PeopleSoft®

EnterpriseOne Xe
Scripting Tool
PeopleBook

September 2000

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

SKU XeEAUA

Table of Contents

OneWorld Scripting Tool User Interface

Scripting Context

Scripting Actions

Script Pane

Play Back

Sample Script

Storing Scripts and Test Results

Glossary

Index

OneWorld Scripting Tool Overview

OneWorld Scripting Tool is an automated testing tool you can use to create scripts to test the execution of OneWorld applications and to perform repetitive tasks, such as loading data, entering sales orders, or performing screen captures. You create the scripts by using OneWorld Scripting Tool to write and insert to your script commands that run essential OneWorld functions and processes such as:

- Launching applications
- Launching forms
- Executing form interconnections
- Running UBEs
- Setting processing options for interactive applications and for UBEs
- Entering data to header controls
- Entering data to grid columns
- Entering data to QBE lines
- Pushing toolbar buttons
- Clicking pushbuttons and bitmaps
- Selecting grid lines
- Performing database validations
- Selecting combo boxes
- Traversing tree paths

The hallmark of OneWorld Scripting Tool is its flexibility. OneWorld Scripting Tool scripts will run on either a OneWorld Windows NT platform or on a OneWorld Java platform. The tool has this flexibility because it reads and loads the specifications for each OneWorld operation that you perform and passes the data through the operating system to OneWorld as a keyboard input. Therefore, you can use the script to test different operating systems, environments, and data mappings without making changes to the script.

OneWorld Scripting Tool's flexibility also allows you to:

- Save scripts on your local drive or in a script repository shared by others

- Run scripts in a stand-alone mode, or with included scripts
- Derive values for input in your scripts from a variety of sources, including literal values, valid value lists, visual assists, and variables, which you create to store values that you can easily change as needed
- Pass variable values within a single script or between multiple scripts
- Test the integrity of data that you add by performing a database validation
- Capture data about OneWorld error and warning messages and about script playback events, including API calls
- Send scripts to and receive scripts from others

In sum, OneWorld Scripting Tool automated testing tool offers several advantages to those wanting to write scripts that test key business processes:

- Decreases the time and effort required to create automated test scripts
- Ensures that the tool will remain viable despite changes in OneWorld because it reads and loads OneWorld specifications directly
- Allows users to write scripts that are compatible with future releases of OneWorld
- Presents a user interface that disguises the complexity of the tool's inner workings
- Provides the user flexibility to customize scripts by changing, for example, Object Configuration Manager (OCM) mappings
- Possesses the ability to work successfully with changing technologies

OneWorld Scripting Tool is designed to allow easy user interaction. It appears as a form with the following components, each of which play a role in helping you to create scripts:

- Command pane, where you make choices to define the processes that you want to run in OneWorld, such as launching an application
- Insert button, which allows you to insert a command to your script and create a script object that defines what the command does
- Script pane, which contains a running log of the commands that you have inserted to your script
- Cool bar, which allows you to navigate the OneWorld Scripting Tool form and to resize it to your specifications
- Menu bar, which contains the options you need to run OneWorld Scripting Tool
- Caption bar, which identifies the script on which you are working
- Status bar, which displays information about a OneWorld Scripting Tool session, including processes that OneWorld Scripting Tool is running and brief definitions of OneWorld Scripting Tool commands

Before You Begin

- ❑ You should have a working knowledge of common OneWorld concepts, which you can find in the *OneWorld Foundation Guide*.
- ❑ You should also have a good understanding of at least one OneWorld application, such as Accounts Payable or Sales Order Entry.

Upgrading from Service Pack 13 (OneWorld Xe)

If you have a previous version of OneWorld Scripting Tool (known as AutoPilot), and you upgrade to OneWorld Service Pack 13 (OneWorld Xe), you should uninstall AutoPilot by deleting all the files in your AutoPilot directory. Do not delete the directory itself, since it contains all of your scripts, valid value lists, and so on. After you upgrade to SP13, change your desktop icon so that it points to *X*: \b7\system\bin32\autopilot.exe, where *X* is the drive on which you have OneWorld Xe installed.

Your Open Database Connectivity (ODBC) settings should remain valid after you upgrade. If you need to reconfigure your ODBC settings for OneWorld Scripting Tool, see *OneWorld Installation/Upgrade Guide*.

This guide discusses the following topics:

- ❑ OneWorld Scripting Tool user interface
- ❑ Scripting context
- ❑ Scripting actions
- ❑ Working with the script pane
- ❑ Playing back the script
- ❑ Creating a sample OneWorld Scripting Tool script
- ❑ Storing scripts and test results

OneWorld Scripting Tool User Interface

OneWorld Scripting Tool User Interface

You work with panes and bars in the OneWorld Scripting Tool form to write commands that make a script. The form consists of two panes: the command pane and the script pane. The command pane is the area in which you make choices that create commands. As you make the choices and insert them to create a script, OneWorld Scripting Tool displays the script as command lines in the script pane, where you can move, delete, and edit commands. The form also contains four bars, the caption bar, menu bar, cool bar, and status bar, all of which assist you in creating and identifying the script.

As you work in the OneWorld Scripting Tool form, you can also change its shape, size, and location on the desktop for ease of use. If you are working with more than one script, you can arrange child forms within the parent form by clicking options in the menu bar. Finally, you can move the cool bar to the most convenient position within the form, or you can detach the cool bar and move it to any position within the form or move it to your desktop.

This section discusses the following topics:

❑ Opening the OneWorld Scripting Tool form

❑ Panes in the OneWorld Scripting Tool form

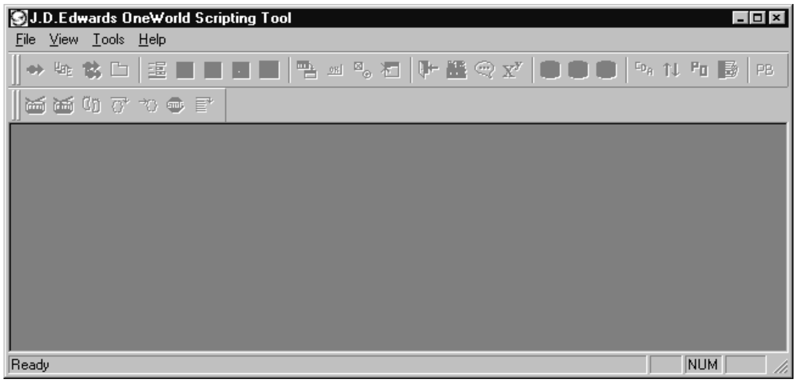
❑ Bars in the OneWorld Scripting Tool form

❑ Manipulating the OneWorld Scripting Tool form

❑ Manipulating the OneWorld Scripting Tool cool bar

Opening the OneWorld Scripting Tool Form

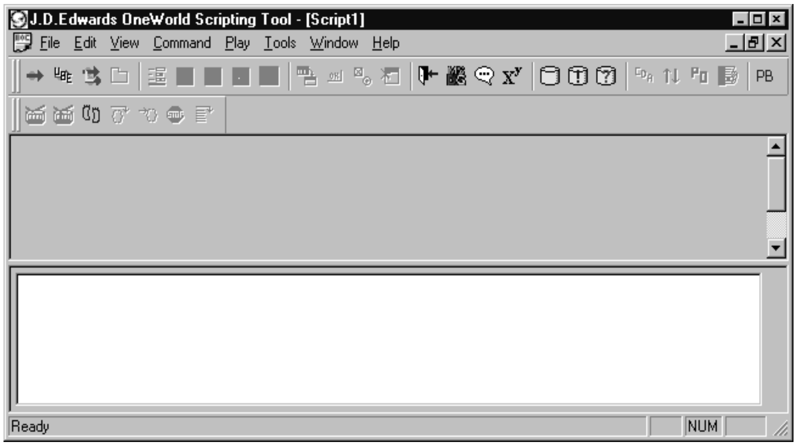
When you click the OneWorld Scripting Tool desktop icon, a splash screen appears, followed by the OneWorld Scripting Tool form. Using this form, you create scripts to test OneWorld applications and carry out repetitive tasks. The OneWorld Scripting Tool form initially is blank.



▶ To open a OneWorld Scripting Tool form for scripting

1. From your desktop or the appropriate directory, launch OneWorld Scripting Tool.
2. Choose File from the OneWorld Scripting Tool menu bar.
3. Click New.

   The OneWorld Scripting Tool form, with some of the cool bar buttons activated, appears.


J.D.Edwards OneWorld Scripting Tool - [Script1]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

The cool bar, which is located directly beneath the menu bar, contains buttons that represent the various commands, such as Application, that you can run in OneWorld Scripting Tool. When you pass the cursor over one of these buttons, or over one of the names in the drop-down menu under Command in the menu bar, words that identify the command appear in the status bar, located at the bottom of the OneWorld Scripting Tool form.

When you place the cursor arrow on the splitter bar, you can change the size of the command or script pane by holding down the mouse button and pulling the bar up or down. Notice that when you initiate a OneWorld Scripting Tool session, the command pane is blank. You make the command pane active by initiating a command, such as Application. You find the names of the commands by clicking Command in the menu bar.

Panes in the OneWorld Scripting Tool Form

The two major components of the OneWorld Scripting Tool form are the command pane and the script pane. In the command pane, you make the choices that define a particular script of commands to be run in OneWorld. The commands that you insert appear sequentially as command lines in the script pane. During or after script creation you can move, edit, or delete the command lines that you have inserted to the script.

Each of the panes has specific components that you use to accomplish script-writing tasks:

Component	**Pane**	**Purpose**
Lists	Command	Make choices from or type entries in these populated and unpopulated areas to define a command to be run in OneWorld.
Insert button	Command	Click this button to insert a command to the script.
Insertion cursor	Script	Point this red arrow to any spot in the script at which you want to insert a new command.

The command pane is the top pane of the form. It is divided into lists, from which you make choices that create commands to be run in OneWorld. The command pane also contains the Insert button, which you click to insert a command to the script.

The script pane is the bottom pane of the form. It displays a running log and detailed description of the commands you insert in the script. From the script pane, you can point the insertion cursor, which appears as a red arrow, to any spot in the script you wish to insert a new command. You can also reorder the script using the mouse to drag and drop command lines, and can edit command lines by using the mouse to highlight them.

This chapter discusses the following topics:

- ❑ The command pane
- ❑ The script pane

The Command Pane

The command pane is the area where you begin writing commands to create your OneWorld Scripting Tool script. You begin the command-writing process by clicking a command in the Command menu of the menu bar or by clicking buttons in the cool bar. Once you have done so, distinct list areas appear in the command pane.

Note that neither lists nor the Insert button appear until you click a command. You make choices from or entries to the command pane lists. When you click the Insert button, a command line appears in the script.

The command pane may also contain options. When you click Application in the command menu, for example, the command pane contains options for Use Default Form and Processing Options Only. When you click Select Grid Line in the command menu, the command pane contains options that allow you to script single clicking or double clicking a grid row in a OneWorld form.

This topic discusses the two main components of the command pane:

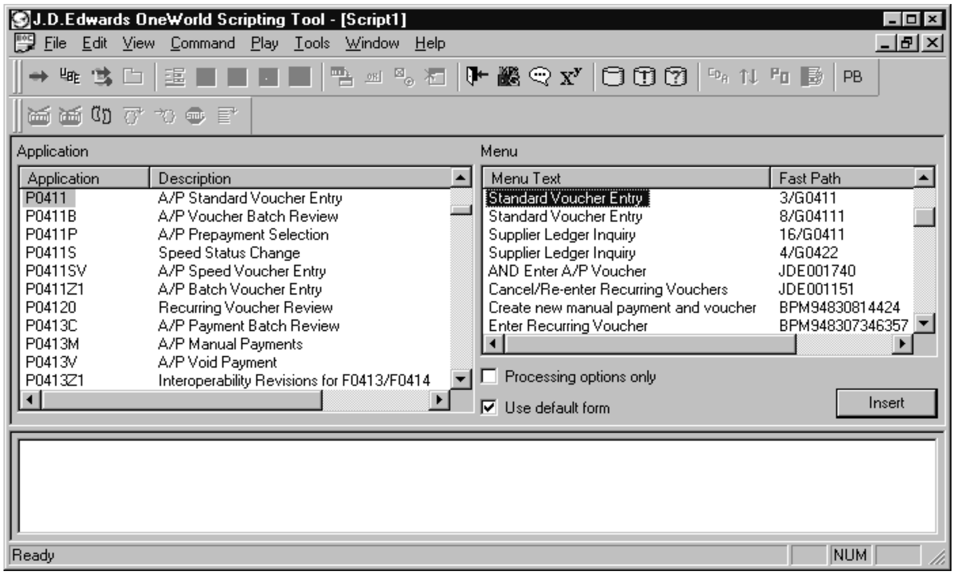
- ❑ Command pane lists
- ❑ Insert button

Command Pane Lists

Lists are distinct areas in the command pane, physically separated from one another and individually captioned. You make entries to or choices from these lists in order to write the commands that you insert to your script. A command pane list can be either populated or unpopulated. You make choices from populated lists and make inputs yourself to unpopulated lists. In either case, however, you use the lists to write script commands.

You work with both populated and unpopulated lists in the OneWorld Scripting Tool command pane. Populated lists in the command pane contain the specific items you choose to create a script command. For example, when you click Application in the command menu or click Launch Application in the cool bar, OneWorld Scripting Tool displays OneWorld applications and descriptions of each application in an Applications list in the command pane.

OneWorld Scripting Tool populates a second list, the Menu list, when you click Launch Application. It displays menu text, or descriptions, of OneWorld forms and interactive versions attached to the menu selections. Versions indicate that processing options exist for the application. The list also displays the Fast Path to the OneWorld form.


J.D.Edwards OneWorld Scripting Tool - [Script1]
File Edit View Command Play Tools Window Help
PB
Application
Menu
Insert
Ready
NUM

Application	Description
P0411	A/P Standard Voucher Entry
P0411B	A/P Voucher Batch Review
P0411P	A/P Prepayment Selection
P0411S	Speed Status Change
P0411SV	A/P Speed Voucher Entry
P0411Z1	A/P Batch Voucher Entry
P04120	Recurring Voucher Review
P0413C	A/P Payment Batch Review
P0413M	A/P Manual Payments
P0413V	A/P Void Payment
P0413Z1	Interoperability Revisions for F0413/F0414

Menu Text	Fast Path
Standard Voucher Entry	3/G0411
Standard Voucher Entry	8/G04111
Supplier Ledger Inquiry	16/G0411
Supplier Ledger Inquiry	4/G0422
AND Enter A/P Voucher	JDE001740
Cancel/Re-enter Recurring Vouchers	JDE001151
Create new manual payment and voucher	BPM94830814424
Enter Recurring Voucher	BPM948307346357

☐ Processing options only

☑ Use default form

As you write the script, the lists in the command pane change to reflect selections you make in the menu bar or the cool bar. Other populated lists might include:

- Names of header controls, grid columns, forms, forms that appear next when you add a form or interconnect to another application, buttons, previously declared variables, previously declared validations, combo box items, or options (such as radio buttons and check boxes) found in OneWorld forms
- Names of processing options for OneWorld applications
- Sources of input to OneWorld forms: literal value, UDC visual assists, valid values lists, variables, form interconnect visual assists, header controls, or grid columns
- Sources of row numbers in a OneWorld form: literal values, valid values lists or variables
- Values to be input to OneWorld forms, which can be derived from an existing valid values list, variable, header control, or grid column
- Sources from which a repeat count value in the script can be defined

Unpopulated lists appear with a caption, but they are empty. You create or modify the script command by typing words, numbers, special characters, spaces, or a combination thereof in the list.

You can enter the following to unpopulated lists:

- Literal values to be input to header controls, grid columns, or a QBE line

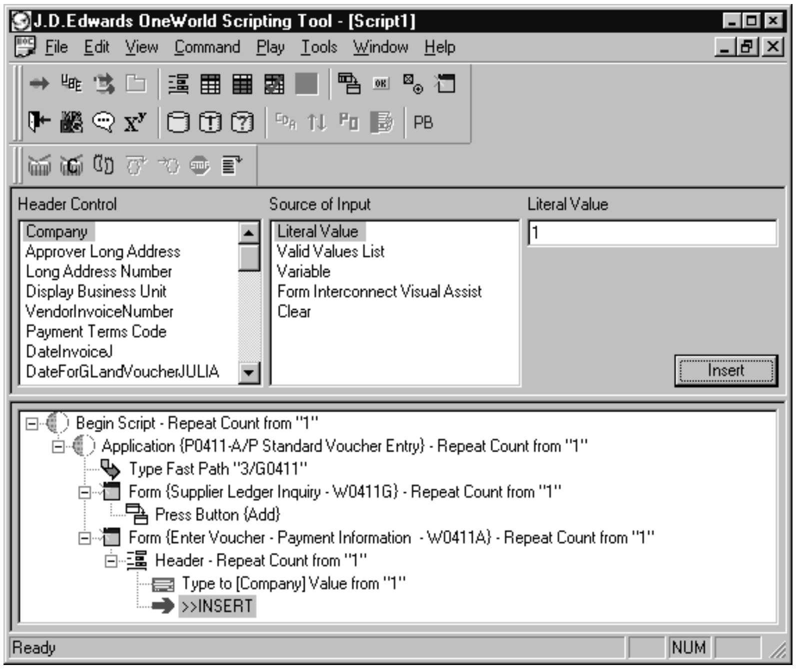
- The name of a variable or validation that you are declaring
- The repeat count for a node in the script, which controls how many times the node, or tree control of commands, plays when you run a script
- The length of a wait period during script playback
- Comments to be inserted to the script
- A DOS command line message to the system
- A tree path that identifies a unique path to a node in a OneWorld form

Insert Button

When you make choices and entries to lists in the command pane and click the Insert button, OneWorld Scripting Tool inserts a command line in the script pane. Each inserted command becomes a part of the script that OneWorld Scripting Tool runs in OneWorld. The insertion cursor, which appears as an arrow in the script pane, follows the last command you insert.

When you choose a OneWorld application and version from the command pane and click the Insert button, you automatically launch OneWorld if you have turned on the Playback button (identified by the initials PB) in the cool bar.

Clicking the Insert button the first time commences the script. As you write the script by inserting new commands, OneWorld Scripting Tool continues to display all scripted commands, in the order of their insertion, in the script pane.



Header Control
- Company
- Approver Long Address
- Long Address Number
- Display Business Unit
- VendorInvoiceNumber
- Payment Terms Code
- DateInvoiceJ
- DateForGLandVoucherJULIA

Source of Input
- Literal Value
- Valid Values List
- Variable
- Form Interconnect Visual Assist
- Clear

Literal Value: 1

```
Begin Script - Repeat Count from "1"
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        >>INSERT
```

With the playback button turned on, as you make selections from the command pane in OneWorld Scripting Tool and click the Insert button, OneWorld Scripting Tool runs the scripted commands in OneWorld.

The Script Pane

As you write your script, you can easily observe its progress because OneWorld Scripting Tool records each command you insert in the script pane. The script pane contains visual expressions of each command that you insert.

The script pane consists of two components, command lines and the insertion cursor. The command lines express, in words and symbols, the selections you make in the command pane. A command line does not appear in the script pane until you have clicked the Insert button. Command lines express either the context in which a command is to be run, or the action that is to be taken in the chosen context. In other words, context commands specify *where* you want all or a portion of the script to take place, while action commands specify *what* you want to take place in the script.

Each command line contains all or some of the following components:

- A symbol designating the command as a script node
- A symbol that identifies the specific type of context or action command that you wrote and inserted to the script, such as an Application command or a Press Toolbar Button command
- A written description of the general context or action in OneWorld, for example, Application
- A written description of the specific context or action in OneWorld, for example, {P0411 - A/P Standard Voucher Entry}
- The source of the input to a OneWorld form and its value. Value from "1," for example, means that you have inserted the literal value 1 to a header control, grid column, or QBE line.

Context commands and action commands make up the command lines in the script pane. A context command establishes the environment in which you write other commands. In other words, in order to click a button in a form, you first establish the context, which in this example is a OneWorld form.

The following table summarizes the context commands that you write using OneWorld Scripting Tool and the results of writing those commands:

Context command	Result of writing command
Application	Launch OneWorld application
UBE	Launch OneWorld UBE, OneWorld application P98305 (Batch Versions) and OneWorld form W98305D (Version Prompting)
UBE Selection	Launch Oneworld Data Selection form
UBE Processing Options	Display processing options for a selection UBE in the OneWorld Scripting Tool command pane
UBE Print	Launch OneWorld Printer Selection form
Application Interconnect	With a OneWorld application and form active, launch a different application or a form in the same application that is outside the normal transaction sequence
Processing Options	Display processing options for a selected application in the OneWorld Scripting Tool command pane
Form	Specify the OneWorld form in which you want to take additional actions
Set Header Control Value	Specify header control to which you want to input data
Set Grid Cell Value	Specify grid cell to which you want to input data
Set QBE Cell Value	Specify grid cell in QBE line to which you want to input data

With a OneWorld context established, you can write action commands. One function of action commands is to define the actions that you take within the context you have specified. If the context is a OneWorld form, an action that you can take within that form is clicking a toolbar button. Therefore, "Press Toolbar Button" is an action command.

You can write other action commands independent of a specific OneWorld context. For example, you can declare a variable (give it a name) and set and store a value for it before you launch a OneWorld application. Likewise, you can declare a validation and associate it with a OneWorld table and columns in the table independently of establishing a context. Still, you take these actions in order to eventually accomplish something in a OneWorld context. For example, you store the value of a variable in order to use it in a header control, grid column, combo box, or tree path.

The following table summarizes the action commands that you write using OneWorld Scripting Tool and the results of writing those commands:

Action Command	**Result(s) of writing command**
Type Fast Path	Type FastPath to OneWorld interactive application or UBE
Type to	Enter data to a header control, or grid cell
Select Grid Row	Select a grid row in the detail area of a OneWorld form
Press Toolbar Button	Press standard buttons in OneWorld form, perform form and row exits, submit UBEs, select a grid tab, or press the grid scroll bar button
Press Push Button	Press special buttons that do not reside on the toolbar of OneWorld forms
Checkbox/Radio Button	Choose checkbox or radio button options in the header portion of OneWorld form
Command Line	Encapsulate path to another program in the OneWorld Scripting Tool script
Comment/Wait	Write a comment about the script and insert it to the script pane; designate a command line and time period for OneWorld Scripting Tool to wait before proceeding with script playback
Variables	Declare a name for a variable, designate the source of its value, set the value; store the value
Declare New Validation	Declare a name for a database validation
Associate a Validation Column	Associate a OneWorld table and a column with the declared validation; specify a value to be validated
Execute Validation	Write an SQL statement to validate whether an expected value is returned from the database
If <var> == <var>	Write a conditional (if/then) statement

Exit OneWorld	Exit OneWorld
Select Item in Combo Box	Choose item(s) in OneWorld forms that use combo boxes instead of header controls
Build Tree Path	Create a unique path to an item in a OneWorld form that uses tree controls

Each time that you write a command and insert it to your script, a command line appears in the script pane, followed by the insertion cursor, which appears as an arrow. You can move the insertion cursor to a point you choose by clicking a command line or by dragging the insertion cursor. The insertion cursor indicates where OneWorld Scripting Tool inserts additional command lines you create from lists in the command pane.

See Also

- *Scripting the Context*
- *Scripting Actions*
- *Understanding the Script Pane Structure*
- *Modifying Scripts*

Bars in the OneWorld Scripting Tool Form

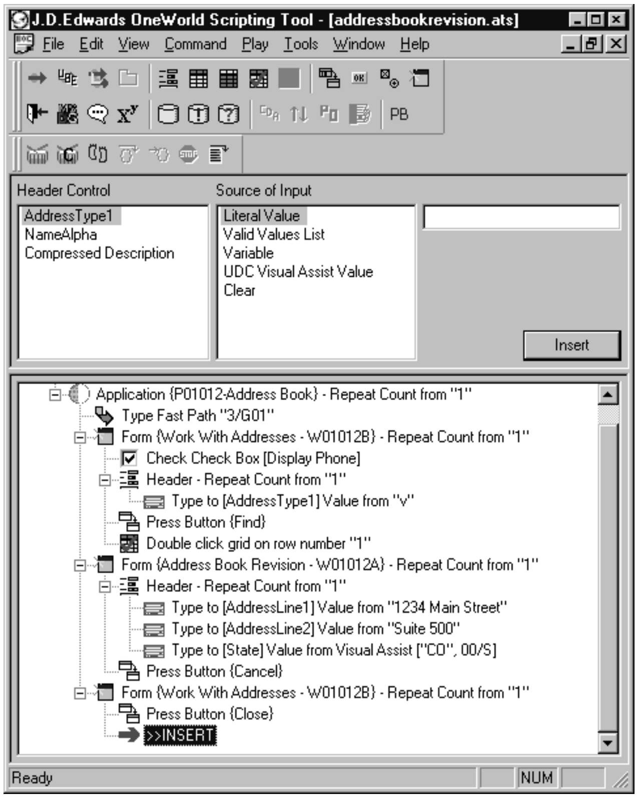
In addition to the command and script panes, the OneWorld Scripting Tool form includes four bars that assist you as you create a script. Bars in the OneWorld Scripting Tool form identify the script, contain options and buttons for scripting commands, and help identify the functions of buttons contained in the form.

This chapter discusses the four bars in the OneWorld Scripting Tool form:

- ❑ Caption bar
- ❑ Menu bar
- ❑ Cool bar
- ❑ Status bar

The Caption Bar

The caption bar is the horizontal bar running across the top of the OneWorld Scripting Tool form. It identifies the name or title of the script that you are writing in OneWorld Scripting Tool. The name of the script is enclosed in brackets in the caption bar.


J.D.Edwards OneWorld Scripting Tool - [addressbookrevision.ats]
File
Edit
View
Command
Play
Tools
Window
Help
PB
Insert
Ready
NUM

Header Control

- AddressType1
- NameAlpha
- Compressed Description

Source of Input

- Literal Value
- Valid Values List
- Variable
- UDC Visual Assist Value
- Clear

```
Application {P01012-Address Book} - Repeat Count from "1"
    Type Fast Path "3/G01"
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
        Check Check Box [Display Phone]
        Header - Repeat Count from "1"
            Type to [AddressType1] Value from "v"
        Press Button {Find}
        Double click grid on row number "1"
    Form {Address Book Revision - W01012A} - Repeat Count from "1"
        Header - Repeat Count from "1"
            Type to [AddressLine1] Value from "1234 Main Street"
            Type to [AddressLine2] Value from "Suite 500"
            Type to [State] Value from Visual Assist ["CO", 00/S]
        Press Button {Cancel}
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
        Press Button {Close}
        >>INSERT
```

The Menu Bar

The menu bar runs horizontally beneath the caption bar. It is composed of items (beginning with File and ending with Help) that contain drop-down menus from which you can make selections that help you to write the script and to set up how OneWorld Scripting Tool runs.

This topic discusses each of the options on the OneWorld Scripting Tool menu bar:

- □ File
- □ View
- □ Command
- □ Play
- □ Tools
- □ Window
- □ Help

The File Option

Many of the choices in the drop-down menu of the File option represent essential Windows functions. You can create a new script, open an existing one, close a script, save it, or print it. However, three of the choices reflect unique OneWorld Scripting Tool features: Send To, Properties, and Repository. Send To enables you to send a script that you have written to another user who has access to OneWorld Scripting Tool. Properties permits you to assign identifying features to the script, such as the function that the script tests. Repository offers access to the script repository, which is a controlled storage location for completed scripts. This location is separate from your local drive and can be accessed by OneWorld Scripting Tool users to obtain examples of scripts that test particular functions.

See Also

- *Property Pages for Scripts*
- *Understanding the Script Repository*

The View Option

You use the View option to set up the way that the cool bar and status bar display in the OneWorld Scripting Tool form. When you click View, the drop-down menu includes Tool Bar and Status Bar. If you click Tool Bar, OneWorld Scripting Tool displays a Toolbars form with two check box options: Construct/Edit and Play. These represent the buttons on the cool bar. Construct/Edit represents the cool bar buttons that you can click to write commands to your script such as Application. Play represents the cool bar buttons you use to play back your scripts. If you check the check box options, these buttons are visible on the cool bar. If you uncheck the options, the buttons are not visible. If you click Status Bar, you toggle the status bar on or off, depending on its original condition.

The Command Option

The drop-down menu that appears when you click Command in the menu bar contains the names of the commands that you can write to your script. These commands are the same ones represented by the Construct/Edit check box option under View in the menu bar, and they also mirror the cool bar buttons that you can click to write commands to the script. Each of the commands in the menu has an underlined letter that functions as a hot key to begin scripting the command. For example, if you click F on the keyboard, the command pane in the OneWorld Scripting Tool form displays the lists you use to script the Form command. If OneWorld Scripting Tool has grayed out a command in the menu, that command is not available for scripting.

