDE0054 within application P0004A.

You might want to set up a prototype of your implementation activator processes using a spreadsheet program.

Activator Name	Activity Name	Description/Question/Activity Condition	Application	Form	Version	Data Dictionary Item	Remarks
	Q00	Set Up Order Flows				C43003010	
		Yes				C43003000	
	Q10	Set Up Document Types	P0004A	W0004A	ZJDE0054		
		Yes					
	Q20	Set Up Activity Codes				C43003020	
		Yes	P0004A	W0004AA	ZJDE0055		
	Q30	Set Up Line Types				C43004010	
		Yes	P40205	W40205A			
	Q40	Set Up Next Numbers				C43004050	
		Yes	P002	W0002A	ZJDE0001		

Glossary

action message. With OneWorld, users can receive messages (system-generated or user-generated) that have shortcuts to OneWorld forms, applications, and appropriate data. For example, if the general ledger post sends an action error message to a user, that user can access the journal entry (or entries) in error directly from the message. This is a central feature of the OneWorld workflow strategy. Action messages can originate either from OneWorld or from a third-party e-mail system.

alphabetic characters. Characters on the keyboard including letters of the alphabet and all other symbols (such as *, &, #), but excluding numerals 0 through 9. For example, "ABC*" is a string of alphabetic characters, but "ABC123" is not. Also referred to as an alpha character. Contrast with alphanumeric characters and numeric characters.

alphanumeric characters. The complete set of characters on the keyboard including letters of the alphabet, symbols, and numerals. For example, "ABC*123" is a string of alphanumeric characters. Contrast with alphabetic characters and numeric characters.

alternate language. A language other than English, which is designated in the user profile. A language preference code is used to set the preferred language for each user to display data for online and printed output.

applet. A small application, such as a utility program or a limited-function spreadsheet. It is generally associated with the programming language Java, and in this context refers to Internet-enabled applications that can be passed from a Web browser residing on a workstation.

application. A computer program or set of programs used to accomplish a task. In OneWorld, there are interactive applications and batch applications. Interactive applications are made up of a set of forms through which the user interacts with OneWorld. Interactive application identifiers begin with "P." For example, Address Book Revisions (P01012) is an interactive application. Batch applications run without user interaction. Reports and table conversions are examples of batch

applications. Batch application identifiers begin with "R." For example, the Print Mailing Labels report (R01401) is a batch application.

application programming interface (API). A software function call that can be made from a program to access functionality provided by another program.

application server. A server in a local area network (LAN) that contains applications used by network clients.

application workspace. The area on a workstation display in which all related forms within an application appear.

architecture. The underlying design of a computer that defines data storage methods, operations, and compatibility requirements with other systems and software. It also refers to specific components of a computer system, the way they interact with each other, and the type of CPU chip that is used as the basis of a computer system.

AS/400 common. The AS/400 library that typically contains WorldSoftware control files. Can also refer to as an AS/400 Common data source used in OneWorld.

AS/400 COMMON. A data source that resides on an AS/400 and holds data that is common to the coexistent library allowing OneWorld to share information with World.

asynchronous. A method of running table conversions in which starting one conversion does not rely on another conversion's successful completion.

audit trail. The detailed, verifiable history of a processed transaction. The history consists of the original documents, transaction entries, and posting of records and usually concludes with a report.

automatic accounting instruction (AAI). A code that refers to an account in the chart of accounts. AAIs define rules for programs that automatically generate journal entries, including interfaces between the Accounts Payable, Accounts Receivable, Financial Reporting, and General

Accounting systems. Each system that interfaces with the General Accounting system has AAIs. For example, AAIs can direct the General Ledger Post program to post a debit to a specific expense account and a credit to a specific accounts payable account.

base release. The first generally available software for a OneWorld release. See also release, release level.

batch. A group of similar records or transactions that the computer treats as a single unit during processing. For identification purposes, the system usually assigns each batch a unique identifier known as a batch number.

batch application. A single task or groups of tasks that the system treats as a single unit during processing. The computer performs batch applications (jobs) with little or no user interaction. Printing reports and purging files are examples.

batch control. A feature that verifies the number of transactions and the total amount in each batch that you enter into the system.

batch input. A group of transactions loaded from an external source.

batch job. A task or group of tasks you submit for processing that the system treats as a single unit during processing, for example, printing reports and purging files. The system performs a batch job with little or no user interaction.

batch processing. A method by which the system selects jobs from the job queue, processes them, and sends output to the outqueue. Contrast with interactive processing.

batch server. A server on which OneWorld batch processing requests (also called UBEs) are run instead of on a client, an application server, or an enterprise server. A batch server typically does not contain a database nor does it run interactive applications.

batch type. A code assigned to a batch job that designates to which system the associated transactions pertain, thus controlling which records are selected for processing. For example, the Post General Journal program selects for posting only unposted transaction batches with a batch type of O.

batch-of-one immediate. A transaction method that allows a client application to perform work on a client workstation, then submit the work all at once to a server application for further processing. As a batch process is running on the server, the client

application can continue performing other tasks. See also direct connect, store-and-forward.

binary large object (BLOB). A database field that has no maximum size limit and holds any digitized information. This field is often used to store objects, such as graphical representations or data structures, rather than standard alphanumeric data.

binary string (BSTR). A length prefixed string used by OLE automation data manipulation functions. Binary strings are wide, double-byte (Unicode) strings on 32-bit Windows platforms.

broadcast message. 1) An e-mail message that you send to multiple recipients. 2) A message that appears on a form instead of in your mailbox.

browser. A client application that translates information sent by the Worldwide Web. A client must use a browser to receive, manipulate, and display Worldwide Web information on the desktop. Also known as a Web browser.

business function. An encapsulated set of business rules and logic that can normally be reused by multiple applications. Business functions can execute a transaction or a subset of a transaction (check inventory, issue work orders, and so on). Business functions also contain the APIs that allow them to be called from a form, a database trigger, or a non-OneWorld application. Business functions can be combined with other business functions, forms, event rules, and other components to make up an application. Business functions can be created through event rules or third-generation languages, such as C. Examples of business functions include Credit Check and Item Availability.

business function event rule. Encapsulated, reusable business logic created using through event rules rather than C programming. Contrast with embedded event rule. See also event rule.

business view. Used by OneWorld applications to access data from database tables. A business view is a means for selecting specific columns from one or more tables whose data will be used in an application or report. It does not select specific rows and does not contain any physical data. It is strictly a view through which data can be handled.

Business View Design Aid (BDA). A OneWorld GUI tool for creating, modifying, copying, and printing business views. The tool uses a graphical user interface.

category code. A type of user defined code for which you can provide the title. For example, if you were adding a code that designated different sales regions, you could change category code 4 to Sales Region, and define E (East), W (West), N (North), and S (South) as the valid codes. Sometimes referred to as reporting codes. See also user defined code.

central objects. Objects that reside in a central location and consist of two parts: the central objects data source and central C components. The central objects data source contains OneWorld specifications, which are stored in a relational database. Central C components contain business function source, header, object, library, and DLL files and are usually stored in directories on the deployment server. Together they make up central objects.