The drop-down menu under the command option also includes two choices that are not represented by cool bar buttons. If <var> = = <var> represents the command to write a conditional statement. Clicking Playback on Creation toggles on or off the Playback button on the cool bar. With the playback button on, OneWorld Scripting Tool plays in OneWorld the commands that you write in your script.

The Play Option

The drop-down menu that appears when you click Play in the menu bar contains the names of the OneWorld Scripting Tool playback functions, each of which is represented by a cool bar button. These functions are also represented by the Play check box option under View in the menu bar. You can activate the function by clicking a hot key that is again identified by an underlined letter in each of the entries in the drop-down menu. For example, if you click T on the keyboard, OneWorld Scripting Tool automatically plays back from the top the script that you have open.

The Tools Option

Using the Tools option allows you to fine-tune the way your script runs, to view the results of test scripts you have run, and to generate data that you can use in your scripts. The following table summarizes the choices available from the Tools menu:

Tools Menu Choice	**Purpose**
Generate Valid Values List	Click to create or select data to store that you store in a text file for use in the script.
Options	Click to launch the Options form, which contains seven tabs that you use to set up the following: OneWorld Scripting Tool directories, playback speed, playback configuration, sign-on parameters, playback against OneWorld Java, playback against OneWorld HTML, and specifications for script creation.
Include Local Script	Click to choose a script stored locally that you want to include with another script.
Include Repository Script	Click to choose a script stored in the repository that you want to include with another script.
Results	Click to review results of OneWorld Scripting Tool tests you have run.
Select OneWorld Client	Click to choose from options to run script to test Windows, Java, or HTML client.

Because the options under Tools serve disparate purposes, it is useful to briefly discuss each of them in more detail:

- ❑ Generate Valid Values List
- ❑ Options for configuring OneWorld Scripting Tool
- ❑ Include Local Script
- ❑ Include Reposited Script
- ❑ Results
- ❑ Select OneWorld Client

Generate Valid Values List

You can create a text or numeric file containing one or more values by clicking Generate Valid Values List under the Tools option. When you do so, a form appears that allows you to select data and to save it in a file. You can then use it as a source of input for your script. When you click Generate Valid Values List, you use the Select Data Files form to create a valid values list either by querying the database or by manually entering values of your own.

See Also

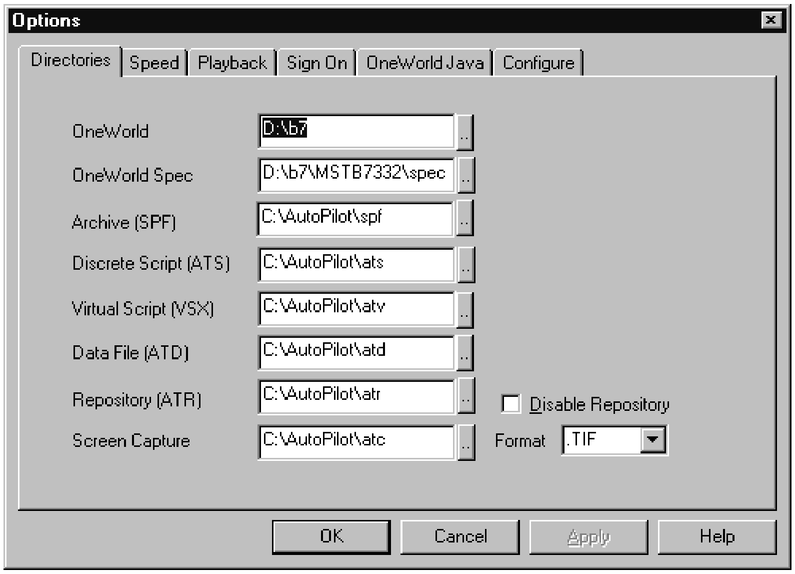
- *Valid Values List*
- *Using a Valid Values List as a Source of Input*

Options for Configuring OneWorld Scripting Tool

Clicking Options produces a form with seven tabs, each of which contain controls and options to help you set up OneWorld Scripting Tool for testing:

- Directories
- Speed
- Playback
- Sign On
- OneWorld Java
- OneWorld HTML
- Configure

The Directories tab allows you to specify where you start OneWorld and where you store your local scripts, screen captures, and so on.



Options

OneWorld: D:\b7

OneWorld Spec: D:\b7\MSTB7332\spec

Archive (SPF): C:\AutoPilot\spf

Discrete Script (ATS): C:\AutoPilot\ats

Virtual Script (VSX): C:\AutoPilot\atv

Data File (ATD): C:\AutoPilot\atd

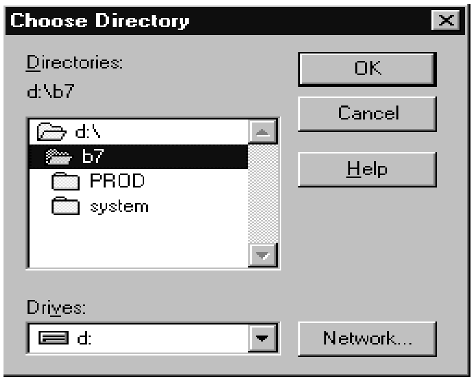
Repository (ATR): C:\AutoPilot\atr

□ Disable Repository

Screen Capture: C:\AutoPilot\atc

Format: .TIF

You can set the path for each directory by clicking the button next to each control. When you click the button, the Choose Directory form appears. You use this form to specify the path to each directory and the network drive on which that directory resides.



Choose Directory

Directories:
d:\b7

d:\
b7
PROD
system

Drives: d:

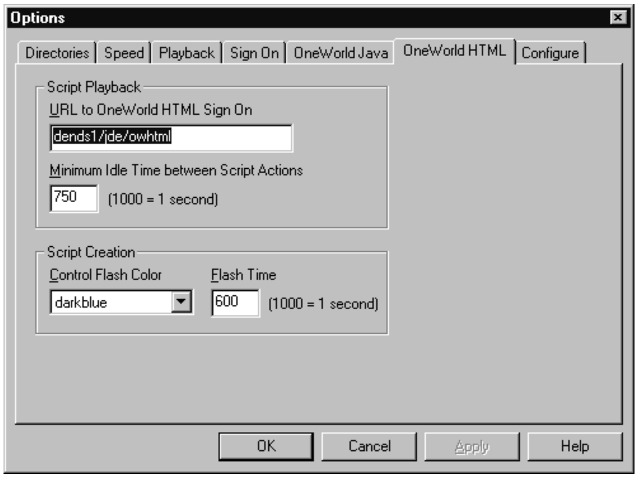
The Format combo box option allows you to choose a particular screen capture extension, such as .tif. If you do not want the option of adding scripts to the repository, you can disable it by choosing the Disable Repository option.

Under the Speed tab, you can set how quickly OneWorld Scripting Tool types to a header control or grid cell in a OneWorld form.

Under the Sign On tab, OneWorld Scripting Tool displays your user ID, password, and the OneWorld environment to which you sign on. If OneWorld signs you on to a different environment than the one that appears in the Environment control, OneWorld Scripting Tool displays a form advising you that you must change the sign-on environment to match the OneWorld environment.

Under the OneWorld Java tab, you can specify a Java application server, against which you can run a OneWorld Scripting Tool script.

Under the OneWorld HTML tab, you can enter the universal resource locator for a OneWorld web server, against which you can run a OneWorld Scripting Tool script.



Script Playback

URL to OneWorld HTML Sign On: dends1/jde/owhtml

Minimum Idle Time between Script Actions: 750 (1000 = 1 second)

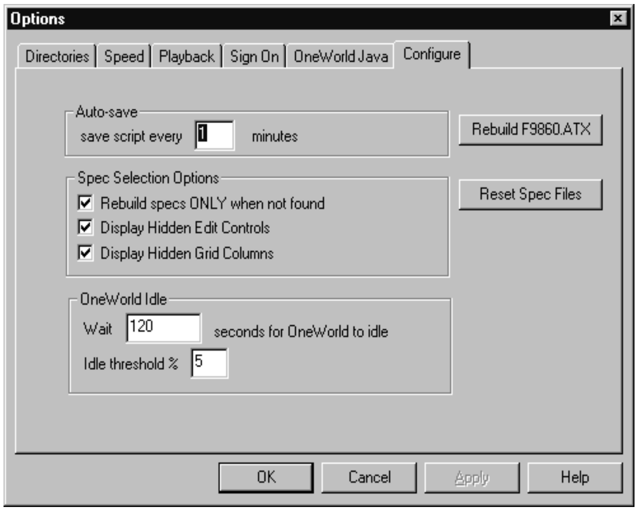
Script Creation

Control Flash Color: darkblue

Flash Time: 600 (1000 = 1 second)

Under the Configure tab, you can:

- Set how quickly you want your script to auto-save
- Select OneWorld specifications such as whether you want hidden edit and grid controls to display and whether you want to rebuild file specifications each time you run an application or only when OneWorld Scripting Tool does not find the specifications
- Set the threshold at which you want OneWorld to idle
- Click the Rebuild F9860.ATX table to load OneWorld application names into OneWorld Scripting Tool



Auto-save

save script every 1 minutes

Spec Selection Options

☑ Rebuild specs ONLY when not found

☑ Display Hidden Edit Controls

☑ Display Hidden Grid Columns

OneWorld Idle

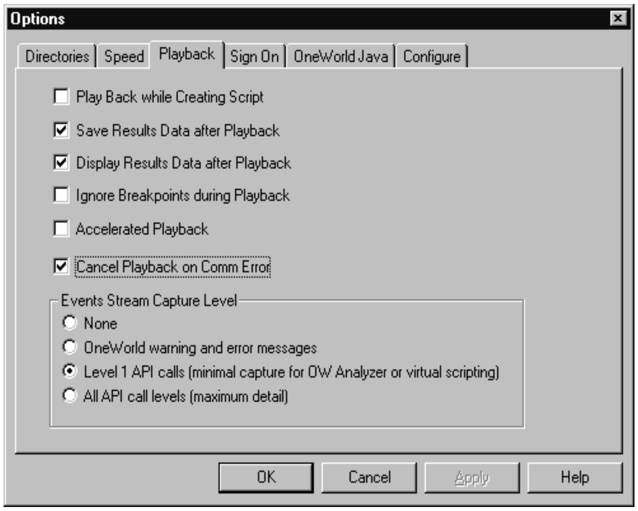
Wait 120 seconds for OneWorld to idle

Idle threshold % 5

The options under the Playback tab are divided into two sections. The top section of options allows you to configure script playback. The following table summarizes the purposes of the playback configure options.

Option	Purpose	Suggested Initial Setting
Playback while Creating Script	OneWorld Scripting Tool plays back in OneWorld each command after you insert it to the script.	Off
Save Results Data after Playback	OneWorld Scripting Tool writes data about script playback events to a OneWorld table, where the results are stored.	On. You must choose this option if you choose any option other than None from the Event Stream Capture Level section.
Display Results Data after Playback	OneWorld Scripting Tool displays a Results form, which contains summarized information about each playback event.	On
Ignore Breakpoints during Playback	During playback, OneWorld Scripting Tool ignores breakpoints that the user manually inserts into the script. If this option is not chosen, playback halts at a breakpoint until the user intervenes.	Off
Accelerated Playback	OneWorld Scripting Tool communicates, through code, directly with the OneWorld run–time engine to determine when a process is complete so that it can go on to the next command, thus speeding up playback.	Off. Choose the option only if you are certain that application launch is controlled by the run-time engine and not by a business function.
Cancel Playback on Comm Error	OneWorld Scripting Tool cancels playback if a communication error between client and server occurs. Choose this option when you are testing processes on a server.	Off

The bottom section of options allows you to set up capture of script playback data. The chronological sequence of events that occurs during script playback is called an event stream. Using the options under the Playback tab, you specify how much of the event stream you want OneWorld Scripting Tool to capture.



Options

☐ Play Back while Creating Script

☑ Save Results Data after Playback

☑ Display Results Data after Playback

☐ Ignore Breakpoints during Playback

☐ Accelerated Playback

☑ Cancel Playback on Comm Error

Events Stream Capture Level

○ None

○ OneWorld warning and error messages

● Level 1 API calls (minimal capture for OW Analyzer or virtual scripting)

○ All API call levels (maximum detail)

You capture event stream data to accomplish two main purposes. You can import the data to the OneWorld Analyzer Tool, which allows you to view data about each playback event in greater detail. For example, you can view the input values and return values of individual API calls, and you can see the time required to run each event. This information can aid in debugging applications.

You can also import the event stream to the Virtual Script Editor, where you can create a virtual script. You can run the virtual script on a single workstation to simulate many users, a process which helps you to test the scalability of your system.

For detailed instructions on how to use the OneWorld Analyzer Tool and the Virtual Script Editor, see *OneWorld Analyzer Tool* and *OneWorld Virtual User Tool* documentation.

The following table summarizes the purposes of the options under the Playback tab that allow you set up the capture of script playback data:

Option	**Purpose**
None	OneWorld Scripting Tool captures no data about script playback.
OneWorld warning and error messages	OneWorld Scripting Tool captures only data about OneWorld warning and error messages.
Level 1 API calls	OneWorld Scripting Tool captures OneWorld warning and error messages and captures API data only about those calls that initiate a business function or database call.
All API call levels	OneWorld Scripting Tool captures data about OneWorld warning and error messages and about all API calls made.

Include Local Script

You might want to write a script and include it with another that tests a related function. To do so, you click Include Local Script in the Tools drop-down menu. This enables you to choose a script you have saved to your local directory and to include it with another script of your choosing.

See Also

- *Including Scripts*

Include Reposited Script

You can also include with an open script a script that has been added to the script repository. To do so, you click Include Repository Script in the Tools drop-down menu. This enables you to browse through the scripts that have been checked into the repository, and you can choose one or more to include with a script of your choosing.

See Also

- *Including Scripts*
- *Understanding the Script Repository*

Results

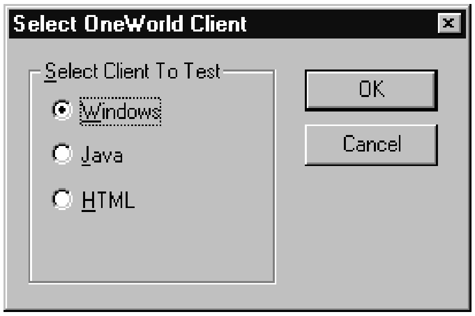
After you have played back a script, you have the option of saving the results of the test. OneWorld Scripting Tool collects those tests whose results you have elected to save and displays a summary of the test results in a Browse Results form when you click Results in the drop-down menu of the Tools option in the menu bar.

See Also

- *Understanding Script Reporting*

Select OneWorld Client

You can run your scripts as a Windows, Java, or HTML client by choosing an option from the Select OneWorld Client form. If you choose the Java or HTML option, you must choose Options from the Tools menu and specify a server under the OneWorld Java tab or the OneWorld HTML tab.



The Window Option

Window provides several options for changing the size and arrangement of OneWorld Scripting Tool forms. For example, you can choose to tile or cascade the forms if you have several open at once. In addition, the drop-down menu under Window displays the script or scripts that you currently have open.

The Help Option

Clicking the Help option displays the version of OneWorld Scripting Tool installed on your machine as well as the date that you took the build.

The Cool Bar

The cool bar is composed of buttons you click to:

- Script context and action commands

- Run script playback

You find the context and action commands represented by cool bar buttons in the Command option on the menu bar. You find the script playback commands represented by cool bar buttons in the Play Menu on the menu bar.

See Also

- *Scripting the Context*
- *Scripting Actions*
- *Playing Back the Script*

The Status Bar

The status bar, located at the bottom of the OneWorld Scripting Tool form, provides information about your OneWorld Scripting Tool session. For example, after you have begun a session and are preparing to enter a new command, the status bar displays the message Ready, meaning that OneWorld Scripting Tool is ready to accept a new command. When you pass the cursor over a cool bar button, the status bar displays a worded explanation of the function of that button.

Likewise, when you pass the cursor over any item that appears in a drop-down menu of the menu bar, the status bar displays a worded explanation of the item's function.

The status bar might also inform you that you need to wait before proceeding. For example, when you open a script for the first time in a session, the status bar asks you to wait while it loads the script specifications and reads the OneWorld specifications for an application that it has not yet found.

Manipulating the OneWorld Scripting Tool Form

You can easily change the arrangement and size of the OneWorld Scripting Tool form and the panes within it. You can focus completely on one pane by manipulating the size of the form. On the other hand, if you are working with multiple scripts, you can keep each of them open and arrange them so that you can conveniently move between them as you work.

You might also want to resize the OneWorld Scripting Tool form and its panes when you are creating a script and playing it back in OneWorld. Adjusting the size of the form allows you to see both the OneWorld Scripting Tool form and the forms that are active in OneWorld as you work.

If you close OneWorld Scripting Tool, then open it again, the size of the OneWorld Scripting Tool form, the arrangement of the panes, and the position of the cool bar appear as they did when you closed the session. This feature can be useful when you set up the form as you like it and want to preserve the setup.

This chapter covers the following topics:

❑ Changing the size of the OneWorld Scripting Tool form

❑ Arranging multiple OneWorld Scripting Tool forms

❑ Sizing panes in the OneWorld Scripting Tool form

Changing the Size of the OneWorld Scripting Tool Form

You can easily change the size of the OneWorld Scripting Tool form by using the mouse. Moving the mouse within the form produces double-headed arrows. You then can size the form by holding down the mouse button and dragging the mouse in the direction you desire.

This topic describes how you can use the cursor arrows to:

- Expand the area of the OneWorld Scripting Tool form
- Change the horizontal area of the OneWorld Scripting Tool form
- Change the vertical area of the OneWorld Scripting Tool form

▶ To expand the area of the OneWorld Scripting Tool form

1. On your desktop, with the OneWorld Scripting Tool form active, bring the cursor arrow to a corner of the screen.
2. When a diagonal doubled-headed arrow appears, left-click the mouse.
3. Holding down the mouse button, drag the mouse in the direction desired.

▶ To change the horizontal area of the OneWorld Scripting Tool form

1. On your desktop, with the OneWorld Scripting Tool form active, place the cursor arrow at one of the vertical edges of the OneWorld Scripting Tool form.
2. When a double-headed horizontal arrow appears, left-click the mouse.
3. Holding down the mouse button, drag the mouse to the left or to the right.

▶ To change the vertical area of the OneWorld Scripting Tool form

1. On your desktop, with the OneWorld Scripting Tool form active, place the cursor arrow over the caption bar or the bottom edge of the OneWorld Scripting Tool form.
2. When a double-headed vertical arrow appears, left-click the mouse.
3. Holding down the mouse button, drag the mouse up or down.

Arranging Multiple OneWorld Scripting Tool Forms

OneWorld Scripting Tool allows you to create and save multiple scripts during a single session or several sessions. You can open several scripts at once, resizing and rearranging them as you see fit.

You use the menu bar Window option to change the size and arrangement of OneWorld Scripting Tool forms when you want to work with more than one script. You can arrange multiple scripts so you can view them simultaneously; likewise, you can easily move from one script to another.

If you decide to work with multiple OneWorld Scripting Tool scripts during a session, you can arrange the OneWorld Scripting Tool forms in either cascade or tile fashion, using the menu bar Window option. The Cascade command arranges the scripts in overlay fashion.

The top OneWorld Scripting Tool form is active. Click another form to make it active. To resize a form, place the cursor arrows on a vertical or horizontal edge, hold down the mouse button, and drag the form in the direction you desire.

The Tile command divides the area of the OneWorld Scripting Tool form so that each existing OneWorld Scripting Tool form appears simultaneously, adjacent to one another.

▶ To arrange multiple OneWorld Scripting Tool forms

1. In the menu bar of the OneWorld Scripting Tool form, choose Window.
2. From the drop-down menu, choose either Cascade or Tile.
3. Use the cursor arrow to change the size of the parent OneWorld Scripting Tool form or of any of the child forms.

Sizing Panes in the OneWorld Scripting Tool Form

You can change the size of panes in the OneWorld Scripting Tool form easily using the Split option, which you find under Window in the menu bar. The split option drops the cursor arrow to the splitter bar, which divides the command pane from the script pane. You can manually place the cursor arrow on the splitter bar at any time to resize the panes to your preferences.

▶ To size panes in the OneWorld Scripting Tool form

1. In the menu bar of the OneWorld Scripting Tool form, choose Window.
2. Choose the script you want to work with from the drop-down menu.
3. Choose Split from the drop-down menu.

   An arrow appears at the splitter bar, which divides the top pane from the bottom pane.

4. Drag the mouse up or down, expanding or shrinking the size of the panes.

Manipulating the OneWorld Scripting Tool Cool Bar

You might work frequently with the cool bar during a OneWorld Scripting Tool session because you use many of its buttons to write context and action commands. You can also move the cool bar and change its size and shape to make your work easier.

For instance, if you want more vertical space for the command pane, you can move the cool bar from near the top horizontal edge of the OneWorld Scripting Tool form to either the right or left vertical edge. You can also float the cool bar, moving it entirely out of the OneWorld Scripting Tool form and onto the desktop. Finally, after you have moved the cool bar from one position to another, you can return it to its original position by double clicking the bar.

This chapter covers the following tasks:

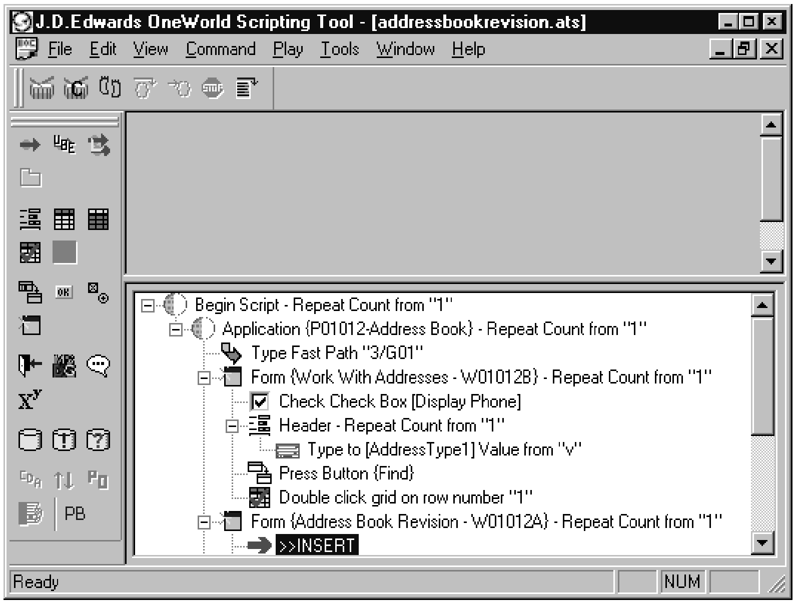
- ❑ Relocating the cool bar
- ❑ Resizing the cool bar
- ❑ Floating the cool bar

Relocating the Cool Bar

You can move the cool bar, using the grabber, which is represented by a pair of vertical bars. Note that there actually two cool bars. One contains the buttons that represent action and context commands that you use to write your scripts. The other contains buttons that you use to play back your scripts. Each bar contains a grabber, so you can move one, the other, or both.

▶ To relocate the cool bar

1. In the OneWorld Scripting Tool form, place the cursor arrow on the grabber, represented by double vertical bars in the cool bar.
2. Left-click the mouse.
3. Holding down the mouse button, drag the cool bar, which you can now place either vertically along the right or left edge or horizontally along the bottom of the OneWorld Scripting Tool form.