Central Objects merge. A process that blends a customer's modifications to the objects in a current release with objects in a new release.

central processing unit (CPU). Computer component which carries out the logic, calculation and decision-making functions. Interprets and executes instructions upon receipt.

character conversion. The process of converting characters of the same language from one encoding scheme to another while sending and receiving data in a heterogeneous environment.

character set. An ordered set of characters representing any particular language.

chart. OneWorld term for tables of information that appear on forms in the software. See forms.

chart of accounts. The structure for general ledger accounts. The chart of accounts lists types of accounts, describes each account, and includes account numbers and posting edit codes.

check-in location. The directory structure location for the package and its set of replicated objects. This is usually \\deploymentserver\release\path_code\package\packagename. The subdirectories under this path are where the central C components (source, include, object, library, and DLL file) for business functions are stored.

child. See parent/child form.

client. 1) A workstation or PC in a client/server environment. 2) The receiving end of the spectrum in a request/supply relationship between programs.

client workstation. The computer on which a user operates OneWorld software applications.

client/server. A relationship between processes running on separate machines. The server process is a provider of software services. The client is a consumer of those services. In essence, client/server provides a clean separation of function based on the idea of service. A server can service many clients at the same time and regulate their access to shared resources. There is a many-to-one relationship between clients and a server, respectively. Clients always initiate the dialog by requesting a service. Servers passively wait for requests from clients.

cluster. A group of two or more servers with identical configurations used to provide protection against failure. If one server fails, the other can continue processing.

code page. An ordered set of characters in which an alphanumeric value (code point) is associated with each character set.

code point. The numeric identifier assigned to a character. Its value is usually expressed in a hexadecimal notation.

coexistence. An AS/400 configuration of J.D. Edwards software that allows shared data interface operation to occur between OneWorld and WorldSoftware.

commit. A process that ensures that all database changes for a single transaction occur simultaneously. The changes are treated as a single unit; either all changes occur or none of the changes occur, thereby maintaining the integrity of the database.

common object request broker architecture. An object request broker standard endorsed by the Object Management Group.

component object model (COM). A specification developed by Microsoft for building software components that can be assembled into programs or add functionality to existing programs running on Microsoft Windows platforms. COM components can be written in a variety of languages, although most are written in C++, and can be unplugged from a program at runtime without having to recompile the program.

Conference Room Pilot environment. A OneWorld environment used as a staging environment for production data, which includes constants and masters tables, such as company

constants, fiscal date patterns, and item master. Use this environment along with the test environment to make sure your configuration works before you release changes to end-users.

configurable client engine. Allows user flexibility at the interface level. Users can easily move columns, set tabs for different data views, and size grids according to their needs. The configurable client engine also enables the incorporation of Web browsers in addition to the Windows 95- and Windows NT-based interfaces.

Configurable Network Computing (CNC). An application architecture that allows interactive and batch applications, composed of a single code base, to run across a TCP/IP network of multiple server platforms and SQL databases. The applications consist of reusable business functions and associated data that can be configured across the network dynamically. The overall objective for businesses is to provide a future-proof environment that enables them to change organizational structures, business processes, and technologies independently of each other.

constants. Parameters or codes that you set and that the system uses to standardize the processing of information by associated programs.

control. Any data entry point allowing the user to interact with an application. For example, check boxes, pull-down lists, hyper-buttons, entry fields, and similar features are controls.

Control Table Workbench. During the Installation Workbench process, Control Table Workbench runs the batch applications for the planned merges that update the data dictionary, user defined codes, menus, and user overrides tables.

Control Tables merge. A process that blends a customer's modifications to the control tables with the data that accompanies a new release.

cumulative update. A version of OneWorld software that includes fixes and enhancements made since the last release or update.

custom gridlines. A grid row that does not come from the database, for example, totals. To display a total in a grid, sum the values and insert a custom gridline to display the total. Use the system function Insert Grid Row Buffer to accomplish this.

custom installation. One of the two types of installations you can set up in the Installation Planner application. A custom installation gives the

customer flexibility in creating a plan with Java and Windows terminal servers, custom environments, and custom data sources. See also typical installation.

custom modifications. Changes customers make to OneWorld to make the software run more efficiently for them or to meet their particular requirements.

customer. The company that purchases and uses OneWorld. A customer contains individual users.

data dictionary. A database table that OneWorld uses to manage the definitions, structures, and guidelines for the usage of fields, messages, and help text. J.D. Edwards has an active data dictionary, which means that it is accessed at runtime.

Data Dictionary merge. A process that updates a customer's data dictionary tables with the data that accompanies a new release.

data mart. Department-level decision support databases. They usually draw their data from an enterprise data warehouse that serves as a source of consolidated and reconciled data from around the organization. Data marts can be either relational or multidimensional databases.

data only upgrade. A process that preserves customer business data when moving from a previous release of OneWorld to a new release. This shortens the upgrade process by eliminating the need to perform the merges and table conversions that incorporate J.D. Edwards data into a customer's existing data.

data replication. In a replicated environment, multiple copies of data are maintained on multiple machines. There must be a single source that "owns" the data. This ensures that the latest copy of data can be applied to a primary place and then replicated as appropriate. This is in contrast to a simple copying of data, where the copy is not maintained from a central location, but exists independently of the source.

data server. A machine required for AS/400 users who need to put central objects in SQL Server or Oracle. Putting central objects on a data server tells OneWorld they are not on the enterprise server.

data source. A specific instance of a database management system running on a computer. Data source management is accomplished through Object Configuration Manager (OCM) and Object Map (OM).

Data Source Workbench. During the Installation Workbench process, Data Source Workbench copies all data sources that are defined in the installation plan from the Data Source Master and Table and Data Source Sizing tables in the Planner data source to the System – release number data source. It also updates the Data Source Plan detail record to reflect completion.

data structure. A group of data items that can be used for passing information between objects, for example, between two forms, between forms and business functions, or between reports and business functions.

data warehouse. A database used for reconciling and consolidating data from multiple databases before it is distributed to data marts for department-level decision support queries and reports. The data warehouse is generally a large relational database residing on a dedicated server between operational databases and the data marts.

database. A continuously updated collection of all information that a system uses and stores. Databases make it possible to create, store, index, and cross-reference information online.

database administrator. The person who has special skills or training in one or more types of database software, for example, SQL Server or Oracle.

database driver. Software that connects an application to a specific database management system.

database management system (DBMS). A computer program that manages data by providing centralized control, data independence, and complex physical structures for efficient access, integrity, recovery, concurrence, control, privacy, and security.

database server. A server that stores data. A database server does not have OneWorld logic.

default. A code, number, or parameter that the system supplies when the user does not specify one.

default. A value that the system assigns when the user does not enter a value. For example, if the default value for an input is N and nothing is entered in that field, the system assumes the default is an N value.

deployment environment. A OneWorld environment used to run OneWorld on the deployment server. This environment administers

information for the system data source, such as user profiles, packages, and environments.

deployment server. The computer used to install, maintain, and distribute OneWorld software to one or more enterprise servers and client workstations.

detail. The specific information that makes up a record or transaction. Contrast with summary.

detail area. An area of a form that displays detailed information associated with the records or data items displayed on the form. See also grid.

development environment. A OneWorld environment used to test modified development objects before transferring them to the conference room pilot environment.

direct connect. A transaction method in which a client application communicates interactively and directly with a server application. See also batch-of-one immediate, store-and-forward.

director. A OneWorld user interface that guides a user interactively through a OneWorld process.

disk. A direct access storage device.

distributed computing environment (DCE). A set of integrated software services that allows software running on multiple computers to perform in a manner that is seamless and transparent to the end-users. DCE provides security, directory, time, remote procedure calls, and files across computers running on a network.

distributed database management system (DDBMS). A system for distributing a database and its control system across many geographically dispersed machines.