J.D.Edwards OneWorld Scripting Tool - [addressbookrevision.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
  Application {P01012-Address Book} - Repeat Count from "1"
    Type Fast Path "3/G01"
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
      Check Check Box [Display Phone]
      Header - Repeat Count from "1"
        Type to [AddressType1] Value from "V"
      Press Button {Find}
      Double click grid on row number "1"
    Form {Address Book Revision - W01012A} - Repeat Count from "1"
      >>INSERT
```

You can split up the two sections of the cool bar and place one along a vertical edge and one along a horizontal edge of the OneWorld Scripting Tool form, or you can place them together.

Resizing the Cool Bar

You may want to change the size and shape of the cool bar. You can do so easily by using the cursor arrows and the mouse button.

▶ To resize the cool bar

1. In the OneWorld Scripting Tool form, place the cursor arrow at the edge of the form.
2. When a double-headed horizontal arrow appears, click and hold down the mouse button.
3. Drag the mouse up, down, or across.

   As you resize the OneWorld Scripting Tool form, the cool bar resizes along with it.

Floating the Cool Bar

It may be more convenient for you to drag the cool bar completely outside of the OneWorld Scripting Tool form and work with it from your desktop. To do so, you simply use the mouse to grab the bar and drag it to the position you desire, or you can double-click the bar. Once the cool bar is in a floating position, you can use the mouse to resize it.

This topic explains the following tasks:

- Floating the cool bar by dragging
- Floating the cool bar by double clicking
- Resizing and reshaping the cool bar from the floating position

▶ To float the cool bar by dragging

1. In the OneWorld Scripting Tool form, place the cursor arrow on the grabber in the cool bar.
2. Click and hold down the mouse button.
3. Drag the cool bar to any position desired.

   An outline of the cool bar appears as you drag it.

4. When the outline of the cool bar assumes the shape that you want, release the mouse.

▶ To float the cool bar by double clicking

1. Place the cursor arrow anywhere within the cool bar.
2. Double click the mouse.
3. To return the cool bar to its original position, place the cursor in the Construct/Edit bar that runs along its top and double click the mouse again.

▶ To resize and reshape the cool bar from the floating position

1. In the OneWorld Scripting Tool form or on your desktop, with the cool bar in a floating position, place the cursor arrow on one of its corners or edges.
2. When the double-headed vertical, horizontal, or diagonal arrow appears, left-click the mouse.
3. Holding down the mouse button, drag the arrow away from the bar until a resized, reshaped outline of the bar appears.
4. Release the mouse.

   The cool bar in its new configuration appears.

5. Click the top of the resized cool bar to automatically resize it again.

6. Repeatedly click the top of the cool bar to return it to its original configuration.

Scripting Context

Scripting the Context

To create a script, you make selections from lists in the command pane. These selections create the commands that you insert to the script, and you then play back these commands to test OneWorld applications.

There are two kinds of commands that you insert to a OneWorld Scripting Tool script: context and action. You use context commands to establish the OneWorld setting that you want to test. These settings include applications, UBEs, interconnected applications, processing options, forms, headers, grid columns, and QBE lines. With a context established, you can write action commands, which accomplish specific tasks you perform in OneWorld, such as pressing a button or typing to a header control.

Context commands can be dependent on other context commands. For example, you write an application command to launch a OneWorld application and form. You write a header command so that you can input data to one or more header controls in the form. Although applications, forms, and header controls are all contexts, you cannot type inputs to the header controls until you have established the application and form contexts in the script.

This section covers the following aspects of context commands:

- ❑ Understanding context commands
- ❑ Writing the script using context commands

Understanding Context Commands

You write context commands during a OneWorld Scripting Tool session to establish the OneWorld context in which you work. Each of these commands establishes a unique environment and you write each command according to the OneWorld Scripting Tool functions that you want to test. Remember that in general you must write context commands before you can decide on what actions you want to take in OneWorld.

The lists in the command pane change to reflect the context command that you have chosen. For example, the lists that appear in the command pane when you are writing an Application command are different from the lists that appear when you are writing a Form command. It is therefore important that you familiarize yourself with the concepts behind each of the context commands.

Context commands also establish a hierarchy in the script pane. For example, you typically begin a script by writing an Application command. In writing this command, you also choose another context, a OneWorld form, and the Form command lines both appear in the script pane. However, OneWorld Scripting Tool indents the Form command line beneath the Application command line. This indentation indicates that the Application command gave rise to the Form command and is therefore its parent and superior to it in the hierarchy of commands.

The most important point about this hierarchy for script-writing purposes is that it is an important factor in script playback. Changes that you make to a parent command affect the commands that are subordinate to it. For example, if you delete a parent command from the script, all the commands that are children of that command are deleted as well.

You might consider the following as a simple hierarchy of OneWorld Scripting Tool commands:

- Primary context commands are Application, Application Interconnect, UBE, and Processing Options. These commands always provide the context for other context and action commands. They appear as parents to other commands in the script.
- Secondary context commands are Form, Header, Grid Column, and QBE. These commands generally are subordinate to primary context commands, but they provide the context for action commands. They appear as both parents and children of other commands in the script.

- Action commands such as clicking a toolbar button are nearly always subordinate to a primary or secondary context command. They nearly always appear as children of other commands in the script.

Keep in mind, however, that there are variations on these generalizations. For example, a Form command, when it gives rise to a Header, Grid Column, or QBE command, is primary to these commands, but secondary to the Application or Application Interconnect command.

This chapter defines each context command:

- ❑ Application command
- ❑ UBE command
- ❑ Application Interconnect command
- ❑ Processing Options command
- ❑ Form command
- ❑ Header command
- ❑ Grid Column command
- ❑ QBE command

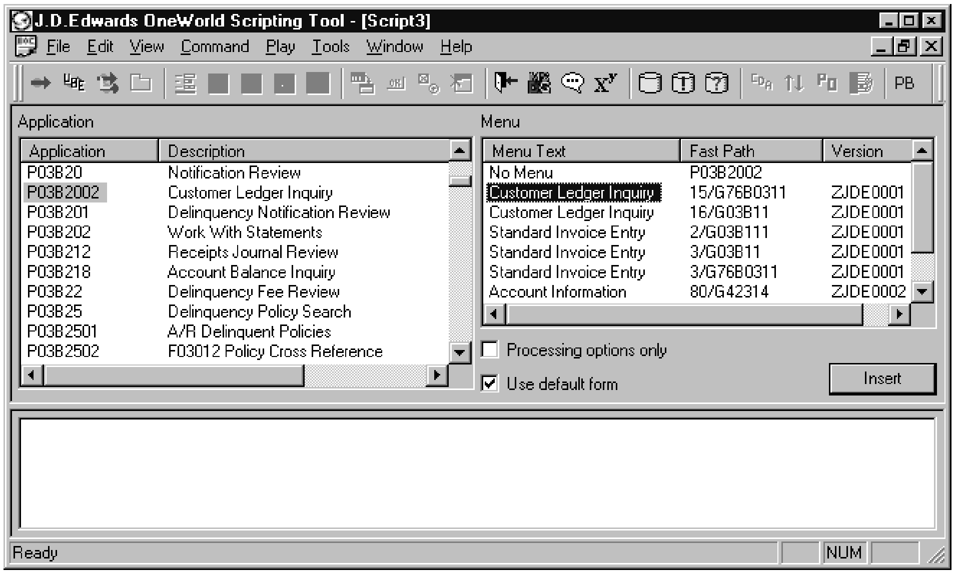
See Also

- *Understanding the Script Pane Structure* for further discussion of the hierarchy of commands.

Application Command

You use the Application command to launch interactive versions of OneWorld applications. Clicking the Application command allows you to choose a OneWorld application, the menu text for that application, and the application's FastPath.

The data that you need to write the Application command appear in lists in the command pane.


J.D.Edwards OneWorld Scripting Tool - [Script3]
File Edit View Command Play Tools Window Help
PB
Application
Menu
Insert
Ready
NUM

Application	Description
P03B20	Notification Review
P03B2002	Customer Ledger Inquiry
P03B201	Delinquency Notification Review
P03B202	Work With Statements
P03B212	Receipts Journal Review
P03B218	Account Balance Inquiry
P03B22	Delinquency Fee Review
P03B25	Delinquency Policy Search
P03B2501	A/R Delinquent Policies
P03B2502	F03012 Policy Cross Reference

Menu Text	Fast Path	Version
No Menu	P03B2002	
Customer Ledger Inquiry	15/G76B0311	ZJDE0001
Customer Ledger Inquiry	16/G03B11	ZJDE0001
Standard Invoice Entry	2/G03B111	ZJDE0001
Standard Invoice Entry	3/G03B11	ZJDE0001
Standard Invoice Entry	3/G76B0311	ZJDE0001
Account Information	80/G42314	ZJDE0002

☐ Processing options only

☑ Use default form

The Application command is a primary context command. You must script it in order to script inputs to header controls, grid columns, or QBE lines in OneWorld forms. An Application command is also often necessary if you want to interconnect to another OneWorld application.

UBE Command

You use the UBE command to launch previously created UBE versions when you want to submit a UBE to OneWorld for processing. OneWorld Scripting Tool allows you to launch UBE versions from a OneWorld menu, from a row or report exit, from an application calling for a blind UBE submission, or from another UBE. Once you have written a UBE command, you can write other commands. You can select data for your report, set UBE processing options, submit UBE versions to the printer, and instruct OneWorld Scripting Tool to wait for the UBE to complete processing before executing additional commands in the script. If it is necessary to do so, you can also write a command instructing OneWorld Scripting Tool to automatically exit the application Work With Batch Versions when you have completed scripting the UBE submission.

This topic covers the following components of launching, submitting, setting processing options for, selecting data for, and printing UBEs:

- Options for the UBE command
- UBE submission
- UBE data selection
- UBE processing options

- ❑ UBE Print command

Options for the UBE Command

You can write the UBE command at various points in the script. The decision to do so depends on the process that you are testing. When you click UBE in the command menu, the command pane lists that appear resemble the lists that appear when you click Application. You can choose from the lists a UBE, a menu Fast Path to the UBE, and a version.

The command pane also contains two options:

- ❑ Execute FASTPATH
- ❑ Create "Work With Batch Versions" commands

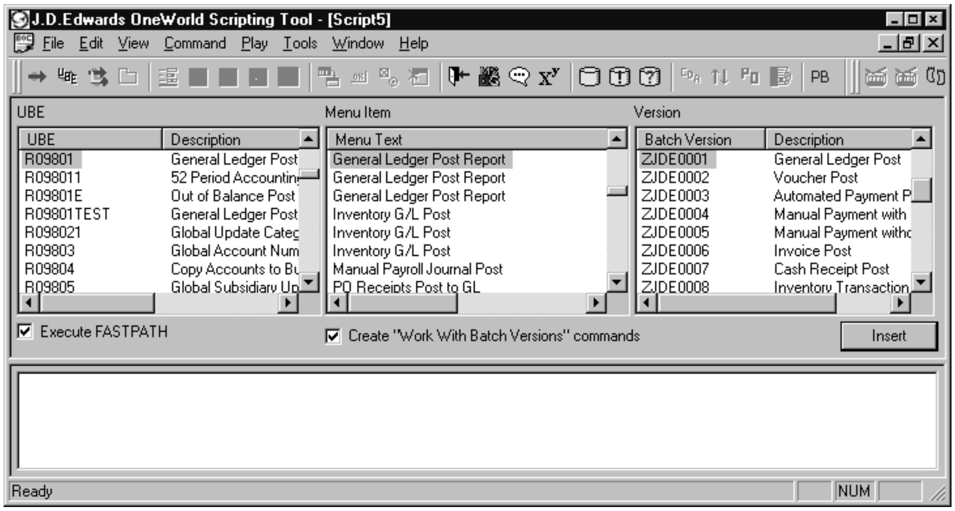
OneWorld Scripting Tool automatically chooses both of these options when you click UBE. The lists in the command pane change to reflect whether you have chosen one, both, or neither of these options. For example, if you choose a UBE, then but do not choose the Execute FASTPATH option, OneWorld Scripting Tool removes the Menu Item list.

This topic also covers:

- ❑ Option combinations

The Execute FASTPATH Option

If you want to launch a UBE from a OneWorld menu, you choose the Execute FASTPATH option and make choices from all three lists in the command pane: UBE, Menu Item, and Version. When you click the Insert button, OneWorld Scripting Tool sends the Fast Path command to OneWorld.



UBE

UBE	Description
R09801	General Ledger Post
R098011	52 Period Accountin
R09801E	Out of Balance Post
R09801TEST	General Ledger Post
R098021	Global Update Categ
R09803	Global Account Num
R09804	Copy Accounts to Bu
R09805	Global Subsidiary Up

Menu Item

Menu Text
General Ledger Post Report
General Ledger Post Report
General Ledger Post Report
Inventory G/L Post
Inventory G/L Post
Inventory G/L Post
Manual Payroll Journal Post
PO Receipts Post to GL

Version

Batch Version	Description
ZJDE0001	General Ledger Post
ZJDE0002	Voucher Post
ZJDE0003	Automated Payment P
ZJDE0004	Manual Payment with
ZJDE0005	Manual Payment withc
ZJDE0006	Invoice Post
ZJDE0007	Cash Receipt Post
ZJDE0008	Inventory Transaction

☑ Execute FASTPATH

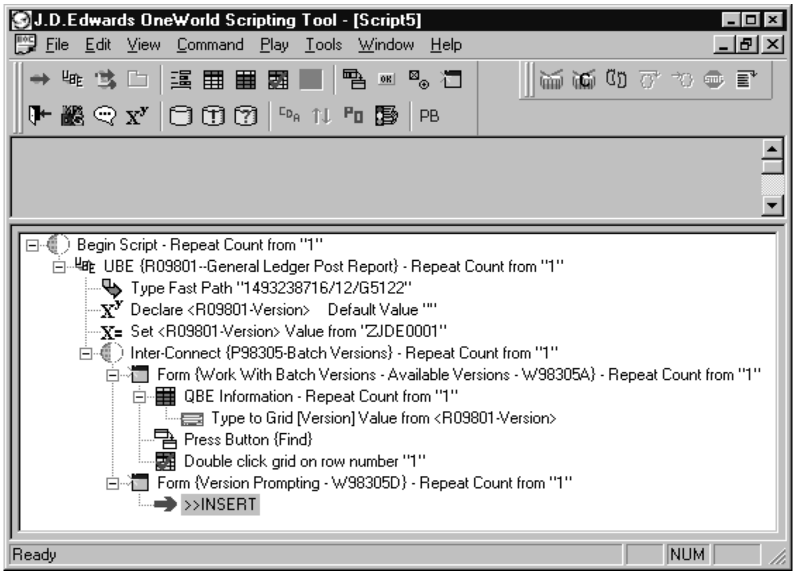
☑ Create "Work With Batch Versions" commands

In some cases you may not want to launch a UBE that uses a Fast Path to menu travel. For example, double clicking a grid row or clicking a button might launch a blind submission of a UBE. You might also access a UBE by performing a report exit in the Work With Batch Versions application. Finally, you might launch a UBE that is coded to automatically submit another UBE. In any of these cases, you leave the Execute FASTPATH option unchosen after you choose a UBE, and OneWorld Scripting Tool removes the Menu Item list that contains the Fast Paths to OneWorld.

The Create "Work With Batch Versions" Commands Option

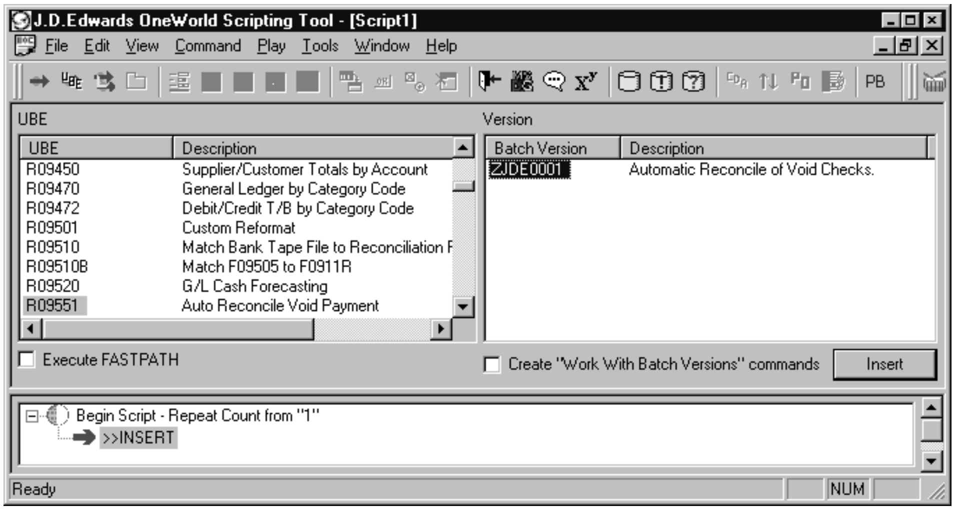
When you have completed your choices from the command pane and click the Insert button, OneWorld Scripting Tool automatically declares and sets a variable that stores the UBE version that you chose or that is automatically submitted. When you choose the Create "Work With Batch Versions" commands option, OneWorld Scripting Tool writes several additional commands to the script without your intervention:

- Interconnects to the P98305-Batch Versions application
- Launches the Form Work With Batch Versions
- Writes a QBE context command to the script
- Inputs the stored variable value to the QBE line
- Runs a Press Toolbar Button {Find} command
- Selects and double-clicks a row
- Confirms the Version Prompting form


J.D.Edwards OneWorld Scripting Tool - [Script5]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
  UBE {R09801--General Ledger Post Report} - Repeat Count from "1"
    Type Fast Path "1493238716/12/G5122"
    Declare <R09801-Version>    Default Value ""
    Set <R09801-Version> Value from 'ZJDE0001"
    Inter-Connect {P98305-Batch Versions} - Repeat Count from "1"
      Form {Work With Batch Versions - Available Versions - W98305A} - Repeat Count from "1"
        QBE Information - Repeat Count from "1"
          Type to Grid [Version] Value from <R09801-Version>
        Press Button {Find}
        Double click grid on row number "1"
      Form {Version Prompting - W98305D} - Repeat Count from "1"
        >>INSERT
```

If you do not choose this option, OneWorld Scripting Tool writes no script lines to exit to the Work With Batch Versions form. You do not choose the option whenever the UBE you are launching is submitted automatically from a menu, an application, or another UBE. When you launch a UBE from a menu that is hard-coded to submit the version automatically, OneWorld Scripting Tool removes the Versions list from the command pane and disables both of the options.


J.D.Edwards OneWorld Scripting Tool - [Script1]
File Edit View Command Play Tools Window Help
PB
UBE
Version
□ Execute FASTPATH
□ Create "Work With Batch Versions" commands
Insert
Ready
NUM

UBE	Description
R09450	Supplier/Customer Totals by Account
R09470	General Ledger by Category Code
R09472	Debit/Credit T/B by Category Code
R09501	Custom Reformat
R09510	Match Bank Tape File to Reconciliation F
R09510B	Match F09505 to F0911R
R09520	G/L Cash Forecasting
R09551	Auto Reconcile Void Payment

Batch Version	Description
ZJDE0001	Automatic Reconcile of Void Checks.

```
Begin Script - Repeat Count from "1"
    >>INSERT
```

When you click the Insert button, OneWorld Scripting Tool automatically submits the UBE. Remember that so long as a menu is not hard-coded to submit a UBE version automatically, you can still exit to the Work With Batch Versions form manually even if you do not choose the Create “Work With Batch Versions” commands option.

Option Combinations

Depending on the OneWorld operation you are testing, you can launch UBEs from different OneWorld locations. The location dictates the combination of options you choose.

The following table shows five different scenarios for launching a UBE and the combination of options that you choose.

UBE Launch	Execute FASTPATH Option	Create “Work With Batch Versions” Commands Option
From a OneWorld Menu	Chosen	Chosen
From a Reports Menu in an interactive application	Unchosen	Chosen
From a Row Exit in an interactive application	Unchosen	Unchosen
From a menu hard-coded to submit the UBE as a blind execution	Disabled	Disabled
From another UBE	Disabled	Unchosen

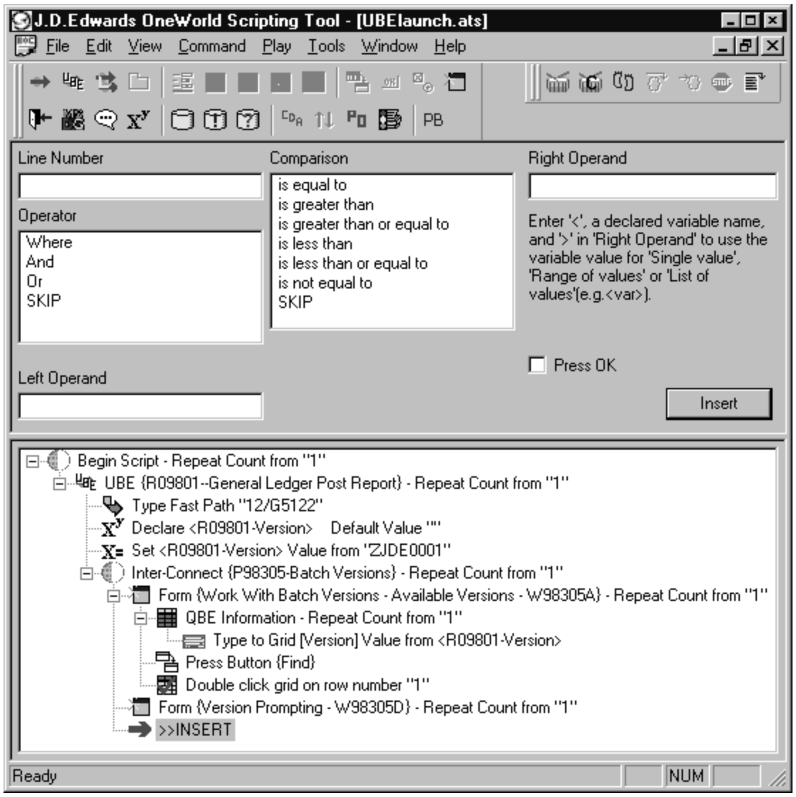
UBE Submission

When you write a script that includes the Work With Batch Versions application, you must write the command to submit the UBE. You do so by writing a Press Toolbar Button {Submit} command. This command presses the Submit button in the Version Prompting form in OneWorld. Note that at this point, OneWorld Scripting Tool can handle scripting data selection but not sequencing, so you should not choose the data sequencing option. If you want to select data for your report, you choose the data selection option, and then submit the report. If the UBE is a blind execution, you do not work with the Version Prompting form. OneWorld automatically submits the UBE, and you can write the command to print it.

UBE Data Selection

If you launch your UBE with the Create “Work With Batch Versions” commands option chosen, you can also use OneWorld Scripting Tool’s Criteria Design Aid feature to select the data for your report. The UBE context command and the UBE data selection action command work together when you script in OneWorld Scripting Tool. After you have launched a UBE and written, either automatically or manually, a series of commands that runs through the Version Prompting form in OneWorld, you can use OneWorld Scripting Tool’s Checkbox/Radio Button command to choose the Data Selection option in OneWorld, and then to submit the form by writing a Press Toolbar Button command. OneWorld Scripting Tool then allows you to script the data selection criteria by making entries to and choices from the command pane.

You script data selection by clicking UBE Selection in the command menu or by clicking the CDA button in the cool bar. When you do so, the command pane appears with five lists: Line Number, Operator, Left Operand, Comparison, and Right Operand. In addition, the command pane contains an option that allows you to press the OK button on the Data Selection form in OneWorld.


J.D.Edwards OneWorld Scripting Tool - [UBElaunch.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

Line Number

Operator

Where
And
Or
SKIP

Left Operand

Comparison

is equal to
is greater than
is greater than or equal to
is less than
is less than or equal to
is not equal to
SKIP

Right Operand

Enter '<', a declared variable name, and '>' in 'Right Operand' to use the variable value for 'Single value', 'Range of values' or 'List of values'(e.g.<var>).

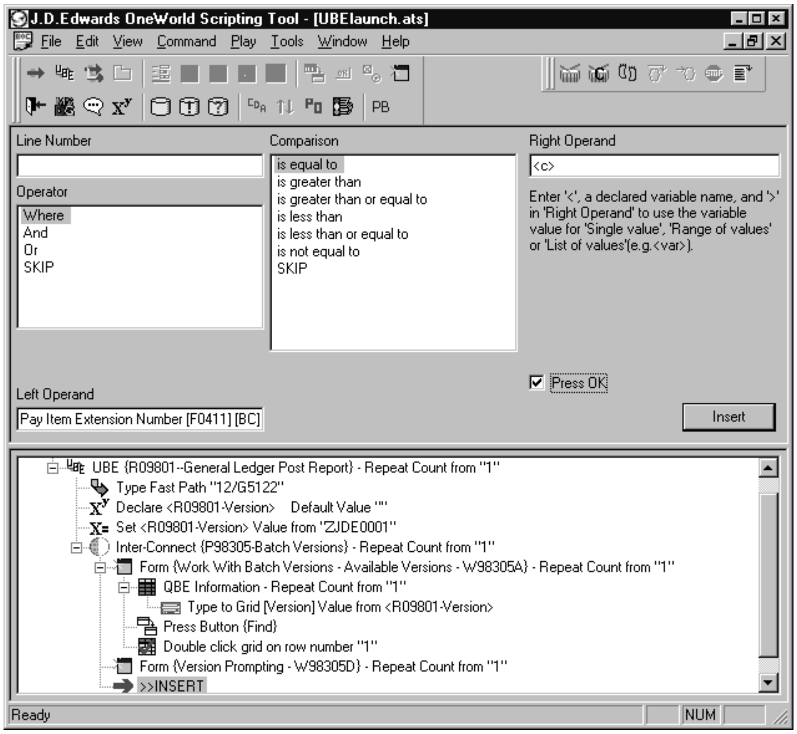
□ Press OK

```
Begin Script - Repeat Count from "1"
  UBE {R09801--General Ledger Post Report} - Repeat Count from "1"
    Type Fast Path "12/G5122"
    Declare <R09801-Version>    Default Value ""
    Set <R09801-Version> Value from "ZJDE0001"
    Inter-Connect {P98305-Batch Versions} - Repeat Count from "1"
      Form {Work With Batch Versions - Available Versions - W98305A} - Repeat Count from "1"
        QBE Information - Repeat Count from "1"
          Type to Grid [Version] Value from <R09801-Version>
        Press Button {Find}
        Double click grid on row number "1"
      Form {Version Prompting - W98305D} - Repeat Count from "1"
      >>INSERT
```

Note that the command pane lists mirror the functions of the Data Selection form in OneWorld. After you enter a line number for data, you determine the logic for the data selection that you want to enter to formulate your criteria. Note that the Operator and Comparison lists contain a SKIP option. If, for example, you have completed your entries to the Data Selection form for one line, and you want to make entries to another line, you can choose the SKIP option if you want the Operator and./or Comparison entries for the new line to duplicate the entries for the previous line.

While OneWorld Scripting Tool's data selection feature essentially mirrors the OneWorld Criteria Design Aid feature, there are at present some limitations that are important to keep in mind. For example, you can choose a left operand in the OneWorld Data Selection form by clicking a selection in a drop-down menu. No such drop-down menu presently exists in OneWorld Scripting Tool. Note that the name of the object that populates the left operand in OneWorld differs from the name that appears in the drop-down list.

You must manually enter the name of the object, exactly as it appears in the OneWorld list.


J.D.Edwards OneWorld Scripting Tool - [UBElaunch.ats]
File Edit View Command Play Tools Window Help
PB
Insert

Line Number

Operator: Where, And, Or, SKIP

Comparison: is equal to, is greater than, is greater than or equal to, is less than, is less than or equal to, is not equal to, SKIP

Right Operand: <c>

Enter '<', a declared variable name, and '>' in 'Right Operand' to use the variable value for 'Single value', 'Range of values' or 'List of values'(e.g.<var>).

☑ Press OK

Left Operand: Pay Item Extension Number [F0411] [BC]

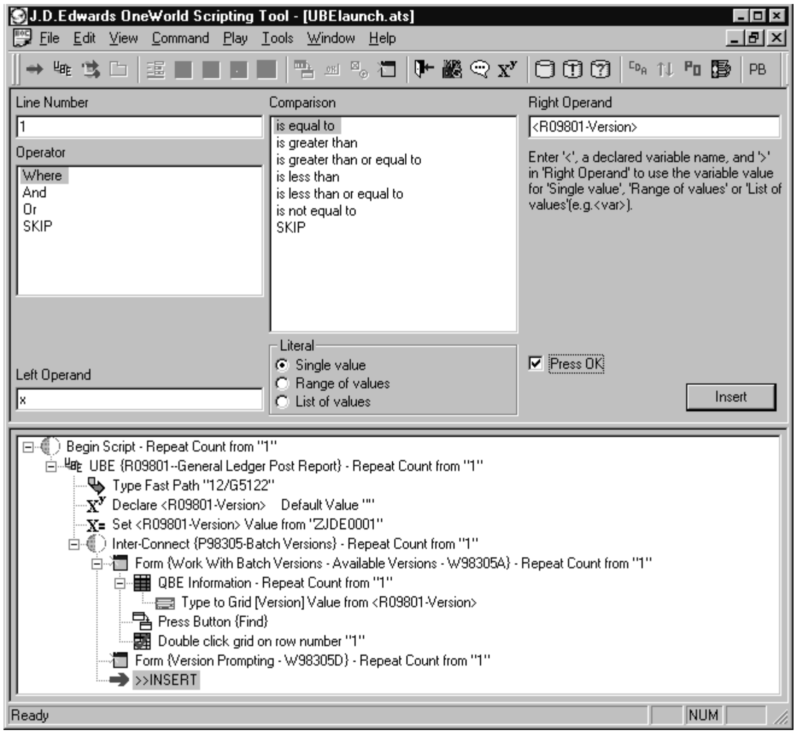
```
UBE {R09801--General Ledger Post Report} - Repeat Count from "1"
    Type Fast Path "12/G5122"
    Declare <R09801-Version>    Default Value ""
    Set <R09801-Version> Value from "ZJDE0001"
    Inter-Connect {P98305-Batch Versions} - Repeat Count from "1"
        Form {Work With Batch Versions - Available Versions - W98305A} - Repeat Count from "1"
            QBE Information - Repeat Count from "1"
                Type to Grid [Version] Value from <R09801-Version>
            Press Button {Find}
            Double click grid on row number "1"
        Form {Version Prompting - W98305D} - Repeat Count from "1"
        >>INSERT
```

Ready | NUM

Likewise, you must manually make entries to the Right Operand list as they appear in the drop-down menu, or you can make an entry of one or more literal values. You can enter multiple and range values to the right operand. You separate multiple values with commas: 1,2,5; you separate a range value with two hyphens: 1- -4.

In addition, while you can declare a variable, set its value, then use that value in the right operand, you must enter the name manually and enclose it in “greater than” and “less than” symbols, for example, <batchno>. In contrast, when you use a variable as a source of input to a header control or grid column, OneWorld Scripting Tool presents in the value selection list the names of all variables you have declared and allows you to choose one.

After you enter the name of a declared variable, OneWorld Scripting Tool displays options in the command pane that prompt you to designate the value of the variable as a literal, a range, or a list.



Line Number: 1

Operator:
- Where
- And
- Or
- SKIP

Comparison:
- is equal to
- is greater than
- is greater than or equal to
- is less than
- is less than or equal to
- is not equal to
- SKIP

Right Operand: <R09801-Version>

Enter '<', a declared variable name, and '>' in 'Right Operand' to use the variable value for 'Single value', 'Range of values' or 'List of values'(e.g.<var>).

Left Operand: x

Literal
- ● Single value
- ○ Range of values
- ○ List of values

☑ Press OK

```
⊟ Begin Script - Repeat Count from "1"
  ⊟ UBE {R09801--General Ledger Post Report} - Repeat Count from "1"
      Type Fast Path "12/G5122"
      Declare <R09801-Version>    Default Value ""
      Set <R09801-Version> Value from "ZJDE0001"
    ⊟ Inter-Connect {P98305-Batch Versions} - Repeat Count from "1"
      ⊟ Form {Work With Batch Versions - Available Versions - W98305A} - Repeat Count from "1"
        ⊟ QBE Information - Repeat Count from "1"
            Type to Grid [Version] Value from <R09801-Version>
          Press Button {Find}
          Double click grid on row number "1"
        Form {Version Prompting - W98305D} - Repeat Count from "1"
        >>INSERT
```

Creating and using variables can make the process of selecting data for your UBE more efficient. For example, you might want to write a script that enters transactions, then launches a UBE and extracts particular data for the report. Creating a variable allows you to store the data, such as a list of particular cost centers, you need for your report. When it comes time to select the data, you simply enter the name of the variable in the right operand of the Data Selection form. You can store in the variable a single value, a series of discrete values, or a range of values.

When you complete one set of criteria, you click the Insert button. With playback turned on, you can observe OneWorld Scripting Tool's entry of your criteria to the Data Selection form in OneWorld. If you click UBE Selection again, you can enter selection criteria to another line. When you have completed your data selection, you choose the OK option in the OneWorld Scripting Tool command pane and click the Insert button. This completes the data selection process. If processing options exist for the UBE version you have launched, they appear next, and you script processing options commands as necessary.

See Also

- *Source of Input List*
- *Using a Variable as a Source of Input*
- *Using the Value Selection List*
- *Creating Variable Links*

- *Selecting Data for a UBE*

UBE Processing Options

After you submit a UBE, some versions ask you to set processing options. You set these options in much the same way that you set processing options for interactive applications. However, the command menu entries and cool bar buttons that you choose to set UBE processing options are separate and distinct from those that you choose to set processing options for interactive applications.

When you set processing options for a UBE version, you click UBE Processing Options in the command menu. You then make choices from the Processing Options list in the command pane, then write a Press Toolbar Button {OK} command. OneWorld Scripting Tool inserts the chosen processing options to the script and runs them during playback.

See Also

- *Processing Options Command*
- *Setting the Context as a Processing Option*

UBE Print Command

You can send your UBE to print after you have submitted it or after OneWorld has automatically submitted it. You do so by clicking UBE Print in the command menu or the “stoplight” cool bar button.

OneWorld Scripting Tool offers three options in the command pane after you click UBE Print: Wait for UBE to complete before continuing, Expect No “Printer Selection” Window, and Create exit “Work With Batch Versions” command. Note that at this point, OneWorld Scripting Tool is not able to send UBEs to the screen in Adobe Acrobat format. When you submit a version, OneWorld Scripting Tool automatically chooses the To Printer radio button option in the Version Prompting form.

This topic discusses the three options that comprise the UBE print command:

❑ Wait for UBE to Complete option

❑ Expect No “Printer Selection” Window option

❑ Create exit “Work With Batch Versions” Command option

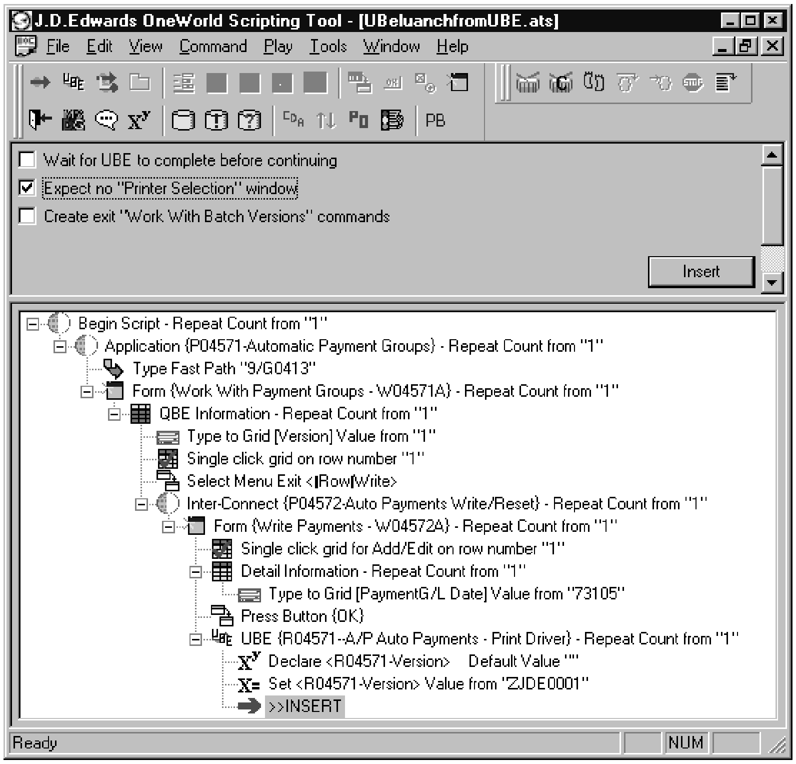
Wait for UBE to Complete Option

If you choose the option Wait for UBE to complete before continuing, OneWorld Scripting Tool submits the UBE to the default printer and waits for it to finish before resuming the script. If the UBE you submit launches additional UBEs, OneWorld Scripting Tool chooses the printer queue. When the printer has completed all of the submitted UBEs, OneWorld Scripting Tool resumes playing the script.

You can choose to leave this option unchosen. In that case, OneWorld Scripting Tool submits any UBEs for printing, but resumes the script without a waiting period. You can submit your UBE from either a local or server environment, but you cannot override the location once you have chosen it. In either environment, OneWorld Scripting Tool handles, without your intervention, all print windows that appear.

Expect No "Printer Selection" Window Option

You use the Expect No "Printer Selection" Window option if the UBE you are running does not require a printer. If this is the case, it is important that you choose this option because if you do not, OneWorld Scripting Tool waits for a printer window in OneWorld to appear before it OneWorld Scripting Tool resumes running the script. If the printer window does not appear, OneWorld Scripting Tool continues to wait, and the script fails to advance. Choosing this option tells OneWorld Scripting Tool not to expect and wait for a print window and to continue running the script after you submit the UBE.


J.D.Edwards OneWorld Scripting Tool - [UBeluanchfromUBE.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

□ Wait for UBE to complete before continuing
☑ Expect no "Printer Selection" window
□ Create exit "Work With Batch Versions" commands

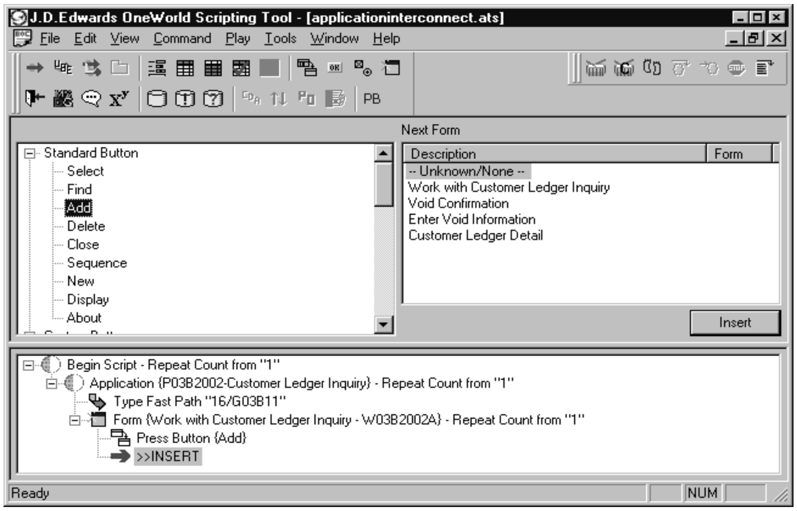
```
Begin Script - Repeat Count from "1"
  Application {P04571-Automatic Payment Groups} - Repeat Count from "1"
    Type Fast Path "9/G0413"
    Form {Work With Payment Groups - W04571A} - Repeat Count from "1"
      QBE Information - Repeat Count from "1"
        Type to Grid [Version] Value from "1"
        Single click grid on row number "1"
        Select Menu Exit <|Row|Write>
        Inter-Connect {P04572-Auto Payments Write/Reset} - Repeat Count from "1"
          Form {Write Payments - W04572A} - Repeat Count from "1"
            Single click grid for Add/Edit on row number "1"
            Detail Information - Repeat Count from "1"
              Type to Grid [PaymentG/L Date] Value from "73105"
            Press Button {OK}
            UBE {R04571--A/P Auto Payments - Print Driver} - Repeat Count from "1"
              Declare <R04571-Version>    Default Value ""
              Set <R04571-Version> Value from "ZJDE0001"
              >>INSERT
```

Create Exit "Work With Batch Versions" Command Option

If you launch a UBE from the Work With Batch Versions application, you can choose the Create exit "Work With Batch Versions" command option. When you do so, OneWorld Scripting Tool automatically writes a Form command line for Work With Batch Versions - Available Versions and writes a Press Toolbar Button {Close} command. These commands confirm and close the form. You return to the OneWorld menu item. If you do not launch a UBE from the Work With Batch Versions application, do not choose the option.

Application Interconnect Command

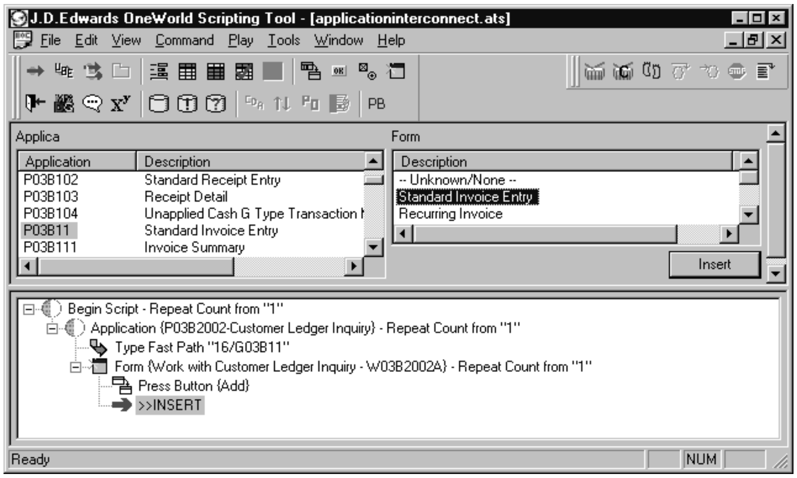
The Application Interconnect command allows you to script the exit from one application to another, which may occur, for example, when you press the Add button.


J.D.Edwards OneWorld Scripting Tool - [applicationinterconnect.ats]
File Edit View Command Play Tools Window Help
PB
Standard Button
Select
Find
Add
Delete
Close
Sequence
New
Display
About
Next Form
Description
Form
Insert
Ready
NUM

-- Unknown/None --
Work with Customer Ledger Inquiry
Void Confirmation
Enter Void Information
Customer Ledger Detail

```
Begin Script - Repeat Count from "1"
    Application {P03B2002-Customer Ledger Inquiry} - Repeat Count from "1"
        Type Fast Path "16/G03B11"
        Form {Work with Customer Ledger Inquiry - W03B2002A} - Repeat Count from "1"
            Press Button {Add}
            >>INSERT
```

Clicking Application Interconnect in the command menu lets you choose and insert to the script pane in OneWorld Scripting Tool the new application and form command lines that mirror the application and form that are active in OneWorld.

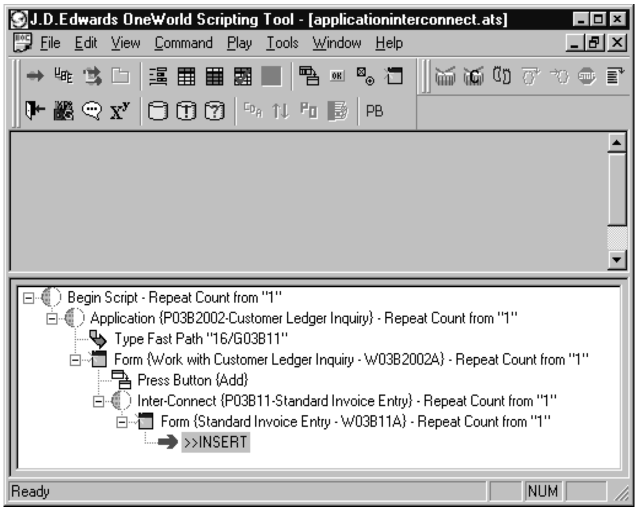

J.D.Edwards OneWorld Scripting Tool - [applicationinterconnect.ats]
File Edit View Command Play Tools Window Help
PB
Applica
Form
Insert
Ready
NUM

Application	Description
P03B102	Standard Receipt Entry
P03B103	Receipt Detail
P03B104	Unapplied Cash G Type Transaction I
P03B11	Standard Invoice Entry
P03B111	Invoice Summary

Description
-- Unknown/None --
Standard Invoice Entry
Recurring Invoice

```
Begin Script - Repeat Count from "1"
  Application {P03B2002-Customer Ledger Inquiry} - Repeat Count from "1"
    Type Fast Path "16/G03B11"
    Form {Work with Customer Ledger Inquiry - W03B2002A} - Repeat Count from "1"
      Press Button {Add}
      >>INSERT
```

You therefore script the Application Interconnect command *reactively*. That is, you have already exited to a new application in OneWorld. You must script an Application Interconnect command so that the OneWorld Scripting Tool script Application command and Form command matches with the application and form that are active in OneWorld.


J.D.Edwards OneWorld Scripting Tool - [applicationinterconnect.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
  Application {P03B2002-Customer Ledger Inquiry} - Repeat Count from "1"
    Type Fast Path "16/G03B11"
    Form {Work with Customer Ledger Inquiry - W03B2002A} - Repeat Count from "1"
      Press Button {Add}
      Inter-Connect {P03B11-Standard Invoice Entry} - Repeat Count from "1"
        Form {Standard Invoice Entry - W03B11A} - Repeat Count from "1"
          >>INSERT
```

If you do not do so, you cannot continue scripting because the Form command line in the script pane does not mirror the form and application that is active in OneWorld.

Remember that OneWorld Scripting Tool also automatically writes an Application Interconnect command to the script when you launch a UBE from a OneWorld menu or from a Reports menu in an interactive application. In each of these cases, you choose the Create "Work With Batch Versions" commands option in the command pane when you choose a UBE. OneWorld Scripting Tool launches the UBE, then automatically writes a series of commands to the script, including an Application Interconnect to the P98305-Batch Versions application.

You can also script an Application Interconnect command by clicking Press Toolbar Button in the command menu and choosing the Press Custom Button option. However, the two scripting approaches cannot be used interchangeably.

In deciding to choose Press Custom Button to script an Application Interconnect command, you *initiate* in OneWorld Scripting Tool the exit to a new application or form. OneWorld Scripting Tool inserts the Application and Form commands to the script and launches the OneWorld application and form.

Suppose you decide to launch Application P0010, Companies, and Form W0010C, Work With Companies. You decide you want to exit to a new application, so you click Press Toolbar Button in the command menu. When you click Press Custom Button, a tree node expands.

You choose Form or Row, and then, by clicking one or the other, choose from various form or row menu exits, which you use to script an Application Interconnect command. These menu exits mirror the lists you find if you click Form or Row in the menu bar of the active OneWorld form.

When you choose a form or row exit in OneWorld Scripting Tool, new lists appear in the command pane. You choose an application and form, then press the Insert button. OneWorld Scripting Tool runs the form or row exit and interconnects to the application that you have chosen.

You might close the interconnected application and return to the previous form. In that case, be sure that you write another Application Interconnect command and Form command in OneWorld Scripting Tool to ensure that the command lines in the script pane match the application and form that are active in OneWorld.

When you move to a new form within the same application in OneWorld and click Form in the command menu, the Form list shows only the forms that are contained within that application.

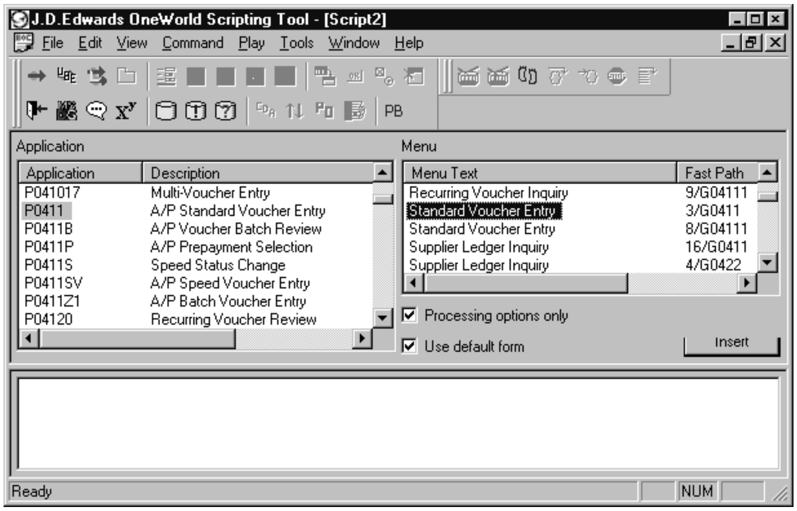
However, if you move to a form that is in a different application or outside of the normal cycle of transactions for the application, the name of that form does not appear in the list when you click Form in the command menu. When you click Application Interconnect in the command menu, you can choose from the command pane lists the new application and form that is active in OneWorld.

See Also

- *Form Command*
- *Setting the Context as a Form*
- *The Create "Work With Batch Versions" Commands Option*
- *The Custom Button Option*
- *Clicking a Custom Button*

Processing Options Command

You can use OneWorld Scripting Tool to set processing options for interactive versions of OneWorld applications you want to run. You set up the processing options as you like. During playback, OneWorld Scripting Tool tests whether the processing options are set in OneWorld as you scripted them. You script the processing options for an application and the interactive version attached to the menu item for the chosen application. To do so, you click Application in the command menu, choose an application and menu item, and then click the Processing options only option in the command pane.


J.D.Edwards OneWorld Scripting Tool - [Script2]
File Edit View Command Play Tools Window Help
PB

Application

Application	Description
P041017	Multi-Voucher Entry
P0411	A/P Standard Voucher Entry
P0411B	A/P Voucher Batch Review
P0411P	A/P Prepayment Selection
P0411S	Speed Status Change
P0411SV	A/P Speed Voucher Entry
P0411Z1	A/P Batch Voucher Entry
P04120	Recurring Voucher Review

Menu

Menu Text	Fast Path
Recurring Voucher Inquiry	9/G04111
Standard Voucher Entry	3/G0411
Standard Voucher Entry	8/G04111
Supplier Ledger Inquiry	16/G0411
Supplier Ledger Inquiry	4/G0422

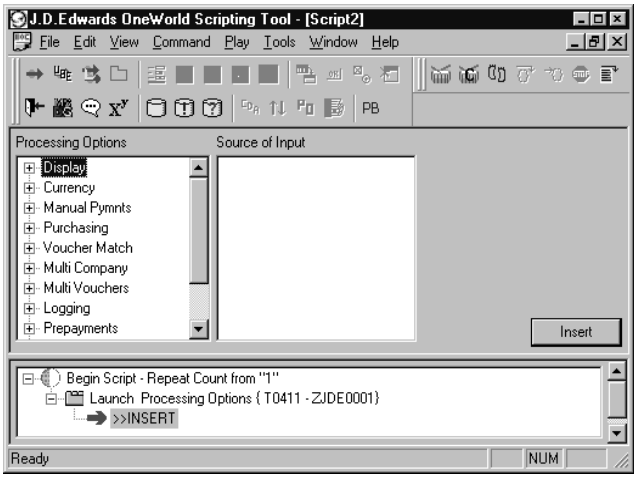
☑ Processing options only

☑ Use default form

Insert

Ready NUM

When you click the Insert button, the command pane displays the tabs with processing options for the application version you have chosen. With playback turned on, you can view the OneWorld Processing Options form and its tabs.


J.D.Edwards OneWorld Scripting Tool - [Script2]
File Edit View Command Play Tools Window Help
PB

Processing Options

- Display
- Currency
- Manual Pymnts
- Purchasing
- Voucher Match
- Multi Company
- Multi Vouchers
- Logging
- Prepayments

Source of Input

Insert

```
Begin Script - Repeat Count from "1"
  Launch  Processing Options { T0411 - ZJDE0001}
    >>INSERT
```

Ready NUM

OneWorld Scripting Tool serializes the processing option IDs when you create the script. When you load the script for playback, OneWorld Scripting Tool finds the matching processing option IDs in OneWorld and displays processing option text that is consistent with the OneWorld release against which you play your script.

If a new OneWorld release changes the processing option ID and the text, OneWorld Scripting Tool displays an error message in the processing options command line of the script pane when you play back the script. You can correct the processing option text in the command pane.

The Form Command

When you script an Application or an Application Interconnect command in OneWorld Scripting Tool, you select from both the Application list and the Menu list in the command pane. Your selection from the Menu list specifies the form and version that appears in OneWorld when you click the Insert button. The Form command line appears automatically in the script pane anytime you choose a OneWorld application and form from these lists and click the Insert button.

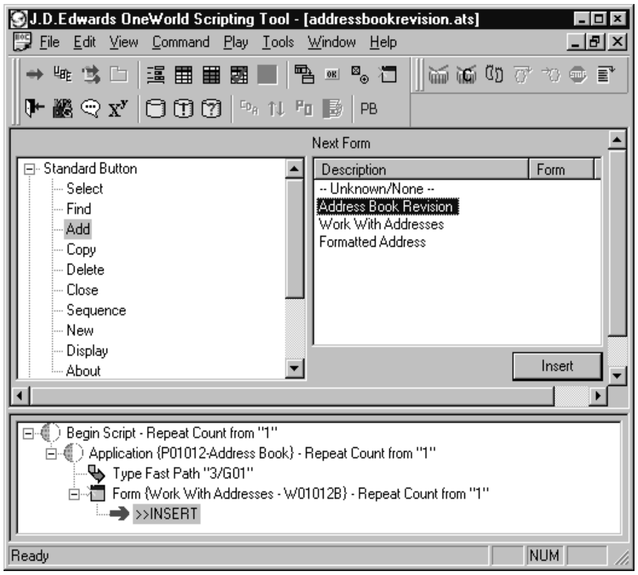
Scripting commands in OneWorld Scripting Tool requires that the Form command line in the script pane mirror the form that is active in OneWorld. You can make sure that the two mirror one another by choosing a form from the Next Form list in the command pane, by clicking Form in the command menu, or by clicking the Confirm Form button in the cool bar.

This topic discusses the following command pane lists related to the Form command:

- ❑ Next Form list
- ❑ Form list

Next Form List

The Next Form list helps you ensure that the Form command line in the script pane matches the form that is active in OneWorld. For example, you might decide to script pressing the Add button in a form such as Work With Addresses to move to another form such as Address Book Revisions. To do so, you click Press Toolbar Button in the command menu, choose Standard Button in the command pane, and choose Add from the tree. In the Next Form list, you can choose Address Book Revisions.



J.D.Edwards OneWorld Scripting Tool - [addressbookrevision.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

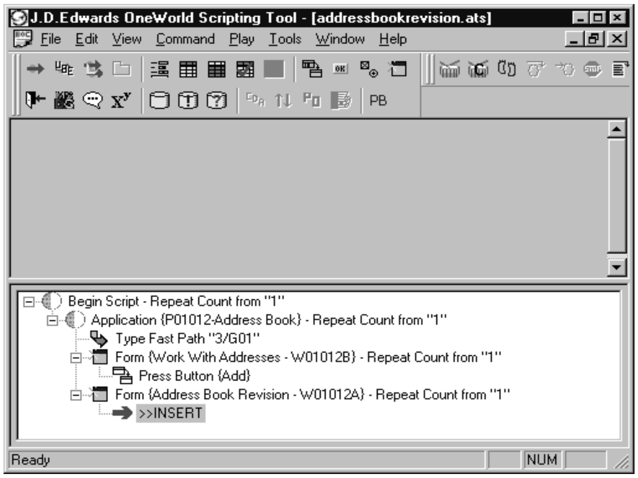
Next Form

Standard Button
- Select
- Find
- Add
- Copy
- Delete
- Close
- Sequence
- New
- Display
- About

Description	Form
-- Unknown/None --	
Address Book Revision	
Work With Addresses	
Formatted Address	

```
Begin Script - Repeat Count from "1"
  Application {P01012-Address Book} - Repeat Count from "1"
    Type Fast Path "3/G01"
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
      >>INSERT
```

You click Insert, OneWorld Scripting Tool inserts the Form command {Address Book Revisions}, and scripting can proceed.

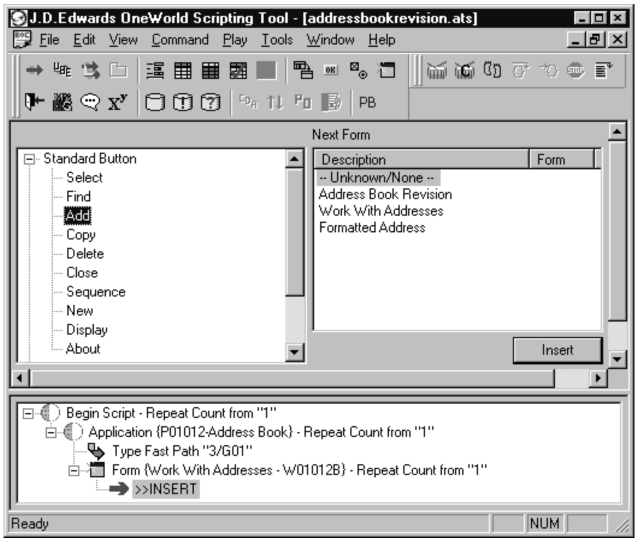


J.D.Edwards OneWorld Scripting Tool - [addressbookrevision.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
  Application {P01012-Address Book} - Repeat Count from "1"
    Type Fast Path "3/G01"
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
      Press Button {Add}
    Form {Address Book Revision - W01012A} - Repeat Count from "1"
      >>INSERT
```

Form List

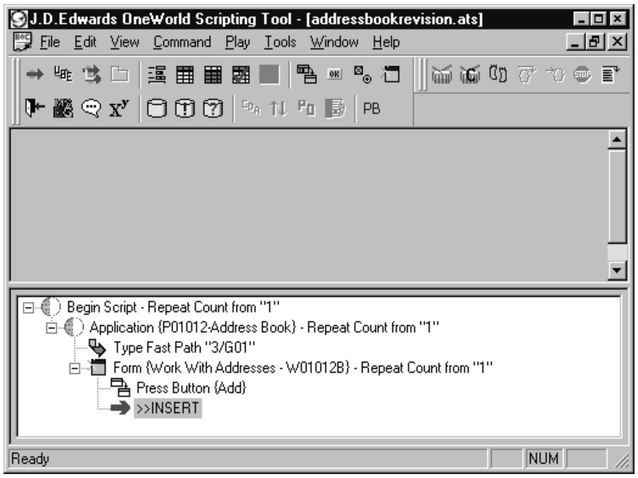
You might not know the form that appears next in OneWorld. Suppose you do not know that the form Address Book Revisions appears when you click Add in the Work With Addresses form. In this case, you can choose Unknown/None from the Next Form list.


J.D.Edwards OneWorld Scripting Tool - [addressbookrevision.ats]
File Edit View Command Play Tools Window Help
PB
Next Form
Standard Button
Select
Find
Add
Copy
Delete
Close
Sequence
New
Display
About
Description
Form
Insert
Ready
NUM

-- Unknown/None --
Address Book Revision
Work With Addresses
Formatted Address

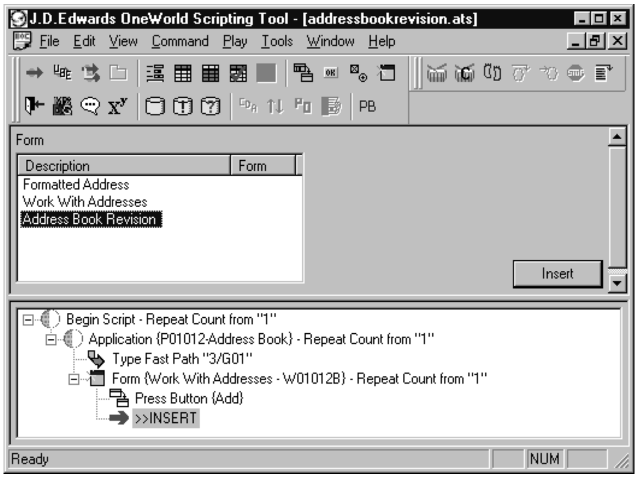
```
Begin Script - Repeat Count from "1"
  Application {P01012-Address Book} - Repeat Count from "1"
    Type Fast Path "3/G01"
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
      >>INSERT
```

However, note that since you chose Unknown/None, the Form command line in the script pane still shows {Work With Addresses}, while the active OneWorld form is Address Book Revisions. If you attempt to continue scripting at this point, OneWorld Scripting Tool gives you an error message.