Do Not Translate (DNT). A type of data source that must exist on the AS/400 because of BLOB restrictions.

double-byte character set (DBCS). A method of representing some characters using one byte and other characters using two bytes. Double-byte character sets are necessary to represent some characters in the a Japanese, Korean, and Chinese languages.

double-byte enabled. A data storage feature that allows a computer to store ideographic characters from Asian languages. J.D. Edwards coding techniques accommodate both ideographic and alphabetic characters, making it easier to translate an application into another language.

driver. A program or portion of a program that controls the transfer of data from an input or output device.

duplicated database. A decision support database that contains a straightforward copy of operational data. The advantages involve improved performance for both operational and reporting environments. See also enhanced analysis database, enterprise data warehouse.

dynamic link library (DLL). A set of program modules that are designed to be invoked from executable files when the executable files are run, without having to be linked to the executable files. They typically contain commonly used functions.

dynamic partitioning. The ability to dynamically distribute logic or data to multiple tiers in a client/server architecture.

Electronic Data Interchange (EDI). The paperless, computer-to-computer exchange of business transactions, such as purchase orders and invoices, in a standard format with standard content.

embedded event rule. An event rule that is specific to a particular table or application. Examples include form-to-form calls, hiding a field based on a processing option value, and calling a business function. Contrast with business function event rule. See also event rule.

Employee Work Center. This is a central location for sending and receiving all OneWorld messages (system and user generated) regardless of the originating application or user. Each user has a mailbox that contains workflow and other messages, including Active Messages. With respect to workflow, the Message Center is MAPI compliant and supports drag and drop work reassignment, escalation, forward and reply, and workflow monitoring. All messages from the message center can be viewed through OneWorld messages or Microsoft Exchange.

encapsulation. The ability to confine access to and manipulation of data within an object to the procedures that contribute to the definition of that object.

end user. An individual who uses OneWorld software.

enhanced analysis database. A database containing a subset of operational data. The data on the enhanced analysis database performs calculations and provides summary data to speed generation of

reports and query response times. This solution is appropriate when external data must be added to source data, or when historical data is necessary for trend analysis or regulatory reporting. See also duplicated database, enterprise data warehouse.

enterprise. Every server, PC, and database that is on an organization's network.

enterprise data warehouse. A complex solution that involves data from many areas of the enterprise. This environment requires a large relational database (the data warehouse) that is a central repository of enterprise data, which is clean, reconciled, and consolidated. From this repository, data marts retrieve data to provide department-level decisions. See also duplicated database, enhanced analysis database.

enterprise server. A database server and logic server. See database server. Also referred to as host.

Enterprise Workflow Management. A OneWorld system that provides a way of automating tasks, such as notifying a manager that a requisition is waiting for approval, using an e-mail-based process flow across a network.

environment. A path code with a set of Object Configuration Manager (OCM) mappings that allow a user to locate data and a specific set of objects. Examples include Conference Room Pilot (CRP), development, production, and pristine.

Environment Checker. An application you can run before installing or upgrading OneWorld to diagnose configuration and setup issues that may exist at the operating system level.

Environment Workbench. During the Installation Workbench process, Environment Workbench copies the environment information and Object Configuration Manager tables for each environment from the Planner data source to the System release number data source. It also updates the Environment Plan detail record to reflect completion.

Ethernet. A commonly used, shared media LAN technology, which broadcasts messages to all nodes on the network segment. Ethernet connects up to 1,024 nodes at 10MB per second over twisted pair cable, coaxial cable, and optical fiber.

event. An action that occurs when an interactive or batch application is running. Example events are tabbing out of an edit control, clicking a push button, initializing a form, or performing a page break on a report. The GUI operating system uses

miniprograms to manage user activities within a form. Additional logic can be attached to these miniprograms and used to give greater functionality to any event within a OneWorld application or report using event rules.

event rule. Used to create complex business logic without the difficult syntax that comes with many programming languages. These logic statements can be attached to applications or database events and are executed when the defined event occurs, such as entering a form, selecting a menu bar option, page breaking on a report, or selecting a record. An event rule can validate data, send a message to a user, call a business function, as well as many other actions. There are two types of event rules:

- 1 Embedded event rules.
- 2 Business function event rules.

executable file.