J.D.Edwards OneWorld Scripting Tool - [addressbookrevision.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
  Application {P01012-Address Book} - Repeat Count from "1"
    Type Fast Path "3/G01"
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
      Press Button {Add}
      >>INSERT
```

To ensure that the Form command line mirrors the form that is active in OneWorld, you click Form in the command menu or click the Confirm Form button on the cool bar. In the command pane, a Form list appears, populated by the names of forms developed for the application you have selected. In this case, you choose Address Book Revisions from the Form list.



Formatted Address
Work With Addresses
Address Book Revision

```
Begin Script - Repeat Count from "1"
  Application {P01012-Address Book} - Repeat Count from "1"
    Type Fast Path "3/G01"
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
      Press Button {Add}
      >>INSERT
```

When you insert the command, the Form command line matches the form that is active in OneWorld, and you can proceed with scripting. You have confirmed that the form that is active in OneWorld matches the form name that appears in the script pane of the OneWorld Scripting Tool form.

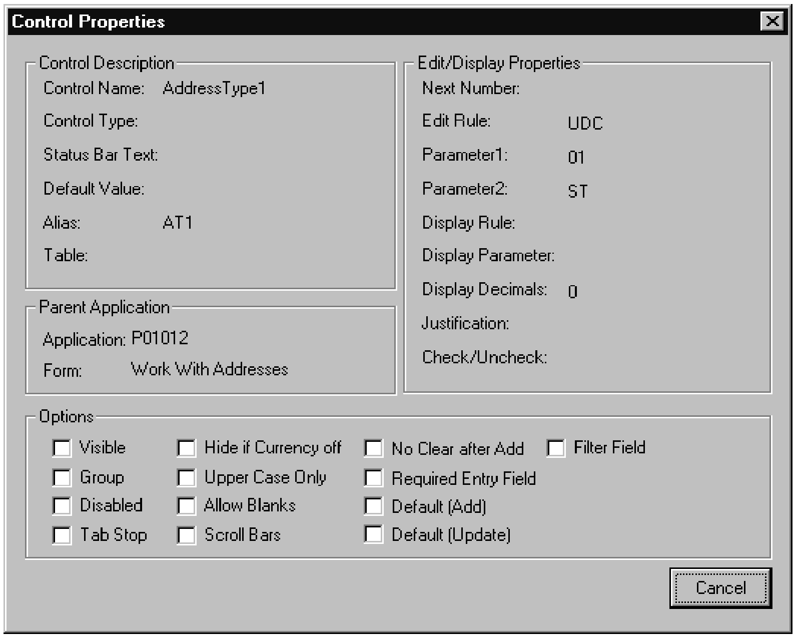
Header Command

The Header command establishes the header portion of a OneWorld form as the context in which additional commands, such as clicking buttons, entering control inputs and choosing options, can take place. You begin scripting the Header command by clicking Set Header Control Value in the command menu.

The header control list that appears in the command pane includes all the controls that are in the form that is active in OneWorld. You can elect to display hidden controls by clicking Tools in the menu bar, then clicking Options, clicking the Configure tab and choosing the Display Hidden Edit Controls box under Spec Selection Options.

You can also look at the properties of any header control if you right-click the name of the control in the Header Control list. When you do so, a Control Properties bar appears. When you click the bar, a form captioned Control Properties appears.

The form includes four sections: Control Description, Parent Application, Edit/Display Properties, and Options. To exit the form, press Cancel.



Control Properties

Control Description

Control Name: AddressType1

Control Type:

Status Bar Text:

Default Value:

Alias: AT1

Table:

Parent Application

Application: P01012

Form: Work With Addresses

Edit/Display Properties

Next Number:

Edit Rule: UDC

Parameter1: 01

Parameter2: ST

Display Rule:

Display Parameter:

Display Decimals: 0

Justification:

Check/Uncheck:

Options

□ Visible
□ Group
□ Disabled
□ Tab Stop
□ Hide if Currency off
□ Upper Case Only
□ Allow Blanks
□ Scroll Bars
□ No Clear after Add
□ Required Entry Field
□ Default (Add)
□ Default (Update)
□ Filter Field

Note that once you choose a header control, you make additional command pane choices, including a source of input to the control and the value of the input. When you click the Insert button, OneWorld Scripting Tool inserts two command lines to the script. The context command line, of course, is Header. However, by choosing a control, a source of input, and a value of the input, you have written an additional command. This command is the action command Type to, which appears in the script pane as a command line that shows the name of the control as well as the source of input and the value.

Grid Column Command

The Grid Column command establishes the grid column in a OneWorld form containing a grid as the context in which additional commands, such as pressing grid buttons and entering inputs to grid columns, can take place. You begin scripting the Grid Column command by clicking Set Grid Cell Value in the command menu.

The grid column list that appears in the command pane includes all the columns that are in the form that is active in OneWorld. You can elect to display hidden columns by clicking Tools in the menu bar, then clicking Options, clicking the Configure tab and choosing the Display Hidden Grid Columns box under Spec Selection Options.

You can also look at the properties of any grid column if you right-click the name of the control in the Grid Column list. When you do so a Control Properties bar appears. When you click the bar, a form captioned Control Properties appears. The form includes four sections: Grid, Column, Edit/Display Properties and Column Properties.

Note that once you choose a grid column, you make additional command pane choices, including a source of input to the control, and the value of the input. When you click the Insert button, OneWorld Scripting Tool inserts two command lines to the script. The context command line appears containing the words Detail Information. By choosing a grid column, a source of input, and a value of the input, you have written an additional command. This command is the action command Type to, which appears in the script pane as a command line that shows the name of the grid column as well as the source of input and the value.

QBE Command

The QBE command establishes the QBE line in a OneWorld form containing a grid as the context in which additional commands, such as entering inputs to the QBE line and pressing the Find button, can take place. You begin scripting the QBE command by clicking Set QBE Cell Value in the command menu. You then choose a grid column to which you want to type inputs.

Note that once you choose a grid column, you make additional command pane choices, including a source of input to the control, and the value of the input. When you click the Insert button, OneWorld Scripting Tool inserts two command lines to the script. The context command line appears containing the words QBE Information. By choosing a grid column, a source of input, and a value of the input, you have written an additional command. This command is the action command Type to, which appears in the script pane as a command line that shows the name of the grid column as well as the source of input and the value.

See Also

- *The Type to Command*
- *Scripting the Type to Command*

Writing the Script Using Context Commands

You can begin scripting context commands in one of three ways: by clicking on Command in the menu bar, by clicking on a hot key on the keyboard, or by clicking on a cool bar button. When you do so, lists appear in the command pane. You make selections from populated lists and enter information to unpopulated lists. When you click the Insert button, OneWorld Scripting Tool inserts a command line or lines into the script pane. The context command is identified in the script pane with words and symbols.

In general terms, the sequence you follow to write primary context commands is as follows:

- Choose a general context, such as an interactive application or UBE, by clicking the command menu, a hot key, or a cool bar button
- Specify a context, such as a particular application and menu item, by making choices from or entries to lists
- Click the Insert button to write the command to the script pane

Some context commands depend on other context commands. For example, Header is a context command, but you set the header as the context only after you have set an application and a form as the context for the script.

The general sequence you follow to write secondary commands is as follows:

- Choose a general context, such as a header, grid, or QBE line
- Specify a context, such as a control or grid column. These are determined by the application and form that you have previously chosen
- Choose a source of input to the specific context
- Choose a value to be input to the specific context

This chapter covers how you write context command with OneWorld Scripting Tool to accomplish the following tasks:

❑ Setting the context as an application

❑ Setting the context as a UBE

❑ Setting the context as an interconnected application

❑ Setting the context as a processing option

- ❑ Setting the context as a form
- ❑ Setting the context as a header
- ❑ Setting the context as a grid column
- ❑ Setting the context as a QBE line

Setting the Context as an Application

You often begin a OneWorld Scripting Tool session by launching an application. This establishes both the OneWorld application and form that you work with in your script. Note that the Menu list includes the text of the menu item in OneWorld Explorer, the Fast Path, and the application version. Different versions of the same OneWorld application may often be launched from different Explorer menus. Be sure to choose the menu item with version and processing options that you want to test.

▶ To set the context as an Application command

1. In the command menu of the OneWorld Scripting Tool form, click Application.
2. From the Application list in the command pane, click a OneWorld application.
3. From the Menu list in the command pane, click the name of a menu item.
4. Click the Insert button.

   OneWorld Scripting Tool inserts the Application, Fast Path and Form command lines into the script pane. In the playback mode, OneWorld Scripting Tool launches the specified version of a OneWorld interactive application. The chosen OneWorld form appears on the screen with the OneWorld Scripting Tool form, and you can move back and forth between the two.

Setting the Context as a UBE

There are several ways that you can set the context as a UBE. You can begin the script by launching the UBE from Work With Batch Versions, or you can launch an interactive application, then perform a report exit to Work With Batch Versions. You can launch an interactive application, then perform a row exit that launches a blind execution. You can launch a UBE from a menu that is hard-coded to submit the version automatically. Finally, you can launch a UBE that launches another UBE. In this case, OneWorld launches any subsequent UBEs then blindly submits them without any further intervention by OneWorld Scripting Tool in the script.

When you click UBE in the command menu, options for executing a Fast Path and for creating a Work With Batch Versions command appear. You can use these options to establish the way that OneWorld Scripting Tool submits the UBE, except when a menu is hard-coded to automatically submit it.

If you choose a UBE that is not automatically submitted, you must write a command to click the Submit button on the Version Prompting form in OneWorld. Before doing so, however, you can click the option that allows you to select data for your report. In some cases, after you submit the UBE and select data, you can set processing options. Doing so requires you to write a UBE Processing Options command to the script and set the options by making choices from the lists in the command pane.

Finally, you can choose the way to print the UBEs that you submit. You can instruct OneWorld Scripting Tool to wait for the UBE to print before resuming running the script, or you can send the UBE to print, but tell OneWorld Scripting Tool to continue running the script. If it is appropriate to the function you are testing, you can also write a command to close the Work With Batch Versions application and return to OneWorld Explorer.

This topic covers the following tasks:

❑ Launching a UBE

❑ Submitting a UBE

❑ Selecting data for a UBE

❑ Setting UBE processing options

❑ Printing a UBE

Launching a UBE

You can use OneWorld Scripting Tool to launch a UBE from a variety of contexts. You might begin your script by launching a UBE from a OneWorld menu. On the other hand, you might launch a UBE after you launch an interactive application. In this case, you might launch the UBE from a report menu in OneWorld, or you might launch it after you perform a row exit. You might also choose to launch a UBE that is automatically submitted. Finally, you can launch a UBE that in turn launches one or more additional UBEs.

This topic covers the steps for launching UBEs:

- From a OneWorld menu
- From a report menu
- From a row exit
- As an automatic submission

- From another UBE

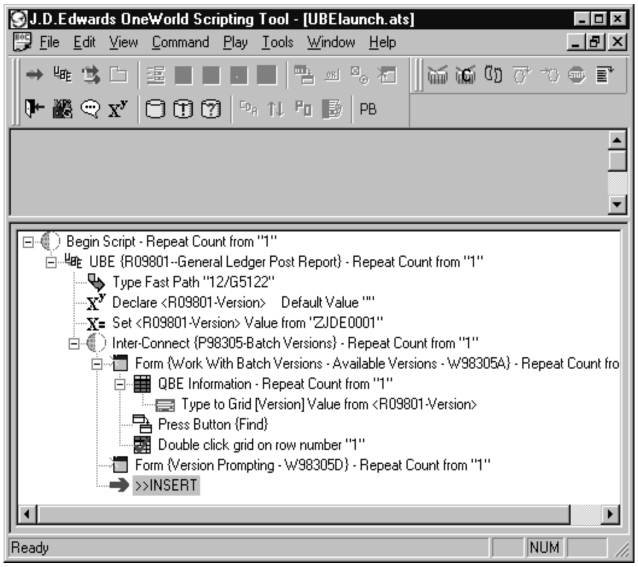
Launching a UBE from a OneWorld Menu

If you want to launch a UBE from a OneWorld menu, you must make choices from each of the three lists that appear in the command pane when you click UBE in the command menu: Application, Menu Item, and Version. You also click both of the options in the command pane: Execute FASTPATH and Create "Work With Batch Versions" commands. The first option establishes the Fast Path OneWorld Scripting Tool uses to access the UBE; the second option commands OneWorld Scripting Tool to automatically perform a QBE search in the Work With Batch Versions - Available Versions form for the UBE version that you chose. When you leave this option unchosen, you write the command to submit the UBE from the Version prompting form.

▶ To launch a UBE from a OneWorld menu

1. In the command menu of the OneWorld Scripting Tool form, click UBE.
2. In the command pane, make choices from the available lists:
    - UBE
    - Menu Item
    - Version
3. Click both the Execute FASTPATH and Create "Work With Batch Versions" commands options.
4. Click the Insert button.

OneWorld Scripting Tool automatically inserts a series of command lines to the script once you click the Insert button. The command sequence ends at the Form {Version Prompting} command line.


J.D.Edwards OneWorld Scripting Tool - [UBElaunch.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
  UBE {R09801--General Ledger Post Report} - Repeat Count from "1"
    Type Fast Path "12/G5122"
    Declare <R09801-Version>    Default Value ""
    Set <R09801-Version> Value from "ZJDE0001"
    Inter-Connect {P98305-Batch Versions} - Repeat Count from "1"
      Form {Work With Batch Versions - Available Versions - W98305A} - Repeat Count fro
        QBE Information - Repeat Count from "1"
          Type to Grid [Version] Value from <R09801-Version>
        Press Button {Find}
        Double click grid on row number "1"
      Form {Version Prompting - W98305D} - Repeat Count from "1"
      >>INSERT
```

Launching a UBE from a Report Menu

You might want to launch a UBE from the Reports pulldown menu of an interactive application. In this case, you begin the script by launching an interactive application. You use the Press Custom Button option to choose a report. You choose the UBE without the Execute FASTPATH option and the Menu Item list containing Fast Paths to the UBEs disappears. You choose the Create “Work With Batch Versions” commands option, meaning that you write the command to submit the UBE from the Version Prompting form.

▶ To launch a UBE from a report menu

1. From the command menu of the OneWorld Scripting Tool form, click Application.
2. In the command pane, make choices from the available lists:
   - Application
   - Menu
3. Click the Insert button.
4. In the command menu, click Press Toolbar Button.
5. In the Button list, click Press Custom Button.
6. Click Form/Row Exit.
7. Click Report.
8. Choose a report.
9. Click the Insert button.
10. In the command menu, click UBE.

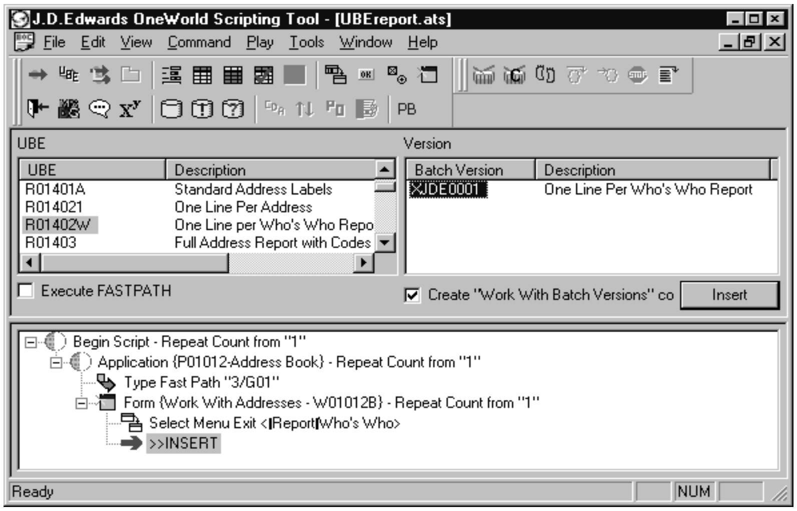
11. In the command pane, choose a UBE.

12. Do not choose the Execute FASTPATH option.

    Caution: If you do not click the Execute FASTPATH option, the Menu Item list disappears. Be sure to not to click this option until you have chosen the UBE. If you click the option before you choose the UBE, OneWorld Scripting Tool chooses the option again after you have chosen the UBE, and you must go through the steps again.

13. Choose a version from the Version list.

14. Click the Leave the Create “Work With Batch Versions” command option.


J.D.Edwards OneWorld Scripting Tool - [UBEreport.ats]
File Edit View Command Play Tools Window Help
PB

UBE

UBE	Description
R01401A	Standard Address Labels
R014021	One Line Per Address
R01402W	One Line per Who's Who Repo
R01403	Full Address Report with Codes

□ Execute FASTPATH

Version

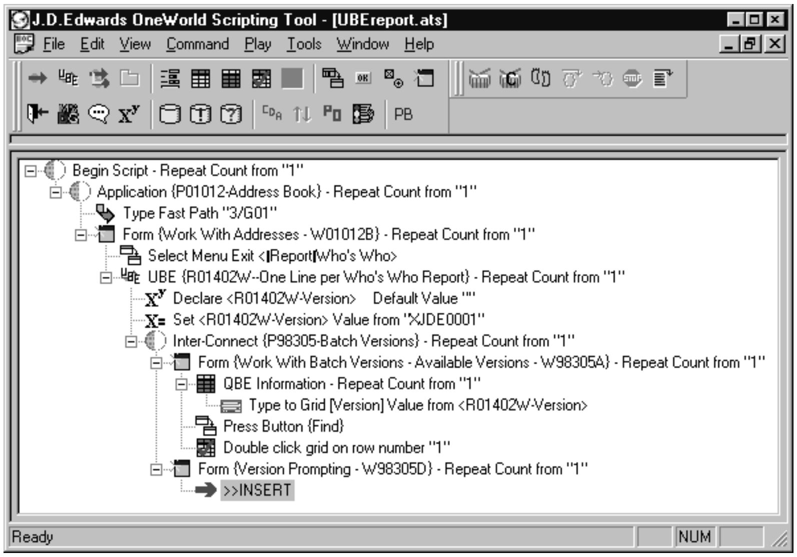
Batch Version	Description
XJDE0001	One Line Per Who's Who Report

☑ Create "Work With Batch Versions" co

```
Begin Script - Repeat Count from "1"
  Application {P01012-Address Book} - Repeat Count from "1"
    Type Fast Path "3/G01"
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
      Select Menu Exit <|Report|Who's Who>
      >>INSERT
```

15. Click the Insert button.

    Because you clicked the Create “Work With Batch Versions” commands option, OneWorld Scripting Tool automatically writes a series of script commands that ends at the Form {Version Prompting} command line.


J.D.Edwards OneWorld Scripting Tool - [UBEreport.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
  Application {P01012-Address Book} - Repeat Count from "1"
    Type Fast Path "3/G01"
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
      Select Menu Exit <|Report|Who's Who>
      UBE {R01402W--One Line per Who's Who Report} - Repeat Count from "1"
        Declare <R01402W-Version>    Default Value ""
        Set <R01402W-Version> Value from "XJDE0001"
        Inter-Connect {P98305-Batch Versions} - Repeat Count from "1"
          Form {Work With Batch Versions - Available Versions - W98305A} - Repeat Count from "1"
            QBE Information - Repeat Count from "1"
              Type to Grid [Version] Value from <R01402W-Version>
            Press Button {Find}
            Double click grid on row number "1"
          Form {Version Prompting - W98305D} - Repeat Count from "1"
            >>INSERT
```

Launching a UBE from a Row Exit

You can launch a UBE from a row exit in an interactive application. To do so, you begin by launching an interactive application. After you have written a row exit command using the Press Custom Button option, you choose the UBE. command. You click neither of the command pane options. The Menu Item list disappears, and you choose a version. If you do not choose the “Work With Batch Versions” option, OneWorld Scripting Tool blindly submits the UBE.

▶ To launch a UBE from a row exit

1. In the command menu of the OneWorld Scripting Tool form, click Application.
2. In the command pane, make choices from the available lists:
   - Application
   - Menu
3. Click the Insert button.
4. In the command menu, click Set QBE Cell Value.
5. In the command pane, make choices from the available lists:
   - Grid Column
   - Source of Input
   - Value selection
6. Click the Insert button.
7. In the command menu, click Press Toolbar Button.
8. In the Button list in the command pane, click Standard Button.

9. Choose Find.

   Caution: Be sure that you do not make a choice from the Next Form list. If you do, and insert the command, OneWorld Scripting Tool launches the new interactive application that you chose.

10. Click the Insert button.
11. In the command menu, click Select Grid Row.
12. In the command pane, click the following options:

- Click by row number
- Single click

13. In the Source of Row Number list, choose a value source.
14. In the value selection list, enter a row number or choose a variable or valid values list.
15. Click the Insert button.
16. In the command menu, click Press Toolbar Button.
17. Click Custom Button.
18. Click Form/Row Exit and Row.
19. Choose a row exit.

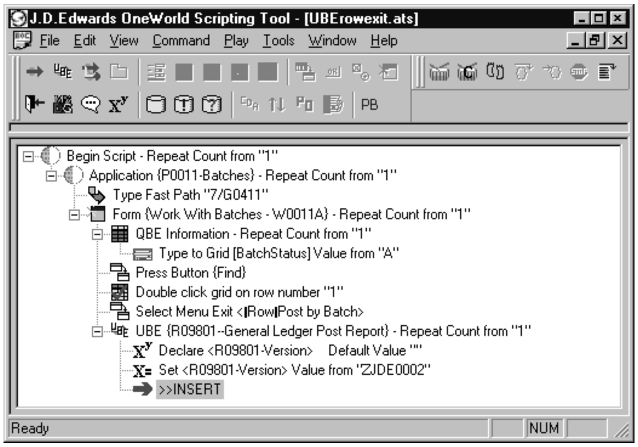
    Caution: Again, be sure that you do not make choices form the Application or Next Form lists because you do not want to launch an application. You want to launch a UBE.

20. Click the Insert button.
21. In the command menu, click UBE.
22. In the command pane, choose a UBE from the UBE list.

    Caution: Make sure that you do not choose either the Execute FASTPATH or the Create “Work With Batch Versions” commands options.

23. Choose a version from the Version list.
24. Click the Insert button.

    OneWorld Scripting Tool automatically submits the UBE.


J.D.Edwards OneWorld Scripting Tool - [UBErowexit.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
  Application {P0011-Batches} - Repeat Count from "1"
    Type Fast Path "7/G0411"
    Form {Work With Batches - W0011A} - Repeat Count from "1"
      QBE Information - Repeat Count from "1"
        Type to Grid [BatchStatus] Value from "A"
      Press Button {Find}
      Double click grid on row number "1"
      Select Menu Exit <|Row|Post by Batch>
      UBE {R09801--General Ledger Post Report} - Repeat Count from "1"
        Declare <R09801-Version>   Default Value ""
        Set <R09801-Version> Value from "ZJDE0002"
        >>INSERT
```

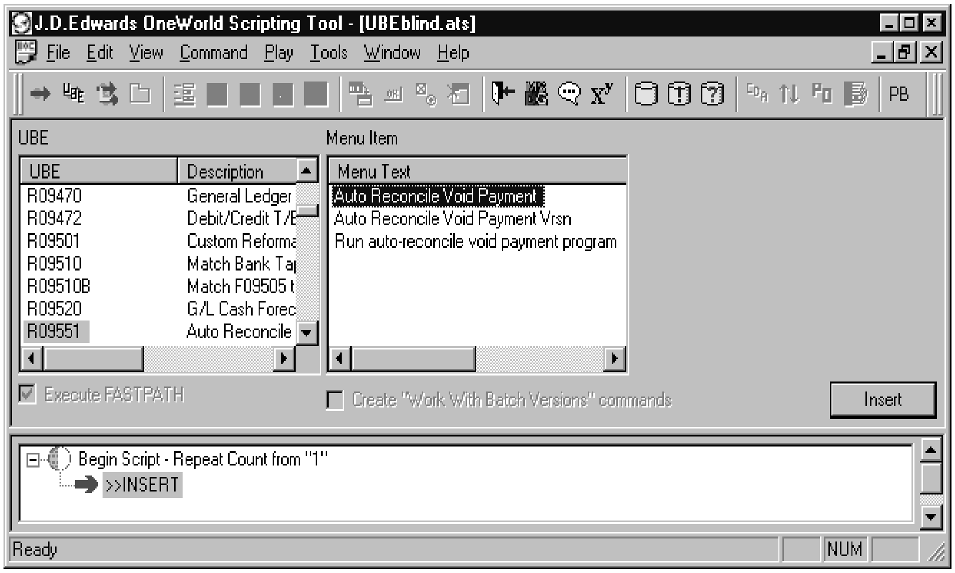
Launching a UBE That Is Automatically Submitted

If you launch a UBE that is hard-coded to submit the version automatically, you cannot click either of the options in the command pane. When you choose the UBE, OneWorld Scripting Tool disables both of the options and the Version list disappears. You choose from the Menu Item list, click the Insert button, and OneWorld Scripting Tool launches the UBE and blindly submits it.

▶ To launch a UBE that is automatically blindly submitted

1. In the command menu of the OneWorld Scripting Tool form, click UBE.
2. In the command pane, choose a UBE from the UBE list.

   When you choose the UBE in this scenario, OneWorld Scripting Tool disables both options and the Versions list disappears.


J.D.Edwards OneWorld Scripting Tool - [UBEblind.ats]
File Edit View Command Play Tools Window Help
PB
UBE
Menu Item
Insert
Ready
NUM

UBE	Description
R09470	General Ledger
R09472	Debit/Credit T/E
R09501	Custom Reforma
R09510	Match Bank Ta
R09510B	Match F09505 t
R09520	G/L Cash Forec
R09551	Auto Reconcile

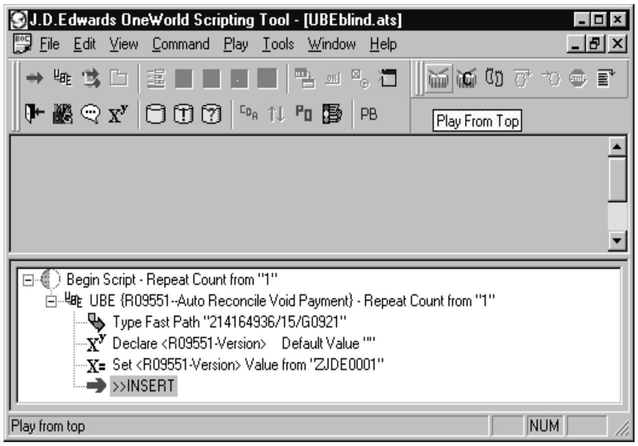
Menu Text
Auto Reconcile Void Payment
Auto Reconcile Void Payment Vrsn
Run auto-reconcile void payment program

☑ Execute FASTPATH

□ Create "Work With Batch Versions" commands

```
Begin Script - Repeat Count from "1"
  >>INSERT
```

3. Choose a menu item from the Menu Item list.
4. Click the Insert button.



J.D.Edwards OneWorld Scripting Tool - [UBEblind.ats]
File Edit View Command Play Tools Window Help
PB
Play From Top

```
Begin Script - Repeat Count from "1"
  UBE {R09551--Auto Reconcile Void Payment} - Repeat Count from "1"
    Type Fast Path "214164936/15/G0921"
    Declare <R09551-Version>    Default Value ""
    Set <R09551-Version> Value from "ZJDE0001"
    >>INSERT
```

Launching a UBE from Another UBE

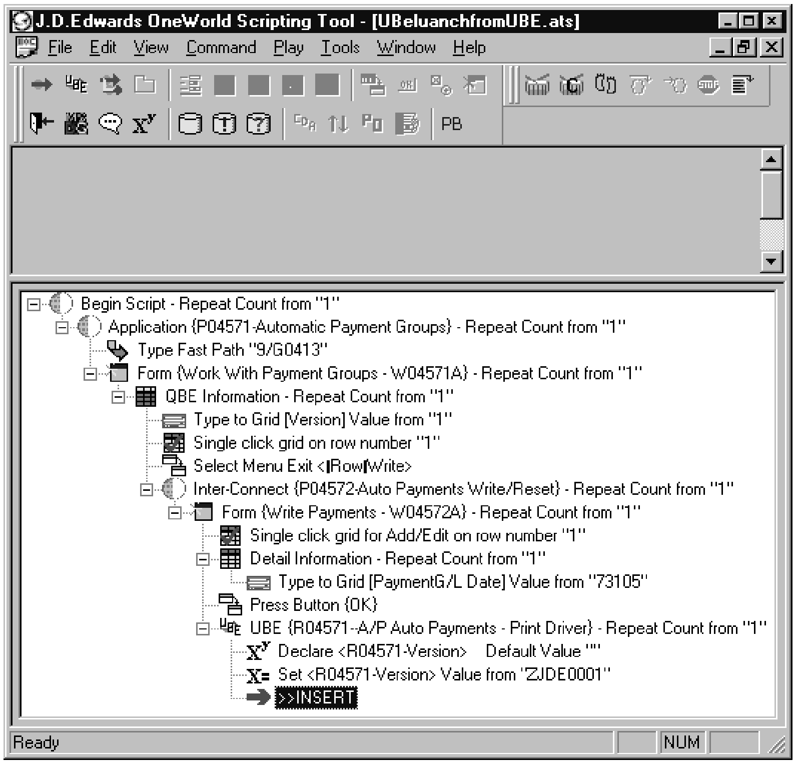
You might launch a UBE from a menu or from an application that in turn launches one or more subsequent UBEs. If this is the case, OneWorld automatically launches any UBEs called by the first one and blindly submits them. You do not choose versions or printing options, or set processing options. OneWorld carries out all the processes associated with the UBEs launched by the first one without further direction from the OneWorld Scripting Tool script.

▶ To launch a UBE from another UBE

1. In the command menu of the OneWorld Scripting Tool form, follow the steps for creating a script that launches a UBE from a OneWorld menu, report exit in an interactive application, or row exit in an interactive application.
2. In the command pane, choose a UBE from the UBE list.

   If the UBE is coded to launch another UBE, OneWorld Scripting Tool disables the Execute FASTPATH option and removes the Menu Item list from the command pane.

3. Choose a version from the Versions list.
4. Do not click the Create “Work With Batch Versions” command option.
5. Click the Insert button.