A computer program that can be run from the computer's operating system. Equivalent terms are "application" and "program."

facility. An entity within a business for which you want to track costs. For example, a facility might be a warehouse location, job, project, work center, or branch/plant. Sometimes referred to as a business unit.

field. 1) An area on a form that represents a particular type of information, such as name, document type, or amount. 2) A defined area within a record that contains a specific piece of information. For example, a supplier record consists of the fields Supplier Name, Address, and Telephone Number.

file. A set of information stored under one name. See also table.

file transfer protocol (FTP). A set of TCP/IP commands used to log on to a network, list directories, and copy files. FTP is also a communications protocol used to transmit files without data loss.

find/browse. A type of form used to:

- 1 Search, view, and select multiple records in a detail area.
- 2 Delete records.
- 3 Exit to another form.
- 4 Serve as an entry point for most applications.

firewall.

A set of technologies that allows an enterprise to test, filter, and route all incoming messages. Firewalls are used to keep an enterprise secure.

fix/inspect. A type of form used to view, add, or modify existing records. A fix/inspect form has no detail area.

form. The element of the OneWorld graphical user interface by which the user exchanges data with interactive applications. Forms are made up of controls, such as fields, options, and the grid. These controls allow the user to retrieve information, add and revise information, and navigate through an application to accomplish a task.

Form Design Aid (FDA). The OneWorld GUI development tool for building interactive applications and forms.

form interconnection. Allows one form to access and pass data to another form. Form interconnections can be attached to any event; however, they are normally used when a button is clicked.

form type. The following form types are available in OneWorld:

- 1 Find/browse.
- 2 Fix/inspect.
- 3 Header detail.
- 4 Headerless detail.
- 5 Message.
- 6 Parent/child.
- 7 Search/select.

fourth generation language (4GL).

A programming language that focuses on what you need to do and then determines how to do it. Structured Query Language is an example of a 4GL.

general release. See release.

generic text structures. See Media Storage Objects.

graphical user interface (GUI). A computer interface that is graphically based as opposed to being character-based. An example of a character-based interface is that of the AS/400. An example of a GUI is Microsoft Windows. Graphically based interfaces allow pictures and other graphic images to be used in order to give people clues on how to operate the computer.

grid. A control that displays detail information on a form. The grid is arranged into rows, which generally represent records of data, and columns,

which generally represent fields of the record. See also detail area.

header. Information at the beginning of a table or form. Header information is used to identify or provide control information for the group of records that follows.

header detail. A type of form used to add, modify, or delete records from two different tables. The tables usually have a parent/child relationship.

headerless detail. A type of form used to work with multiple records in a detail area. The detail area is capable of 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 nonexistent 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

A

Accessing computer based training (CBT), 3–7
Action activity
 adding, 4–16
 naming conventions, 4–2
Activating implementation activators, 4–33
Activator processes
 creating, 4–5
 creating dependencies for, 4–37
 defining the flow, 4–23
Activator question forms, modifying text within, 4–41
Activators, implementation
 activating or deactivating, 4–33
 modifying, 4–33
 setup, 4–1
Activators Cross Reference, forms, 4–38
ActiveEra Implementation Activators Guide, who should use it, 1–2
ActiveEra implementation activators overview, 1–1
Activities
 defining the flow, 4–23
 flow, example, 4–27
 process activity, creating, 4–19
 understanding the flow, 4–25
Activity conditions
 adding, 4–23
 for implementation activators, 4–23
Activity names, naming conventions, 4–2
Activity Revisions, forms, 4–10, 4–13, 4–16, 4–19
Adding a completion activity, 4–13
Adding activities to an implementation activator, 4–9
Adding activity conditions, 4–23
Adding additional information to an activator, 3–9
Adding an action activity, 4–16
Adding an attachment to an implementation activator, 3–7

Adding supplemental information to an implementation activator, 4–34
Additional information, adding to an activator, 3–9
Application Suite Selection, forms, 2–5