J.D.Edwards OneWorld Scripting Tool - [UBeluanchfromUBE.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
  Application {P04571-Automatic Payment Groups} - Repeat Count from "1"
    Type Fast Path "9/G0413"
    Form {Work With Payment Groups - W04571A} - Repeat Count from "1"
      QBE Information - Repeat Count from "1"
        Type to Grid [Version] Value from "1"
        Single click grid on row number "1"
        Select Menu Exit <|Row|Write>
        Inter-Connect {P04572-Auto Payments Write/Reset} - Repeat Count from "1"
          Form {Write Payments - W04572A} - Repeat Count from "1"
            Single click grid for Add/Edit on row number "1"
            Detail Information - Repeat Count from "1"
              Type to Grid [PaymentG/L Date] Value from "73105"
            Press Button {OK}
            UBE {R04571--A/P Auto Payments - Print Driver} - Repeat Count from "1"
              Declare <R04571-Version>    Default Value ""
              Set <R04571-Version> Value from "ZJDE0001"
              >>INSERT
```

Submitting a UBE

You are required to write a command to submit the UBE only when you have clicked the Create “Work With Batch Versions” commands option in the command pane. When you have done so, OneWorld Scripting Tool automatically writes commands that culminate with the Form {Version Prompting} command line. You then use the Press Standard Button option to write a command to press the Submit button in this form.

Remember that you can click the option for data selection in the Version Prompting form. Choosing this option allows you to design the criteria for you report. However, you should not click the option for data sequencing since OneWorld Scripting Tool currently cannot carry out this function.

▶ To submit a UBE

1. After you have written commands through the command line Form {Version Prompting}, go to the command menu and click Press Toolbar Button.
2. In the Button list in the command pane, click Standard Button.
3. Choose Submit.
4. Click the Insert button.

Selecting Data for a UBE

After you have launched a UBE, you might want to refine the data that appear in your report. If so, you can use OneWorld Scripting Tool's Criteria Design Aid feature, which you can access either by clicking UBE Selection in the command menu or by clicking the CDA button on the cool bar. This feature allows you to script entries to the OneWorld Data Selection form.

You can use the Criteria Design Aid feature when you launch a UBE with the Create "Work With Batch Versions" commands option clicked. If you launch the UBE from a OneWorld menu, OneWorld Scripting Tool automatically inserts a series of commands that ends at the From {Version Prompting} command line. If you launch the UBE from a report menu, you write a series of commands that culminates at the same point. In either case, however, when your script reaches the Form {Version Prompting} command line, you can write a command to click the Data Selection option and a command to submit the report for data selection.

At this point, you can click the UBE Selection command and use the OneWorld Scripting Tool command pane to script entries to the Data Selection form in OneWorld. OneWorld Scripting Tool allows you to insert as many lines to this form as you need. When you are finished, you can click the OK option in the command pane. Note that if you stored values in a variable earlier in your script, you can use these values in the right operand of the Data Selection form. You must, however, type the name of the variable into the Right Operand list of the command pane. In addition, the variable name must be enclosed in the "greater than" and "less than" symbols: <>.

After you have entered a variable to the right operand, OneWorld Scripting Tool displays options that you use to designate the value of the variable as a single value, a range of values, or a list of values.

Note that you use Criteria Design Aid in conjunction with writing a UBE command, not as a stand-alone command. In addition, some UBEs that you launch allow you to set processing options. If that is the case, you use OneWorld Scripting Tool to set the UBE processing options that you desire after you have selected the data that you want to appear in your report. You complete the UBE submission process by sending the report to the printer.

> **Caution:** Remember to enter the object name in the left operand list exactly as it appears in the drop-down menu of the list in the Data Selection form in OneWorld. Likewise, you enter an object name in the right operand list exactly as it appears in the drop-down menu of the list in the Data Selection form in OneWorld, unless you decide to enter a literal value. If you enter a literal value, you can enter a single value, multiple values, or a range of values. You separate multiple values with commas; you separate a value range with a hyphen.

▶ To select data for a UBE

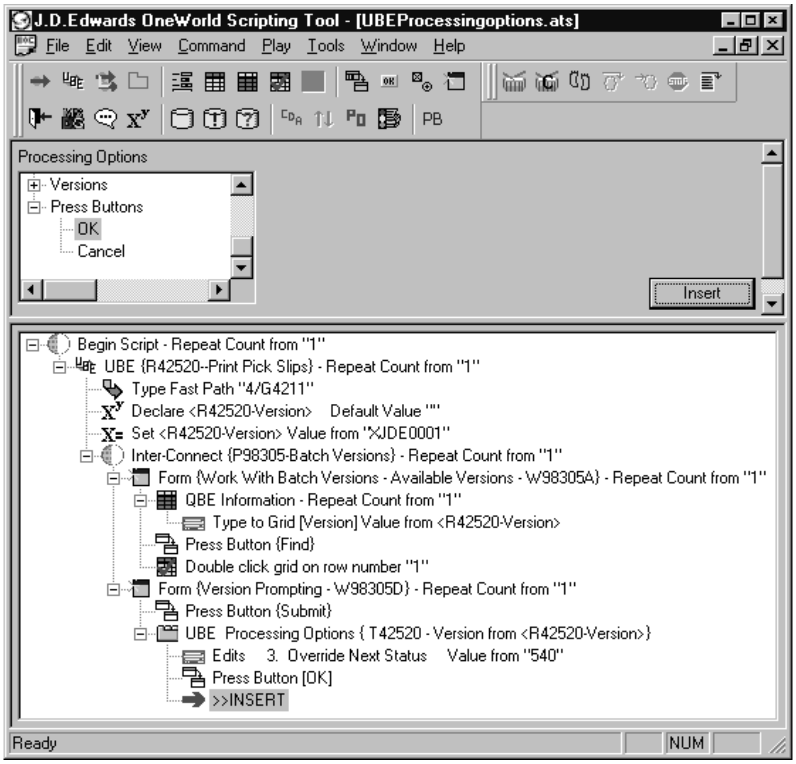
1. From the command menu of the OneWorld Scripting Tool form, click UBE.
2. In the command pane, make choices from the following lists:
   - UBE
   - Menu Item
   - Version
3. click both the Execute FASTPATH and Create "Work With Batch Versions" commands options.
4. Click the Insert button.
5. In the command menu, click Checkbox/Radio Button.
6. In the command pane, click DataSelectionYN in the Radio Button or Check Box list.
7. In the Source of Input list, click Check.
8. Click the Insert button.
9. In the command menu, click Press Toolbar Button.
10. From the Button list, choose Standard Button and Submit.
11. Click the Insert button.
12. In the command menu, click UBE Selection.
13. In the command pane, make entries to or choices from the available lists:
    - Line Number
    - Operator
    - Left Operand
    - Comparison
    - Right Operand
14. If you enter a variable to the Right Operand list, click one of the options that appears in the command pane in order to specify the type of value:
    - Single value
    - Range of values
    - List of values
15. Click the Insert button.
16. After you have written as many UBE Selection commands as you desire, click the Press OK option and click the Insert button.

Setting UBE Processing Options

After you submit a UBE version, you might see the OneWorld Processing Options form appear. In this case, you must set processing options for the UBE before you can print the report. To do so, you choose UBE Processing Options in the command menu, then make choices from the command pane.

▶ To set UBE processing options

1. In the command menu of the OneWorld Scripting Tool form, click UBE Processing Options.
2. In the Processing Options list of the command pane, click the node of a processing options tab.
3. Choose a processing option from the tree that unfolds.
4. Choose a source of value from the Source list.
5. If the value is literal, enter it in the unpopulated Literal Value list. If you choose Variable as the value source, OneWorld Scripting Tool populates the Variables list, which contains the UBE version you are working with as well as the names of any variables whose values you have set.
6. Click the Insert button.
7. In the command pane, click the Press Toolbar Buttons node in the Processing Options list.
8. If you are satisfied with the processing options you have set up, click OK. If you are not, click Cancel.
9. Click the Insert button.


J.D.Edwards OneWorld Scripting Tool - [UBEProcessingoptions.ats]
File Edit View Command Play Tools Window Help
PB
Processing Options
Versions
Press Buttons
OK
Cancel
Insert
Ready
NUM

```
Begin Script - Repeat Count from "1"
  UBE {R42520--Print Pick Slips} - Repeat Count from "1"
    Type Fast Path "4/G4211"
    Declare <R42520-Version>    Default Value ""
    Set <R42520-Version> Value from "XJDE0001"
    Inter-Connect {P98305-Batch Versions} - Repeat Count from "1"
      Form {Work With Batch Versions - Available Versions - W98305A} - Repeat Count from "1"
        QBE Information - Repeat Count from "1"
          Type to Grid [Version] Value from <R42520-Version>
        Press Button {Find}
        Double click grid on row number "1"
      Form {Version Prompting - W98305D} - Repeat Count from "1"
        Press Button {Submit}
        UBE  Processing Options { T42520 - Version from <R42520-Version>}
          Edits    3.  Override Next Status    Value from "540"
          Press Button [OK]
          >>INSERT
```

See Also

- *Setting the Context as a Processing Option*
- *Using a Variable as a Source of Input*

Printing a UBE

Clicking UBE Print in the command menu produces three options in the command pane. If you click the option Wait for UBE to complete before continuing, OneWorld Scripting Tool submits the UBE to the printer and waits for it to complete before resuming the script playback. If you do not click this option, OneWorld Scripting Tool continues with script playback without waiting for the UBE to print.

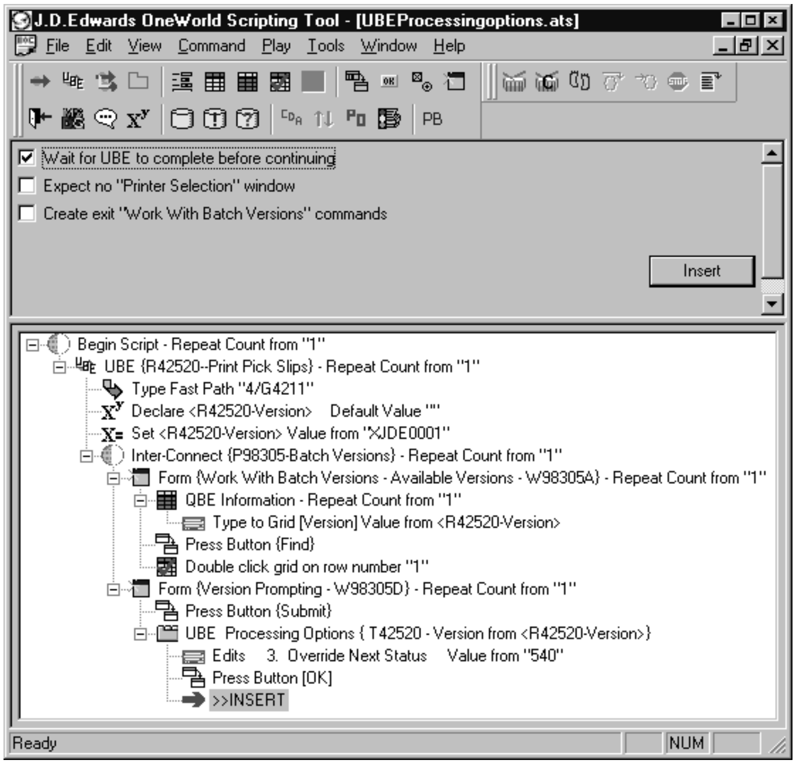
Clicking the second option, Create exit “Work With Batch Versions” commands, allows you to automatically write a Form command line for Work With Batch Versions - Available Versions and a Press Toolbar Button {Close} command to return to the form that was active previous to launching the UBE. You click this option only if you launched your UBE with the Create “Work With Batch Versions” commands option clicked.

If the UBE you run does not print, you click the Expect No “Printer Selection” Window option. This ensures that OneWorld Scripting Tool does not wait for a printer window to appear before resuming the script.

▶ To print a UBE

1. After you have submitted the UBE and set any necessary processing options, or if OneWorld Scripting Tool has blindly submitted the UBE, go to the command menu of the OneWorld Scripting Tool form and click UBE print.
2. In the command pane, click one or more desired options:
   - Wait for UBE to complete before continuing
   - Expect no "Printer Selection" window
   - Create exit "Work With Batch Versions"

   Caution: Remember that you click the Create exit "Work With Batch Versions" option only if you launched your UBE from Work With Batch Versions.


J.D.Edwards OneWorld Scripting Tool - [UBEProcessingoptions.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

☑ Wait for UBE to complete before continuing
□ Expect no "Printer Selection" window
□ Create exit "Work With Batch Versions" commands

```
Begin Script - Repeat Count from "1"
  UBE {R42520--Print Pick Slips} - Repeat Count from "1"
    Type Fast Path "4/G4211"
    Declare <R42520-Version>    Default Value ""
    Set <R42520-Version> Value from "XJDE0001"
    Inter-Connect {P98305-Batch Versions} - Repeat Count from "1"
      Form {Work With Batch Versions - Available Versions - W98305A} - Repeat Count from "1"
        QBE Information - Repeat Count from "1"
          Type to Grid [Version] Value from <R42520-Version>
        Press Button {Find}
        Double click grid on row number "1"
      Form {Version Prompting - W98305D} - Repeat Count from "1"
        Press Button {Submit}
        UBE  Processing Options { T42520 - Version from <R42520-Version>}
          Edits     3.  Override Next Status     Value from "540"
          Press Button [OK]
          >>INSERT
```

3. Click the Insert button.

Setting the Context as an Interconnected Application

In scripting a command to press a standard button, such as Add, you might exit from one OneWorld application to another. When this occurs, you must script an application interconnection by clicking Application Interconnect in the command menu or by clicking Press Toolbar Button and choosing the Press Custom Button option.

For example, you might want to write a script using the application 3B2002, Customer Ledger Inquiry. If you launch the application, choose the menu item Work With Customer Ledger Inquiry, then script pressing the Add button and Unknown/None from the Next Form list, OneWorld exits to a new application, P03B11, Invoice Entry. The menu item is Standard Invoice Entry.

If you click Form in the command menu, you notice that Standard Invoice Entry does not appear in the Form list in the command pane. This tells you that by pressing the Add button, you exited to another application in OneWorld. You cannot continue scripting until the Application and Form command lines in OneWorld Scripting Tool mirror the application and form that are active in OneWorld.

Using the Application Interconnect command, you can ensure that your script includes the new application and form in the script pane so that you can continue scripting. Remember that you use the Application Interconnect command *after* you have exited to a new application in OneWorld. If you want to script an application interconnect in OneWorld Scripting Tool *before* exiting to a new OneWorld application, you use the Press Custom Button option.

▶ To set the context as an interconnected application

1. From the menu bar in the OneWorld form to which you have exited, click Help.
2. Choose About OneWorld.
3. Note the application ID and form name.
4. click the OneWorld Scripting Tool form to return to OneWorld Scripting Tool for scripting.
5. In the command menu of the menu bar, Click Form.

   Note that Standard Invoice Entry does not appear in the Form list.

6. In the command menu, click Application Interconnect.
7. In the command pane, choose from the lists that appear:
   - Application (choose the application that is active in OneWorld)
   - Menu (choose the form that is active in OneWorld)
8. Click the Insert button.

   OneWorld Scripting Tool interconnects to the new application or form, and the Application and Form command lines in the script pane now mirror the application and form that are active in OneWorld. You can now script additional commands.

See Also

- *The Custom Button Option*
- *Pressing a Custom Button*

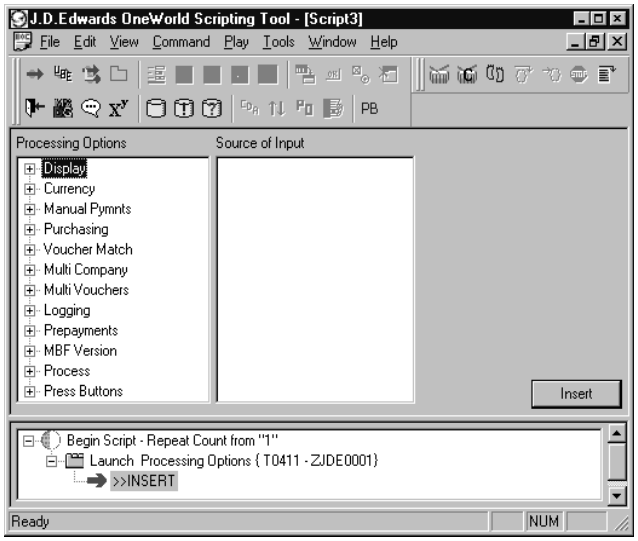
Setting the Context as a Processing Option

You may want to set processing options for a particular application before you begin writing secondary commands to the application. To do so, you choose an application and menu item from the command pane as if you are launching an application. However, before clicking the Insert button, you click the Processing options only option in the command pane. This allows you to choose processing options from lists in the command pane.

▶ To set the context as a processing option

1. In the command menu of the OneWorld Scripting Tool form, click Processing Options.
2. From the Application list in the command pane, click a OneWorld application.
3. From the Menu list in the command pane, click the name of a menu item.
4. In the command pane, click the Processing options only option.
5. Click the Insert button.

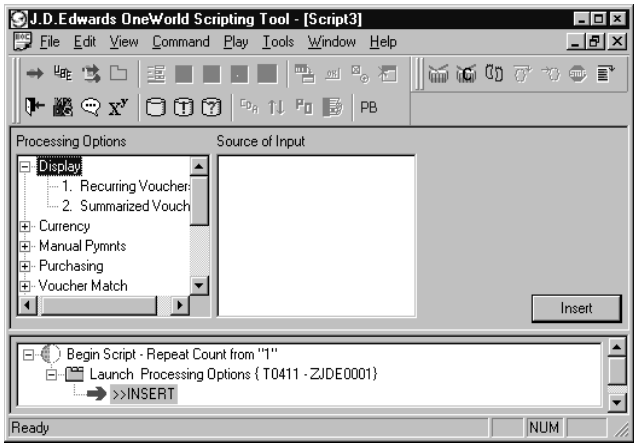
   The command pane now displays a list of the processing options tabs for the OneWorld application version you have chosen. In the script pane, the command line shows the Launch Processing Options symbol, the template for the application, and the version of the application that you chose.



Begin Script - Repeat Count from "1"
Launch Processing Options { T0411 - ZJDE0001}
>>INSERT

6. Click the node of one of the tabs.

   A tree opens, showing the processing options for the tab you have chosen.



Begin Script - Repeat Count from "1"
Launch Processing Options { T0411 - ZJDE0001}
>>INSERT

7. Choose a processing option.

   OneWorld Scripting Tool populates the Source list in the command pane with two sources of input: literal and variable.

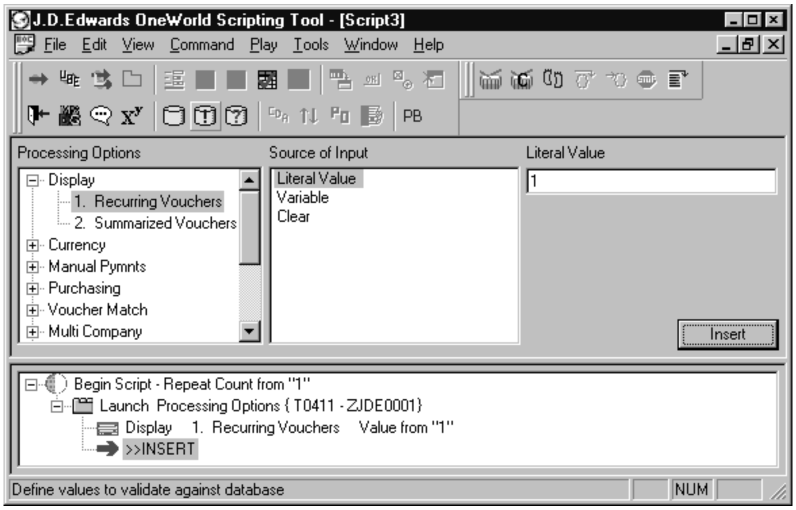
8. Choose a source of value.

   When you do so, a value selection list appears in the command pane.

9. If the value is literal, enter it in the unpopulated Literal Value list. If the value is a variable, OneWorld Scripting Tool populates the Variables list with the names of any variables whose value(s) you have set.

10. Click the Insert button.

OneWorld Scripting Tool enters to the script pane a command line that summarizes the processing option(s) you have chosen.



Literal Value: 1

```
Begin Script - Repeat Count from "1"
  Launch Processing Options { T0411 - ZJDE0001}
    Display   1. Recurring Vouchers   Value from "1"
    >>INSERT
```

Define values to validate against database

Note: With playback turned on, when you insert a processing option value in OneWorld Scripting Tool, that OneWorld Scripting Tool inserts the value to the corresponding control in the Processing Options form in OneWorld.

11. In the command pane, click the Press Toolbar Buttons node in the Processing Options list.
12. If you are satisfied with the processing options you have set up, click OK. If you are not, click Cancel.
13. Click the Insert button.

Note: You can insert as many processing options to the script as you wish. You can then launch the application, if you desire. When you do so, be sure not to click the Processing options only option in the command pane.

See Also

- *Using a Variable as a Source of Input*

Setting the Context as a Form

When you insert an Application command to the script, OneWorld Scripting Tool inserts to the script pane the name of the form that you selected from the menu list. Whenever you decide to move to a new OneWorld form, you must establish the context in OneWorld Scripting Tool, either by:

- Running the Form command using the Next Form list

- Running the Form command using the command menu

▶ To run the Form command using the Next Form list

1. In the command menu of the OneWorld Scripting Tool form, click a command that allows you to switch forms, for example, Press Toolbar Button.
2. In the Button list, click a button-pressing option that takes you to another OneWorld form, for example, Add.
3. In the Next Form list, click the name of the OneWorld form that appears next.

   The Next Form list contains the names of the OneWorld forms defined in the current application.

4. Click the Insert button.

▶ To run the Form command using the command menu

1. In the command menu of the OneWorld Scripting Tool form, click Form.

   The Form list appears in the command pane. It displays the names of all OneWorld forms defined in the current application.

   Note: You can also display the Form list in the command pane by clicking the Confirm Form button in the cool bar.

2. Choose a new form to be confirmed that matches the active form in OneWorld.
3. Click the Insert button.

   In the script pane, the new Form command line contains the name of the active form in OneWorld.

Setting the Context as a Header

Once you have launched an application and chosen a form in OneWorld, you can establish the header portion of the form as the context for further scripting. When you click the Set Header Control Value in the command menu, a Header Control list appears in the command pane, from which you can choose a specific control to further refine the context.

▶ To set the context as a header

1. In the command menu of the OneWorld Scripting Tool form, click Set Header Control Value.
2. In the Header list, choose a control to which you want to input data.
3. Choose a source of input from the Source of Input list.
4. In the value selection list, choose or input a value and click the Insert button.

   When you click the Insert button, OneWorld Scripting Tool writes both a Header context command line and a Type to action command line in the script pane.

Setting the Context as a Grid Column

Once you have launched an application and chosen a form in OneWorld, you can establish a grid column in the form as a context for further scripting.When you click Set Grid Cell Value in the command menu, a Grid Column list appears in the command pane, if the form that is active has a grid detail area. From this list, you can choose a specific column to further refine the context.

▶ To set the context as a grid column

1. In the command menu of the OneWorld Scripting Tool form, click Set Grid Cell Value.
2. In the Grid Column list, choose a grid column to which you want to input data.
3. Choose a source of input from the Source of Input list.
4. In the value selection list, choose or input a value and click the Insert button.

   When you click the Insert button, OneWorld Scripting Tool writes both a Grid (or Detail Information) context command line and a Type to action command line in the script pane.

Setting the Context as a QBE Line

The QBE line provides another context in which you can script commands once you have launched an application and chosen a form in OneWorld.

When you click Set QBE Cell Value in the command menu, a Grid Column list appears in the command pane, if the form that is active has a grid detail area. This list mirrors the Grid Column list that appears when you click the Set Grid

Cell Value in the cool bar. You can choose a specific column to further refine the context.

▶ To set the context as a QBE line

1. In the command menu of the OneWorld Scripting Tool form, click Set QBE Cell Value.
2. In the Grid Column list, choose a grid column to which you want to input data.
3. Choose a source of input from the Source of Input list.
4. In the value selection list, choose or input a value and click the Insert button.

   When you click the Insert button, OneWorld Scripting Tool writes both a Grid (or Detail Information) context command line and a Type to action command line in the script pane.

See Also

- *The Type to Command*
- *Scripting the Type to Command*

Scripting Actions

Scripting Actions

Action commands designate the specific actions the script performs, such as pressing buttons, choosing options, and entering data, within a context, such as an application or form. Action commands require a context, yet they are essential in that they specify the unique steps you want to take within the chosen environment. For example, you must write action commands in order to move between forms, enter data to header controls, grid columns, or QBE lines, select lines in a grid, perform database queries and updates, and so on.

Action commands also allow you to move in the midst of a script to a non-OneWorld application, such as Microsoft Excel. You do this by sending a message to your system via a command line in OneWorld Scripting Tool. You can also use the Command Line Message command to capture OneWorld windows and store for later use the images in a file that you designate. Thus you can maintain a file of images that you might wish to use in a document that you are creating.

You also use action commands when you want to build on scripts that you have written. For example, you can write an action command to include a previously created script within another script. You might include a stand-alone script that tests the entry of dates, for example, with another script that requires the entry of dates, but which then goes on to test other functions.

When you want to play back a script that you have created, you can utilize action commands to customize the playback. For example, you can insert a Wait command in the script. This command tells OneWorld Scripting Tool to wait the amount of time that you specify, at the point in the script you specify, before proceeding with playback. In addition, you can insert comments to the script that identify what it is testing or that communicate to another person who might run the script what occurred at a particular point during playback.

After you have scripted the entry of data in OneWorld forms, you might want to verify, or validate, that these data have indeed been entered in the database that you specified. OneWorld Scripting Tool offers another action command, database validation, that enables you to do that.

In sum, then, action commands help you to:

- Build scripts that test a specific set of OneWorld processes
- Test whether data you have entered during the course of scripting has been properly entered to the OneWorld database
- Modify and comment on scripts you have already created

- Customize the way your script runs
- Move outside OneWorld and utilize other applications whose functions you can use to run your script or draw on to assist you in other tasks, such as documentation.

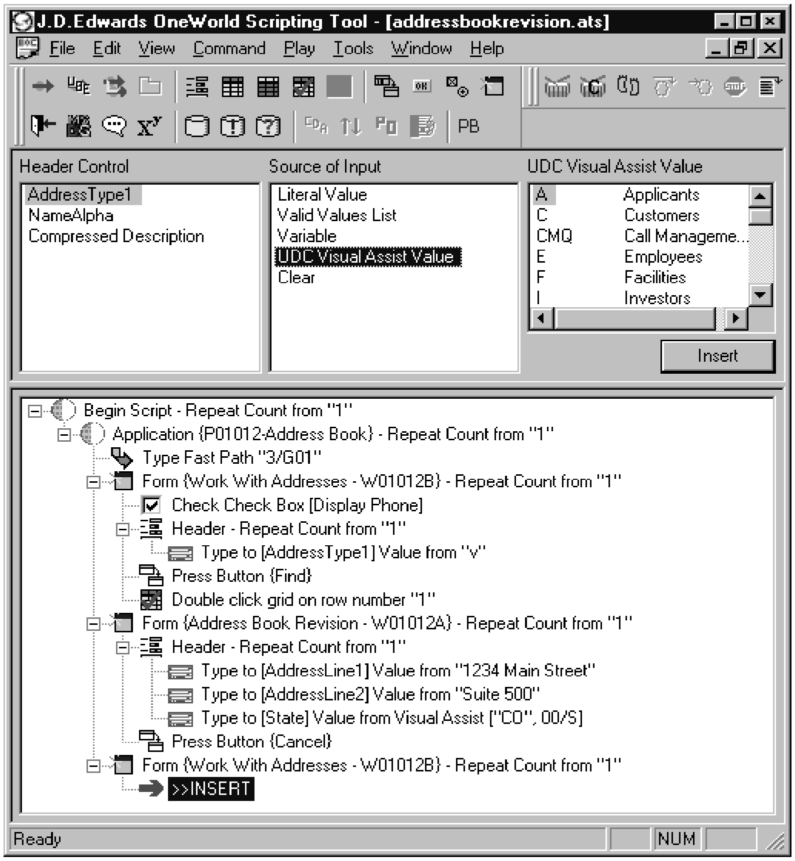
This section discusses the following action commands and the steps you follow to script each one:

- ❑ The Type to command
- ❑ Scripting the Type to command
- ❑ The Select Grid Line command
- ❑ Scripting the Select Grid Line command
- ❑ The Press Toolbar Button command
- ❑ Scripting the Press Toolbar Button command
- ❑ The Press Push Button command
- ❑ Scripting the Press Push Button command
- ❑ The Select Item in Combo Box command
- ❑ Scripting the Select Item in Combo Box command
- ❑ The Build Tree Path command
- ❑ Scripting the Build Tree Path command
- ❑ The Database Validation command
- ❑ Scripting the Database Validation command
- ❑ The Command Line Message command
- ❑ Scripting the Command Line Message command

The Type to Command

You use the Type to command to script inputs to header controls, grid columns, or QBE lines in a OneWorld form. Unlike some action commands, such as Press a Button, no command menu item or cool bar button represents the Type to command. To write it, you use lists to specify the context as a header, grid, or QBE line, choose a specific header control or grid column designate a source of input to the control or column, and choose a value to input to the control or column. The value can be literal or you can derive it from a variable, valid values list, UDC visual assist, or a form interconnect visual assist.

The lists that you use to write the Type to command appear in the command pane of the OneWorld Scripting Tool form.


J.D.Edwards OneWorld Scripting Tool - [addressbookrevision.ats]
File
Edit
View
Command
Play
Tools
Window
Help
PB
Insert
Ready
NUM

Header Control: AddressType1, NameAlpha, Compressed Description

Source of Input: Literal Value, Valid Values List, Variable, UDC Visual Assist Value, Clear

UDC Visual Assist Value	
A	Applicants
C	Customers
CMQ	Call Manageme...
E	Employees
F	Facilities
I	Investors

```
Begin Script - Repeat Count from "1"
  Application {P01012-Address Book} - Repeat Count from "1"
    Type Fast Path "3/G01"
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
      Check Check Box [Display Phone]
      Header - Repeat Count from "1"
        Type to [AddressType1] Value from "v"
      Press Button {Find}
      Double click grid on row number "1"
    Form {Address Book Revision - W01012A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [AddressLine1] Value from "1234 Main Street"
        Type to [AddressLine2] Value from "Suite 500"
        Type to [State] Value from Visual Assist ["CO", 00/S]
      Press Button {Cancel}
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
      >>INSERT
```

This chapter defines the three lists you work with to write the Type to command:

- ❑ Header Control or Grid Column list
- ❑ Source of Input list
- ❑ Value selection list

The Header Control or Grid Column List

The Header Control and Grid Column lists are populated with alpha descriptions of the data dictionary items located in the header, grid, and QBE portions of the OneWorld form, which is the context you have set. You click a control or column to which you want to script an input.

Source of Input List

After you choose a header control or grid column, the Source of Input list gives you a choice sources from which to get a value to input to the header control or grid column:

- ❑ Literal value
- ❑ Valid values list
- ❑ Variable
- ❑ UDC visual assist value
- ❑ Form interconnect visual assist
- ❑ Clear source of input

Literal Value

When you choose Literal Value as a source of input, you specify that an entry in a control, grid column, or QBE line of a OneWorld form appears exactly as it displays in the value selection list. For example, the literal value of a NameAlpha control entry might be "John Q. Public," which is exactly what OneWorld Scripting Tool should enter in the header control of a OneWorld form when the script runs.

Valid Values List

Entry of a literal value assigns only one value to a header control, grid column, or QBE line in a OneWorld form. Choosing Valid Values List as a source of input enables you to create a text or numeric file that can contain multiple values, any of which you can enter to the header control, grid column, or QBE line. You can create a valid values list either by assigning your own values or by choosing a database and querying it for values to include in the list.

You might choose Valid Values List as a source of input when you want to run a script multiple times with a different value entered each time to a specified header control, grid column, or QBE line. As OneWorld Scripting Tool loops through the script you have created, the value it enters to the control or column changes, reflecting the values included in the list. Alternatively, you might want to run a script once, but enter five different values to a grid column. Again, creating a single valid values list that contains five items enables you to do this.

If you exit a script, exit OneWorld Scripting Tool, then open the script again, OneWorld Scripting Tool resets the valid values list so that when you play back the script, the first value you entered to the list appears first. If you exit a script without exiting OneWorld Scripting Tool, then open the script again, OneWorld Scripting Tool uses the value that was next in order in the list when you exited the script.

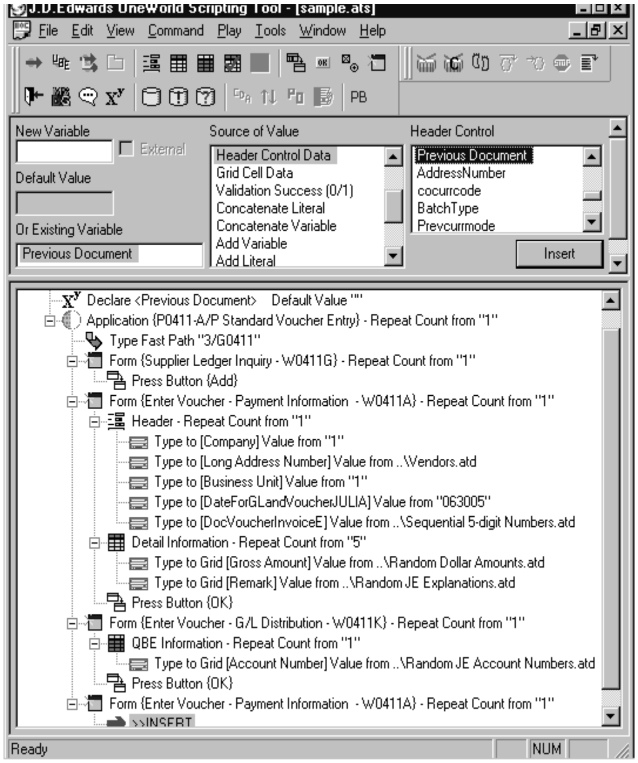
Variable

You might want to choose a value, store it, then use it at a later point in the script. You might also want to use the value more than once in the script. If that is the case, you use as a source of input a variable, the value of which you store and insert in the script at any point you want. You declare the variable to assign it a name. Then you set and store its value, which you can get from a valid values list, header control, grid column, another variable, or from your literal input.

The command pane in the OneWorld Scripting Tool form displays six components when you write a variable command:

- New variable list: You enter the name of the variable in this list, thereby declaring the variable.
- External variable option: By choosing this option, you specify that the variable can be linked to a variable in another script so that its value can be passed between scripts.
- Default variable list: You can enter a value that OneWorld Scripting Tool uses even if you do not set a value for the variable.
- Existing variable list: OneWorld Scripting Tool displays the names of any existing variables that you have declared in the script.
- Source of value list: You choose a source of value for the variable, such as a literal value, a valid values list, a header control, a grid cell, or another variable. In addition, you can choose variable manipulations, such as adding or subtracting a literal value or a variable value from the variable.
- Value selection list: You enter a literal value or choose the object that contains the value that you store in the variable. For example, if you chose a header control as the source of value, you choose from the value

selection list the specific control that contains the value. You can also draw variable values from valid values lists, from variables to which you have previously assigned a value, or from OneWorld sources, such as error and warning messages or grid row counts.



New Variable:

□ External

Default Value:

Or Existing Variable: Previous Document

Source of Value:
- Header Control Data
- Grid Cell Data
- Validation Success (0/1)
- Concatenate Literal
- Concatenate Variable
- Add Variable
- Add Literal

Header Control:
- Previous Document
- AddressNumber
- cocurrcode
- BatchType
- Prevcurrmode

```
Declare <Previous Document>    Default Value ""
Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
  Type Fast Path "3/G0411"
  Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
    Press Button {Add}
  Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
    Header - Repeat Count from "1"
      Type to [Company] Value from "1"
      Type to [Long Address Number] Value from ..\Vendors.atd
      Type to [Business Unit] Value from "1"
      Type to [DateForGLandVoucherJULIA] Value from "063005"
      Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
    Detail Information - Repeat Count from "5"
      Type to Grid [Gross Amount] Value from ..\Random Dollar Amounts.atd
      Type to Grid [Remark] Value from ..\Random JE Explanations.atd
    Press Button {OK}
  Form {Enter Voucher - G/L Distribution - W0411K} - Repeat Count from "1"
    QBE Information - Repeat Count from "1"
      Type to Grid [Account Number] Value from ..\Random JE Account Numbers.atd
    Press Button {OK}
  Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
    >>INSERT
```

The following table summarizes other key terms that relate to the use of variables in OneWorld Scripting Tool scripts:

Variable Term	Meaning/Application
Variable scope	The range of commands within a script that the value of a variable can be used.
Global variable	A variable whose value can be used throughout an entire script.
Local variable	A variable whose value can be used only within a portion of a script.
External variable	A variable that can be linked to a variable in another script so that a variable value can be passed between scripts.

Default value A value you assign to a variable that is used if you do not set the value of the variable elsewhere in the script.

Conditional statement An If/then statement that you write by comparing the values of two variables. The statement stipulates that if a condition exists in the script, then the script should run other commands.

Variable concatenation The stringing together of two or more variables to create a new variable.

System variable A variable whose value is derived from OneWorld data, such as error and warning messages.

Valid values count A variable whose value is derived from the number of items in a valid values list.

Variable watch list A list that keeps track of variable values that are used during script playback.

Validation success A variable whose value indicates the success or failure of a database validation.

This topic further discusses each of these terms:

- □ Variable scope
- □ Local variables
- □ Global variables
- □ External variables
- □ Default values for variables
- □ Conditional statements
- □ Variable addition
- □ Variable subtraction
- □ Variable concatenation
- □ System variables
- □ Valid values count

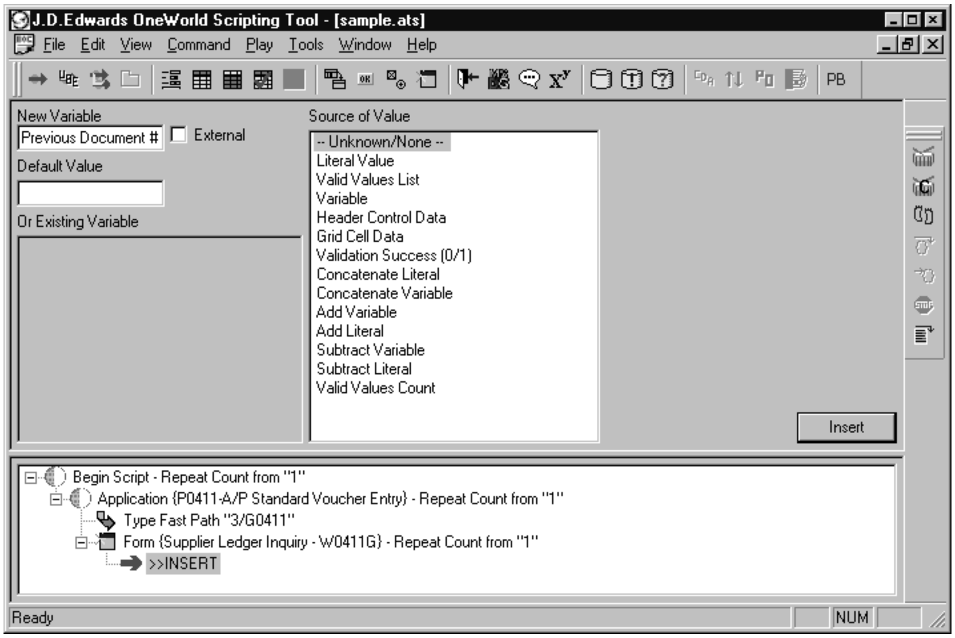
- ❑ Variable watch list
- ❑ Validation success

Variable Scope

Variable scope refers to how broadly you can use the value of the variable within your script. You create a node each time you write a context command. The node in which you declare a variable determines its scope. For example, if you declare a variable within an Application command node, the variable's scope extends only within that node, and you can use a value you set for the variable only within that node. If, for example, you declare a variable within an Application command node, then launch another application, you cannot use the value that you set for this variable within the new Application command node. If you declare a variable within a Form command node, its scope extends only within that form.

Local Variables

You can use the value of a local variable only within a portion of the script, specifically the node to which it is attached. You might, for example, declare the variable immediately after you launch an application and a form.



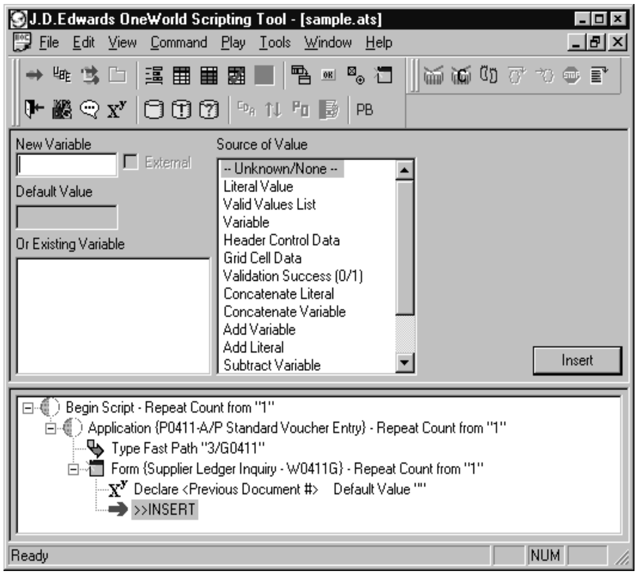
J.D.Edwards OneWorld Scripting Tool - [sample.ats]

New Variable: Previous Document #
☐ External
Default Value:
Or Existing Variable:
Source of Value: -- Unknown/None --

```
⊟ Begin Script - Repeat Count from "1"
   ⊟ Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
      Type Fast Path "3/G0411"
      ⊟ Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
         >>INSERT
```

Ready

In this instance, you could set the variable's value and use the value only for any command lines you script within the Form command node because the Declare variable command you write is a child of the Form command node.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

New Variable □ External

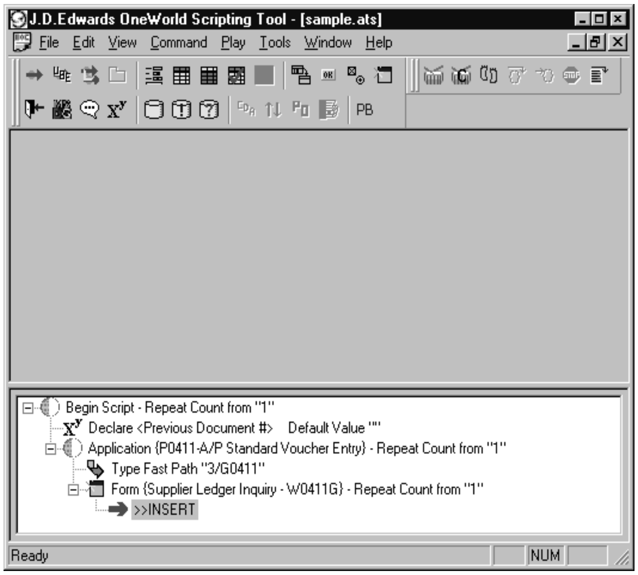
Default Value

Or Existing Variable

Source of Value
- -- Unknown/None --
- Literal Value
- Valid Values List
- Variable
- Header Control Data
- Grid Cell Data
- Validation Success (0/1)
- Concatenate Literal
- Concatenate Variable
- Add Variable
- Add Literal
- Subtract Variable

```
Begin Script - Repeat Count from "1"
    Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
        Type Fast Path "3/G0411"
        Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
            Declare <Previous Document #>    Default Value ""
            >>INSERT
```

You can expand the variable's scope by dragging it to another node higher in the script, and dropping it in that node. For example, you might drag the Declare Variable command from the Form command node to the Application command node and drop it, making it a child of the Application command. Now you have broadened the scope of the variable, and you can use the value you set for it anywhere within the application.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
    Declare <Previous Document #>    Default Value ""
    Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
        Type Fast Path "3/G0411"
        Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
            >>INSERT
```

However, the variable's scope is still local. If you launch another application later in the script, you cannot use the value of the variable within that new Application command, unless you make this command a child of the first Application command.

See Also

- *Changing the Scope of a Variable*
- *Indented Nodes*
- *Drag and Drop*

Global Variables

A variable's scope is global when you can use its value throughout the entire script. To establish global scope for a variable, you must make the Declare variable command a child of the Begin Script node, which is always the first node in the script.

External Variables

A variable with global scope allows you to pass a value you set to header controls, grid columns, and QBE lines throughout a single, stand-alone script. OneWorld Scripting Tool also allows you to declare a variable as external, which means that you want to link the variable to a variable in another script. You use external variables when you want to pass variable values between scripts. For example, you might store a batch number in one script. If you declare the variable that stores the batch number as external, you can link that variable to external variables in one or more other scripts, and pass the batch number value along for use in the other scripts.

See Also

- *Variable Linking between Scripts*
- *Creating Variable Links*
- *Script Includes*
- *Including Scripts*

Default Values for Variables

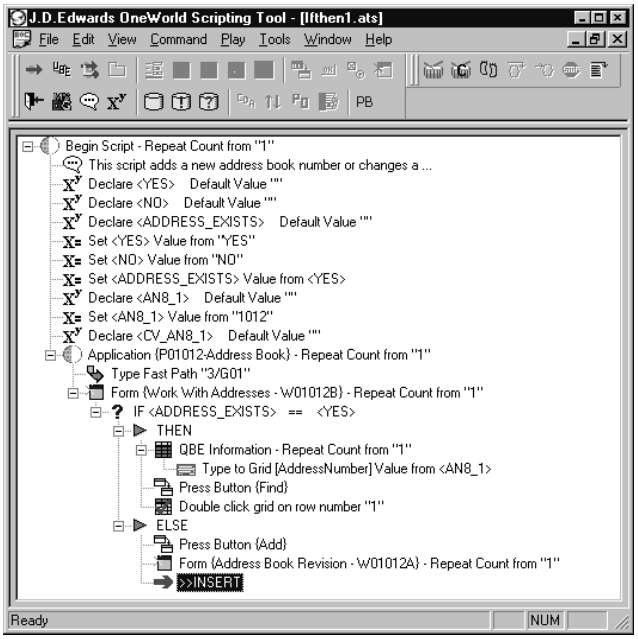
You might want to create a script that you can play both in stand-alone mode and with other scripts. To do so, you assign a default value to your variable. For example, you might create a Script B that links to Script A, which passes along a batch number value. Suppose, however, that you want to play Script B by itself. If you set a default value in Script B, OneWorld Scripting Tool uses that value when you play Script B in stand-alone mode.

You can also assign a default value to a variable that you do not declare as external. OneWorld Scripting Tool uses the default value of the variable each time that you use it as a source of input to a header control, grid column, or QBE line. If you write a command to set a value for the variable, that value overrides the default value.

Conditional Statements

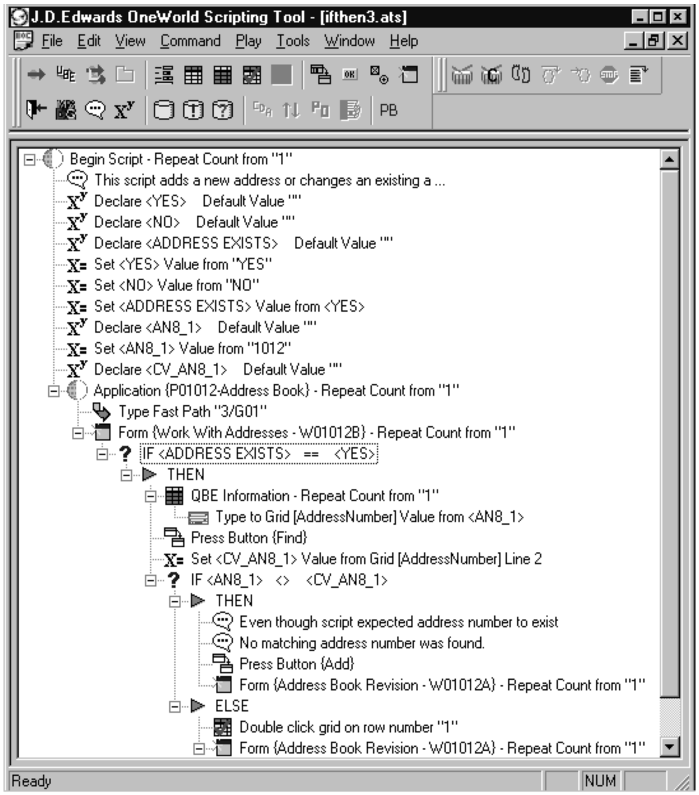
Conditional statements allow you write If/Then/Else commands that compare the values of two variables whose names you have declared and whose values you have set. If the script meets the conditions set forth in the statement, then OneWorld Scripting Tool runs an additional branch of the script. On the other hand, you can write commands connected to an Else branch in the script or you allow the script to end if the script does not meet the condition set forth in the statement.

You might write a conditional statement to ensure that your script tests an application even if the script does not meet the conditions that you expected to exist. For example, you might want to test making revisions to an existing Address Book number. If the Address Book number exists, then OneWorld Scripting Tool selects the grid line in the Work With Addresses form, double clicks the line, then makes revisions to the existing Address Book number in the Address Book Revisions form. However, if the Address Book number does not exist, your script fails unless you write a conditional statement that stipulates that if the Address Book number does not exist, OneWorld Scripting Tool should add a form and run commands to create a new Address Book entry.


J.D.Edwards OneWorld Scripting Tool - [Ifthen1.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
    This script adds a new address book number or changes a ...
    Declare <YES>    Default Value ""
    Declare <NO>    Default Value ""
    Declare <ADDRESS_EXISTS>    Default Value ""
    Set <YES> Value from "YES"
    Set <NO> Value from "NO"
    Set <ADDRESS_EXISTS> Value from <YES>
    Declare <AN8_1>    Default Value ""
    Set <AN8_1> Value from "1012"
    Declare <CV_AN8_1>    Default Value ""
    Application {P01012-Address Book} - Repeat Count from "1"
        Type Fast Path "3/G01"
        Form {Work With Addresses - W01012B} - Repeat Count from "1"
            IF <ADDRESS_EXISTS>  ==   <YES>
                THEN
                    QBE Information - Repeat Count from "1"
                        Type to Grid [AddressNumber] Value from <AN8_1>
                    Press Button {Find}
                    Double click grid on row number "1"
                ELSE
                    Press Button {Add}
                    Form {Address Book Revision - W01012A} - Repeat Count from "1"
                    >>INSERT
```

Another conditional statement might test the converse: if Address Book number does not equal the number that OneWorld Scripting Tool returns to the QBE line of the Work with Addresses form, the Address Book number does not exist. If that condition is met, OneWorld Scripting Tool adds a form and creates a new entry. If the Address Book number does exist, OneWorld Scripting Tool double clicks the grid line and makes revisions to Address Book entry. This is the “else” portion of the statement.


J.D.Edwards OneWorld Scripting Tool - [ifthen3.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
  This script adds a new address or changes an existing a ...
  Declare <YES>   Default Value ""
  Declare <NO>   Default Value ""
  Declare <ADDRESS EXISTS>   Default Value ""
  Set <YES> Value from "YES"
  Set <NO> Value from "NO"
  Set <ADDRESS EXISTS> Value from <YES>
  Declare <AN8_1>   Default Value ""
  Set <AN8_1> Value from "1012"
  Declare <CV_AN8_1>   Default Value ""
  Application {P01012-Address Book} - Repeat Count from "1"
    Type Fast Path "3/G01"
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
      IF <ADDRESS EXISTS>   ==   <YES>
        THEN
          QBE Information - Repeat Count from "1"
            Type to Grid [AddressNumber] Value from <AN8_1>
          Press Button {Find}
          Set <CV_AN8_1> Value from Grid [AddressNumber] Line 2
          IF <AN8_1>   <>   <CV_AN8_1>
            THEN
              Even though script expected address number to exist
              No matching address number was found.
              Press Button {Add}
              Form {Address Book Revision - W01012A} - Repeat Count from "1"
            ELSE
              Double click grid on row number "1"
              Form {Address Book Revision - W01012A} - Repeat Count from "1"
```

You can also compare variables between scripts by declaring a variable as external in one script, including the script with a parent script, and linking the external variable to a variable in the master script. You then build your conditional statement on these two variables.

You can use OneWorld Scripting Tool to set conditional statements of data equality or inequality, but the tool does not allow you to develop compound conditionals that link together.

Variable Addition

Variable addition enables you to scroll through a grid from top to bottom. You could, for example, write a command that adds one to the row number of the grid each time OneWorld Scripting Tool plays back the node of the script. If you set the repeat count of the node to match the number of lines in the grid, OneWorld Scripting Tool scrolls through the entire grid, one line at a time, from top to bottom.

Variable Subtraction

Variable subtraction enables you to scroll through the entire grid, one line at a time, from bottom to top. You write a command that subtracts one from the row number of the grid each time OneWorld Scripting Tool plays back the node of the script.

Variable Concatenation

Variable concatentation enables you to string together existing variable values to create a new variable. For example, you might have created two variables with values of 10 and 25. You could concatenate them to create a new variable whose value you use to select a range of values for a UBE.

System Variables

System variables draw their values from OneWorld, rather than from your input in OneWorld Scripting Tool. You do not need to declare a variable or set its value to use a system variable, since its value is determined during script playback.

The following table names the system variables and presents examples of how they might be used in script writing:

Name of system variable	Meaning	Possible use in script
Errors	OneWorld Scripting Tool records the number of OneWorld error messages that occur during script playback.	In conjunction with a conditional statement. For example, set condition that if the number of error messages returned is greater than 0, then OneWorld Scripting Tool should run the Exit OneWorld command.
Warnings	OneWorld Scripting Tool records the number of OneWorld warning messages that occur during script playback.	In conjunction with a conditional statement. For example, set condition that if OneWorld sends a warning message when the OK button is clicked, then OneWorld Scripting Tool should click the OK button twice.
Grid Row Count	OneWorld Scripting Tool records the number of filled rows in a grid.	To set a node's repeat count to ensure that OneWorld Scripting Tool touches each line in the grid during script playback.

Note: The value of the Grid Row Count system variable is determined by the number of rows that OneWorld fills after you click Find, not by the total number of rows in a filled grid. If you want the Grid Row Count value to reflect the actual number of rows in the grid, use the Select Grid Row command to go to the bottom of the grid. Once you have scripted this command, the Grid Row Count value is the total number of filled rows in the grid.

See Also

- *Performing Grid Row Operations.*

Valid Values Count

If you choose valid values count as a source of input, you also choose from the value selection list a valid value that you or someone else has created. The list can be stored either on your local drive or on a server. OneWorld Scripting Tool counts the number of items in the list and stores that value as a variable. You might use the valid values count as a source of input in conjunction with establishing the repeat count for a node by setting the value of the grid row count from the valid values count. The number of times OneWorld Scripting Tool plays back the script node matches the number of items in the valid values list, and you can write a command for OneWorld Scripting Tool to enter the values from the valid values list to a cell in each line of the grid.

See Also

- *Valid Values List*

Variable Watch List

The variable watch list, which you can select from the OneWorld Scripting Tool View menu, is a separate form that displays the values of variables in the script during playback. Because the watch list rests outside the OneWorld Scripting Tool form, you can display it at all times, even as you exit and open scripts.

The watch list contains two column headers: Variable and Value. During script playback, each time OneWorld Scripting Tool sets the value of a variable, it adds the variable name and its value to the list. If the variable value that OneWorld Scripting Tool enters to a header control or grid column is invalid and script playback stops, OneWorld Scripting Tool ceases adding values to the watch list. Each time that you stop the script and replay it, OneWorld Scripting Tool clears the watch list.

Validation Success

The Validation Success variable enables you to verify quickly if the data that you expected to exist as a result of running your script actually does. After you declare a validation, you choose Validation Success as a source of input, and choose from the value selection list the name of the validation whose success or failure you want to verify.

After you associate and run the validation and run the script, OneWorld Scripting Tool displays in the watch list the name and value of the validation success variable that you created. A value of 1 indicates that the validation was successful; a value of 0 indicates that the validation failed. In the rare instance that you declare and associate a validation, but do not run the validation, OneWorld Scripting Tool returns a value of 2 at the completion of playback.