C

CBT. *See* Computer based training
Completion activity, adding, 4–13
Completion form activity, naming conventions, 4–2
Completion Options, forms, 3–6
Computer based training, accessing, 3–7
Creating a rough cut customer configuration, 2–3
Creating Activator processes, 4–5
Creating an interactive activity, 4–9
Customer Configuration, forms, 3–5, 3–11
Customer Rough Cut, forms, 2–9

D

Data dictionary items, modifying text within, 4–41
Data items, naming conventions, 4–2
Deactivating implementation activators, 4–33
Define Conditional Rule Paths, forms, 4–29
Defining the flow of activator processes and activities, 4–23
Dependencies, creating for activator processes, 4–37

E

Example of an implementation activator process prototype, A–3

F

Flow of activities

- example, 4-27
- special cases, 4-25
- understanding, 4-25

Form Interconnections, forms, 4-12, 4-15, 4-18

Forms

- Activators Cross Reference, 4-38
- Activity Revisions, 4-10, 4-13, 4-16, 4-19
- Application Suite Selection, 2-5
- Completion Options, 3-6
- Customer Configuration, 3-5, 3-11
- Customer Rough Cut, 2-9
- Define Conditional Rule Paths, 4-29
- Form Interconnections, 4-12, 4-15, 4-18
- Geographical Region Revisions, 2-7
- Implementation Activator Generation, 2-8
- Implementation Activators, 3-4
- Industry Revisions, 2-7
- Media Objects, 3-8
- Process Revisions, 4-6
- Product Selections, 2-6
- Question and Answer, 3-5
- Subprocess Activity Definition, 4-20
- Work With Applications, 4-11, 4-14, 4-17
- Work With Forms, 4-12, 4-15, 4-17

G

Geographical Region Revisions, forms, 2-7

Glossary activity, adding, 4-10

Glossary items, defining before you add them to an activator question form, 4-3

I

Implementation Activator Generation, forms, 2-8

Implementation activator process, example of prototype, A-3

Implementation activator process flows, prototyping, 4-3

Implementation activator setup, 4-1

Implementation Activators, forms, 3-4

Implementation activators

- activating or deactivating, 4-33
- adding activities to, 4-9
- adding attachments to, 3-7
- benefits of, 1-1
- naming conventions, 4-1
- using, 3-3

Industry Revisions, forms, 2-7

Interactive application, activities, creating, 4-9

M

Media Objects, forms, 3-8

Media objects, working with, 3-7

Modifying implementation activators, 4-33

Modifying text within an activator question form, 4-41

N

Naming conventions

- action activity, 4-2
- activity names, 4-2
- completion form activity, 4-2
- data items, 4-2
- implementation activators, 4-1
- process confirm activity, 4-2
- question activity, 4-2

Naming conventions used in implementation activators, 4-1

O

Overview, ActivEra implementation activators, 1-1

P

Process confirm activity, Naming conventions, 4-2

Process Revisions, forms, 4-6

Processes

- Activator, creating, 4-5
- activator, creating dependencies for, 4-37

starting implementation activators, 3–3
Product Selections, forms, 2–6
Prototyping implementation activators, 4–3

Working with media objects, 3–7

Q

Question activity, naming conventions, 4–2
Question and Answer, forms, 3–5

R

Reasons, reviewing for an implementation activator, 3–10
Reviewing a reason for an implementation activator, 3–10
Rough cut customer configuration, creating, 2–3

S

Starting implementation activators, 3–3
Subprocess Activity Definition, forms, 4–20
Subprocesses, creating, 4–19
Supplemental information, adding to an implementation activator, 4–34

T

Text, modifying text within an activator question form, 4–41

U

Understanding the flow of activities, 4–25
Using implementation activators, 3–3

W

Work With Applications, forms, 4–11, 4–14, 4–17
Work With Forms, forms, 4–12, 4–15, 4–17