See Also

- *The Database Validation Command*

UDC Visual Assist Value

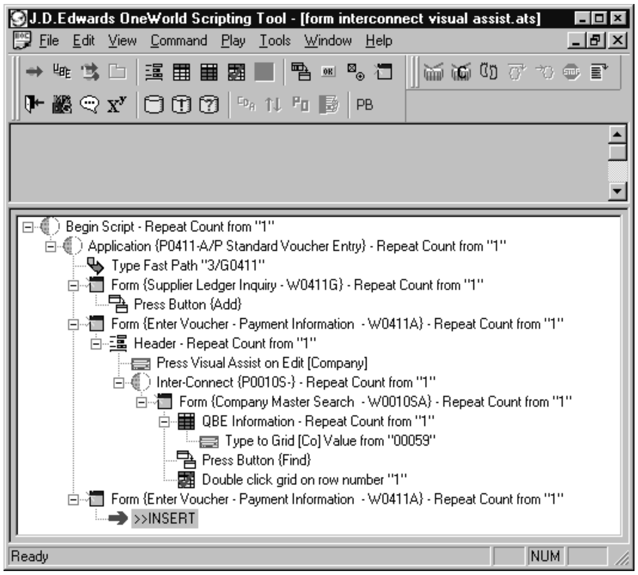
You can also use as a source of input a value you get from a UDC visual assist. Choosing UDC Visual Assist Value from the Source of Input list populates the UDC Visual Assist Value list with the same UDC values that you find by clicking the flashlight button in the OneWorld form. When you insert the value to your script in the playback mode, OneWorld Scripting Tool tests the code that opens the form and chooses the value that you specified for your script.

UDC Visual Assist Value appears in the Source of Input list only if a header control or grid column in the OneWorld form contains a visual assist.

Form Interconnect Visual Assist

You might need to use a visual assist other than a UDC visual assist. For example, you might want to run a Company Master Search. In OneWorld, when you click the flashlight icon for Company Master Search, you exit the current application to a new application.

You script this form interconnect in your OneWorld Scripting Tool script by clicking Form Interconnect Visual Assist in the Source of Input list. Note that when you choose Form Interconnect Visual Assist and insert the command to the script, OneWorld Scripting Tool tests the code that triggers a series of events in OneWorld. OneWorld Scripting Tool clicks the flashlight button, runs an Application Interconnect command, and confirms the new form. OneWorld Scripting Tool inserts both the Application Interconnect and Form commands to the script pane, and you can script any additional commands you need.

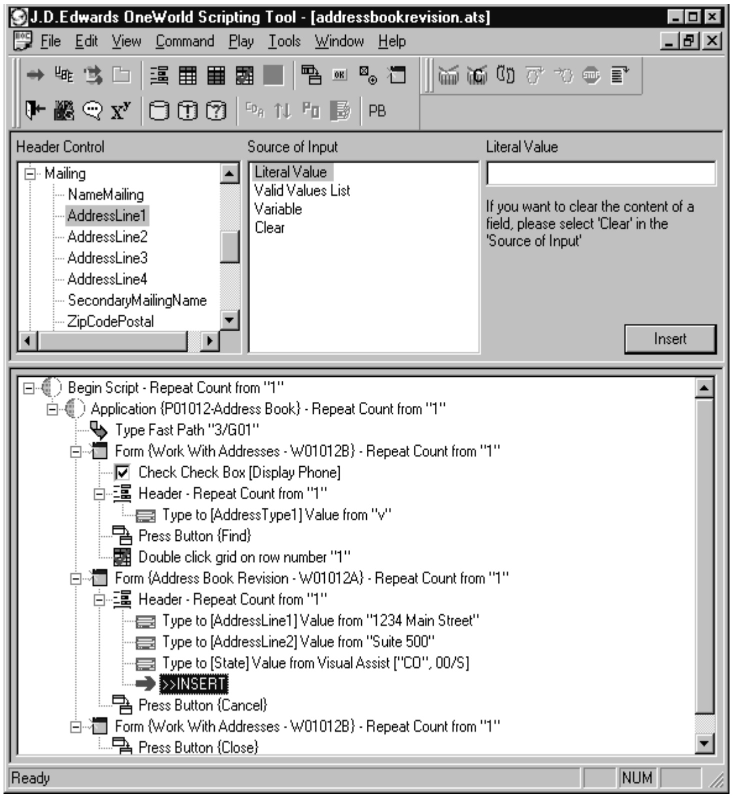

J.D.Edwards OneWorld Scripting Tool - [form interconnect visual assist.ats]
File Edit View Command Play Tools Window Help
PB

```
Begin Script - Repeat Count from "1"
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Press Visual Assist on Edit [Company]
        Inter-Connect {P0010S-} - Repeat Count from "1"
          Form {Company Master Search  - W0010SA} - Repeat Count from "1"
            QBE Information - Repeat Count from "1"
              Type to Grid [Co] Value from "00059"
            Press Button {Find}
            Double click grid on row number "1"
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      >>INSERT
```

The Form Interconnect Visual Assist choice appears in the Source of Input list only when a header control or grid column contains a visual assist that requires an exit to a new application. Also note that you do not use the value selection list when you choose Form Interconnect Visual Assist. This source of input requires that you choose a value after you have exited to the new application.

Clear Source of Input

The Clear command in the Source of Input pane allows you to remove an entry to a header control, grid column, or processing option form control. Each time you choose a control or grid column from the command pane and choose Literal Value from the Source of Input list, OneWorld Scripting Tool provides a reminder in the value selection list that you can clear the content of the control or column.



J.D.Edwards OneWorld Scripting Tool - [addressbookrevision.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

Header Control

- Mailing
  - NameMailing
  - AddressLine1
  - AddressLine2
  - AddressLine3
  - AddressLine4
  - SecondaryMailingName
  - ZipCodePostal

Source of Input

- Literal Value
- Valid Values List
- Variable
- Clear

Literal Value

If you want to clear the content of a field, please select 'Clear' in the 'Source of Input'

```
Begin Script - Repeat Count from "1"
  Application {P01012-Address Book} - Repeat Count from "1"
    Type Fast Path "3/G01"
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
      Check Check Box [Display Phone]
      Header - Repeat Count from "1"
        Type to [AddressType1] Value from "v"
      Press Button {Find}
      Double click grid on row number "1"
    Form {Address Book Revision - W01012A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [AddressLine1] Value from "1234 Main Street"
        Type to [AddressLine2] Value from "Suite 500"
        Type to [State] Value from Visual Assist ["CO", 00/S]
        >>INSERT
      Press Button {Cancel}
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
      Press Button {Close}
```

The Value Selection List

After choosing a Source of Input, you must specify the value or values of the input to the header control, grid column, or QBE line. You can use any of the following methods to supply the value:

- Entering a literal value of numbers, letters, spaces, or a combination thereof

To access English documentation updates, see https://knowledge.jdedwards.com/JDEContent/documentationcbt/overview/about_documentation_updates.pdf

- Choosing a valid values list that you have created
- Choosing a variable you have declared and set
- Choosing a system variable
- Choosing a UDC or form interconnect visual assist value

Remember that the caption of the value selection list changes to reflect the choice that you make in the Source of Input list.

Scripting the Type to Command

You work with the Header Control/Grid Column, Source of Input, and value selection lists to create a Type to command. Remember that you create this action command in conjunction with creating the context commands Header, Grid (or Detail Information), and QBE. When you click the Insert button, OneWorld Scripting Tool writes the context command and the Type to command and indents the Type to command beneath the context command to reflect the script pane command hierarchy.

This chapter covers the following tasks included in scripting the Type to command:

- ❑ Using the Header Control/Grid Column list
- ❑ Using the Source of Input list
- ❑ Using the value selection list
- ❑ Running the Type to command

Using the Header Control/Grid Column List

You begin writing the Type to command when you click Set Header Control Value, Set Grid Cell Value, or Set QBE Cell Value in the command menu. When you click one of these commands, the Header Control/Grid List appears, populated by the controls or columns that are contained within the active OneWorld form. You click a control or column to which you want to type data.

▶ To use a header control or grid column list

1. In the command menu of the OneWorld Scripting Tool form, click Set Header Control Value, Set Grid Cell Value, or Set QBE Cell Value.
2. In the Header Control/Grid Column list of the command pane, choose a header control or grid column to which you want to enter data.

Using the Source of Input List

After you have chosen a header control or grid column, you must choose a source of input to it. You click any one of four possible sources of input: Literal

Value, Valid Values List, Variable, or UDC Visual Assist Value, depending on the process you are testing and how you want your script to run.

This topic covers choosing the following topics:

❑ Using a literal value as a source of input

❑ Using a valid values list as a source of input

❑ Using a variable as a source of input

❑ Using a UDC visual assist value as a source of input

❑ Using a form interconnect visual assist value as a source of input

❑ Clearing an input to a header control or grid column

For the purposes of discussing using the Source of Input list, we will include the step that follows choosing a source of input, which is entering or choosing a value from the value selection list. We will, however, also discuss the value selection list separately in this chapter.

Using a Literal Value as a Source of Input

When you run a script, a literal value appears in the OneWorld form exactly as you type it in the unpopulated list captioned Literal Value. Some controls in OneWorld appear with visual assists, such as Master Business Search, that can be used to secure the literal value you enter in the value selection list in OneWorld Scripting Tool.

▶ To use a literal value as a source of input

1. From the command pane in the OneWorld Scripting Tool form, choose Literal from the Source of Input list.
2. In the Literal Value list, type an input as you would normally enter it in the header control, grid column, or QBE line of the OneWorld form. The entry can be letters, numerals, special characters, spaces, or a combination thereof.
3. Click the Insert button.

Using a Valid Values List as a Source of Input

After you have created a valid values list, you can use it as a source of input to a header control, grid column, or QBE line. You use a valid values list when you want to enter values multiple times to the chosen OneWorld environment, or when you want to run a script multiple times and input a different value each time.

To access English documentation updates, see
https://knowledge.jdedwards.com/JDEContent/documentationcbt/overview/about_documentation_updates.pdf

For example, the list may contain five items. You can change the repeat count for the node in which the list is contained to 5. During one script playback, OneWorld Scripting Tool loops through the node five times and inserts a different item from the list each time. On the other hand, you can leave the repeat count for the node in which the list is contained at 1, but change the repeat count at Begin Script to 5. During each of the five playbacks of the entire script, OneWorld Scripting Tool inputs one different value from the list.

If you close the script, leave OneWorld Scripting Tool open, then run the script again, OneWorld Scripting Tool takes the next value in the list and inputs it to the appropriate context. Once you have created the valid values list, you have stored the values in it, and you can draw on it as a source of value in subsequent scripts.

You must create a valid values list before you choose it as a source of input to a OneWorld form. To do so, you decide on the kind of valid values list you want to create, put values in the list, and give it a name.

OneWorld Scripting Tool allows you to create two kinds of valid values lists.

This topic covers the following components of using a valid values list as a source of input:

- ❑ Creating a list of literal values
- ❑ Creating a valid values list from a simple database query
- ❑ Using a valid values list in the script
- ❑ Updating the repeat count in a node

Creating a List of Literal Values

A list of literal values is a list that contains values that you choose. Before you create the list, you should verify that the values you create are valid for the OneWorld application you want to test.

▶ To create a list of literal values

1. From the menu bar of OneWorld Scripting Tool, click Tools.
2. Choose Generate Valid Values List.
3. Choose List of Literal Values from the Select Data File Type window.
4. Click Next.
5. Type a file name in the File Properties list.

   If you type the name of an existing file, the values in it auto-fill the list.

6. Type one or more values in the Values list, separating them by pressing Enter. The values should be stacked vertically in the box.
7. Click Finish.

Creating a Valid Values List from a Simple Database Query

A simple database query produces a valid values list that contains values that OneWorld Scripting Tool retrieves from the database and includes in the list, based on the table and column that you choose. You can limit the number of records in the list, and you can establish the way that you want OneWorld Scripting Tool to sort the records, such as in ascending, descending, or random order.

▶ To create a valid values list from a simple database query

1. From the menu bar in OneWorld Scripting Tool, click Tools.
2. Choose Generate Valid Values List.
3. Choose Simple Database Query from the Select Data File Type window.
4. Click Next.
5. In the Select Table window, double-click a table.
6. In the Select Column window, choose a table column.
7. Click Next.
8. Specify format and sort options by clicking the appropriate boxes.

   Note: To view the contents of the valid values list, click the Preview button. Advanced users can enter SQL statements at this point.

9. Click Next.
10. In the Finish window, assign a file name to the valid values list by typing to the control.
11. Click Finish.

Using a Valid Values List in the Script

Once you have created a valid values list, you return to the command menu and click the context in which you want to write commands. You choose a header control or grid column, then choose Valid Values List from the Source of Input list. When you choose Valid Values List, the name of the list you created appears in the value selection list.

▶ To use a valid values list as a source of input

1. From the header control or grid column list in the command pane of the OneWorld Scripting Tool form, choose a header control or grid column.
2. Choose Valid Values List from the Source of Input list.
3. Choose the name of a valid values list you have created.
4. Click the Insert button.

Updating the Repeat Count in a Node

The valid values list can have multiple items in it. If you want to use all the values as inputs to a header control, grid column, or QBE line, you change the repeat count in the Form command node of the script to match the number of items in the list. This ensures that OneWorld Scripting Tool successively types each value to the control, grid column or QBE line during playback until it exhausts the list.

▶ To update the repeat count in a node

1. In the script pane of the OneWorld Scripting Tool form, click the Form line for the node in which you scripted the valid values list as the value.
2. In the command pane, type the Repeat Count list the number of times that you want the node to play back.
3. Click the Update button.

Using a Variable as a Source of Input

To use a variable as a source of input you must first declare it, or give it a name. The variable can be declared at any point in the script. However, if you make the variable global, the value that you give it later can be used at any point in the script. After you have declared the variable, you can set it, or give it a value, which you store for later use in the script in the variable you declared. After you have set the value of the variable, you can change it at any point in the script. If you declare and set the value of more than one variable, you can write conditional statements to compare their values. This can be useful if, for example, you want to verify that a value exists in the database. If the conditional statement shows that the value does not exist, you can modify the script with commands to add the value.

This topic covers the following tasks:

❑ Declaring a variable

❑ Changing the scope of a variable

❑ Setting the value of a variable

❑ Using a variable in the script

❑ Updating the value of an existing variable

❑ Setting conditional statements

❑ Adding a value to a variable

❑ Subtracting a value from a variable

❑ Concatenating a variable

❑ Creating a variable to confirm validation success

❑ Creating a variable to store a valid values list count

Declaring a Variable

Declaring a variable means only that you give it a name. Giving it a name establishes the place where a value you set can be stored. You can insert the Declare variable command at any point in the script. Where you declare it determines its scope. However, you can change the scope of a variable by dragging it from one point in the script to another.

▶ To declare a variable

1. In the command menu of the OneWorld Scripting Tool form, click Variables.

   Note: You can perform this step at any point in the script.

2. In the command pane, type a name for the variable in the New Variable list.
3. In the Source of Value list, click None.

   Note: Choosing None means that you have simply given the variable a name at this point. You have not yet assigned a value to it.

4. Click the Insert button.

Changing the Scope of a Variable

The scope of a variable is the context within which you can use a value that you assign to it. The variable's scope can extend locally, to a OneWorld form or a single OneWorld application, or globally, throughout the entire script, regardless of how many applications you launch. If you make the variable

To access English documentation updates, see
https://knowledge.jdedwards.com/JDEContent/documentationcbt/overview/about_documentation_updates.pdf

global, you can choose any point in the script to set its value, and you can use the value at any point in the script.

You can declare the variable at any point in the script, then change its scope by moving it up or down the hierarchy of script pane commands. You do so by clicking the Declare command and dragging it to the point you choose.

As you drag the mouse, watch the arrow that appears over the Declare command line. The arrow pointing up indicates that the command line you are dragging will be placed above the line that is highlighted when you release the mouse button. The arrow pointing down indicates that the command line you are dragging will be placed below the line that is highlighted when you release the mouse button.

▶ Changing the scope of a variable

1. In the script pane of the OneWorld Scripting Tool form, Click the mouse on the Declare command line.

   Note: The Declare variable command is attached to the node established by the context command Form. A value assigned to this variable can be used only within this node. The variable's scope is local to the form.

2. Holding down the mouse button, drag the Declare command line to another context. To make the variable global, drag the Declare command to the top of the script.
3. When the Declare command line is on top of the Application command line and the arrow is pointing up, release the mouse button.

Setting the Value of a Variable

After you have declared the variable, you set its value. You store the value in the declared variable so that you can use it at points in the script that you determine by establishing the scope of the variable.

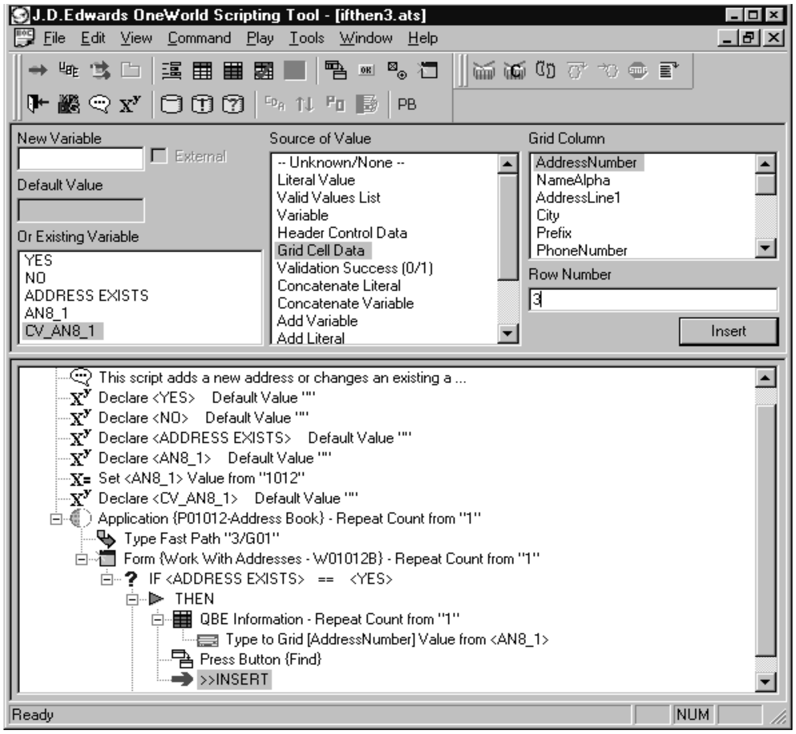
▶ To set the value of a variable

1. In the command menu of the OneWorld Scripting Tool form, click Variables.
2. In the Existing Variable list of the command pane, choose the name of the declared variable to which you want to assign a value.
3. In the Source of Value list, choose one of the five sources for the value:
   - Literal Value
   - Valid Values List
   - Variable

- Header Control Data
- Grid Cell Data

4. If you assign a literal value, type that number or word into the Literal Value list in the command pane. If you have created a list of valid values or declared and set the value of another variable, click the name of one in the list. If you want to derive the value from a header control or grid column, click the name of one that populates the list.

   Note: If you choose Grid Cell Data as the source of value, OneWorld Scripting Tool displays the grid columns for the form in which you are working and an unpopulated Row Number list. If you enter a grid row number to this list and click the Insert button, OneWorld Scripting Tool stores the value from the row that you specified.



New Variable

□ External

Default Value

Or Existing Variable: YES, NO, ADDRESS EXISTS, AN8_1, CV_AN8_1 (selected)

Source of Value: -- Unknown/None --, Literal Value, Valid Values List, Variable, Header Control Data, Grid Cell Data (selected), Validation Success (0/1), Concatenate Literal, Concatenate Variable, Add Variable, Add Literal

Grid Column: AddressNumber (selected), NameAlpha, AddressLine1, City, Prefix, PhoneNumber

Row Number: 3

```
This script adds a new address or changes an existing a ...
Declare <YES>    Default Value ""
Declare <NO>    Default Value ""
Declare <ADDRESS EXISTS>    Default Value ""
Declare <AN8_1>    Default Value ""
Set <AN8_1> Value from "1012"
Declare <CV_AN8_1>    Default Value ""
Application {P01012-Address Book} - Repeat Count from "1"
    Type Fast Path "3/G01"
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
        IF <ADDRESS EXISTS>   ==   <YES>
            THEN
                QBE Information - Repeat Count from "1"
                    Type to Grid [AddressNumber] Value from <AN8_1>
                Press Button {Find}
                >>INSERT
```

5. Click the Insert button.

   OneWorld Scripting Tool sets the value according to the choice you make and stores it in your declared variable.

Using a Variable in the Script

After you have declared a variable and set its value, you can use the value as a source of input. Remember that the scope that you established for the variable determines where you can use its value in the script.

▶ To use a variable in the script

1. In the command menu of the OneWorld Scripting Tool form, click Variables.
2. In the header control or grid column list of the command pane, click a header control or grid column of a OneWorld form to establish the context to which you want to input the value of the variable.
3. In the Source of Input list, choose Variable.
4. In the Variable list, choose the name of the variable you declared at the outset of the scripting process and whose value you set.
5. Click the Insert button.

   OneWorld Scripting Tool inputs to the header control, grid column, or QBE line the variable value that you set.

Updating the Value of an Existing Variable

You might decide that you want to change the value of an existing variable. You can do so at any point in the script that you wish if you have declared the variable and set its value.

▶ To update the value of an existing variable

1. In the script pane of the OneWorld Scripting Tool form, click the Set command line.
   - This is the line where you assigned a value to the declared variable.
2. In the command menu, click Variables.
3. In the Variable list of the command pane, choose the name of the variable that you want to update.
4. Make a choice from the Source of Value list.
5. Choose or enter a value.
6. Click the Update button.

Setting Conditional Statements

You can also use OneWorld Scripting Tool to compare the values of two variables. You do so by clicking If <var> in the command menu. When you do so, the command pane displays three populated lists that permit you to write a conditional If/Then statement, which also includes an Else statement. OneWorld Scripting Tool populates two of the lists with the names of variables that you have declared. You write the left and right side of your conditional statement by making a choice from each of these lists. You can also compare the left variable to a literal on the right, rather than a variable, by entering a literal value in the Right Literal list.

The third list contains conditional operators such as equal to, not equal to, greater than, and so on. In addition, the command pane contains a check box option, Numeric Comparison. When you check this option, OneWorld Scripting Tool converts the variables that you have declared in words to numeric values before it compares them. You can also write a conditional statement that uses a string, rather than a numeric. To do so, you choose the option Is Not In from the list of conditional operators.

Note that you are not required to write any commands as part of the Else branch of the script. You write commands that are part of the Else branch when you want OneWorld Scripting Tool to run a series of commands only if the first part of your conditional statement is not true.

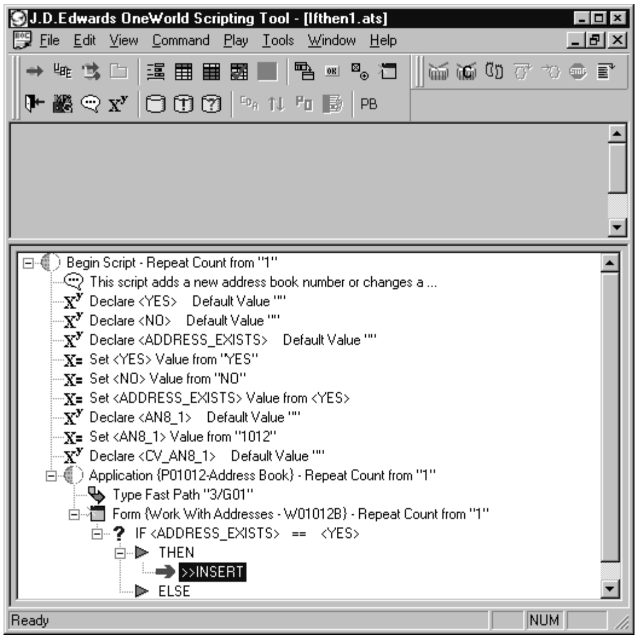
▶ To set a conditional statement

1. In the OneWorld Scripting Tool form, declare and set the value of two variables.

   Note: You can declare the variables and set their values at any point in the script prior to writing the conditional statement.

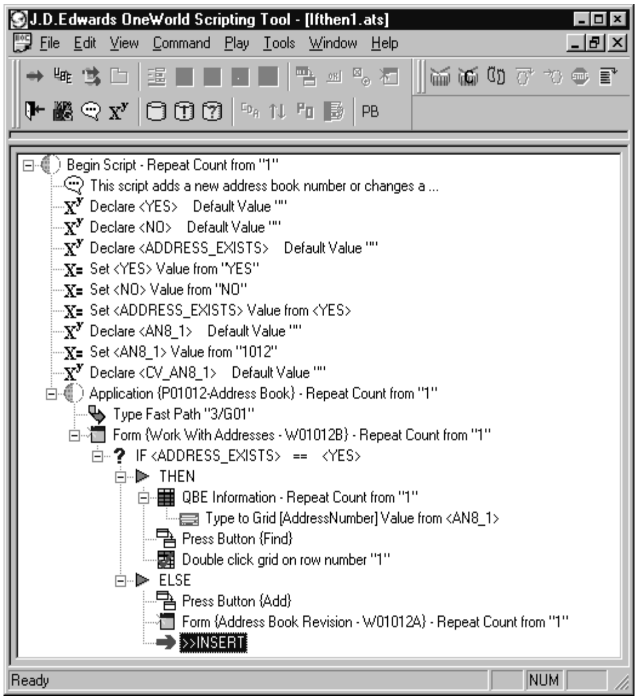
2. In the command menu, click If <var> = = <var>.
3. In the command pane, make choices from the available lists:
   - Left Variable
   - Operator
   - Right Variable
   - Right Literal
4. If the variables’ values are numeric, click the Numeric option. This is optional.
5. Click the Insert button.

   OneWorld Scripting Tool enters the If portion of the conditional statement to the script pane. The Then and Else portions are blank.


J.D.Edwards OneWorld Scripting Tool - [Ifthen1.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
    This script adds a new address book number or changes a ...
    Declare <YES>    Default Value ""
    Declare <NO>    Default Value ""
    Declare <ADDRESS_EXISTS>    Default Value ""
    Set <YES> Value from "YES"
    Set <NO> Value from "NO"
    Set <ADDRESS_EXISTS> Value from <YES>
    Declare <AN8_1>    Default Value ""
    Set <AN8_1> Value from "1012"
    Declare <CV_AN8_1>    Default Value ""
    Application {P01012-Address Book} - Repeat Count from "1"
        Type Fast Path "3/G01"
        Form {Work With Addresses - W01012B} - Repeat Count from "1"
            IF <ADDRESS_EXISTS>   ==   <YES>
                THEN
                    >>INSERT
                ELSE
```

6. Write and insert to the script commands that constitute the Then branch of the conditional statement.
7. Click the Insert button.
8. If you want to add an Else condition, drag the insertion cursor beneath the Else command line in the script pane.
9. Write and insert to the script commands that constitute the Else branch of the conditional statement.


J.D.Edwards OneWorld Scripting Tool - [Ifthen1.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
    This script adds a new address book number or changes a ...
    Declare <YES>    Default Value ""
    Declare <NO>    Default Value ""
    Declare <ADDRESS_EXISTS>    Default Value ""
    Set <YES> Value from "YES"
    Set <NO> Value from "NO"
    Set <ADDRESS_EXISTS> Value from <YES>
    Declare <AN8_1>    Default Value ""
    Set <AN8_1> Value from "1012"
    Declare <CV_AN8_1>    Default Value ""
    Application {P01012-Address Book} - Repeat Count from "1"
        Type Fast Path "3/G01"
        Form {Work With Addresses - W01012B} - Repeat Count from "1"
            IF <ADDRESS_EXISTS>   ==   <YES>
                THEN
                    QBE Information - Repeat Count from "1"
                        Type to Grid [AddressNumber] Value from <AN8_1>
                    Press Button {Find}
                    Double click grid on row number "1"
                ELSE
                    Press Button {Add}
                    Form {Address Book Revision - W01012A} - Repeat Count from "1"
                    >>INSERT
```

10. Click the Insert button.

You can include branches of script whose execution you make dependent on the conditional statement.

Adding a Value to a Variable

You can add to a variable the value of another variable or a literal value, and you can do so whether you have merely declared a variable or if you have both declared a variable and set its value. Remember, however, that if you declare a variable with the intention of setting its value by adding a value later in the script, you should enter a default value so that OneWorld Scripting Tool knows the value to add to. Remember also that if you want to add the value of one variable to another variable, you must have first declared and set the value of the variable whose value you want to add.

▶ To add a value to a variable

1. In the Command menu of the OneWorld Scripting Tool form, choose Variables.
2. Choose a variable from the Existing Variable list.
3. Make a choice from the Source of Value list:
   - Add Variable
   - Add Literal
4. If you want to add a variable value, choose the name of a variable from the value selection list. If you want to add a literal value, enter a literal value in the value selection list.
5. Click the Insert button.

Subtracting a Value from a Variable

You can subtract from a variable either the value of another variable or a literal value. You can accomplish value subtraction whether or not you have set the variable's value, provided that when you declare the variable you set a default value.

▶ To subtract a value from a variable

1. In the Command menu of the OneWorld Scripting Tool form, choose Variables.
2. Choose a variable from the Existing Variable list.
3. Make a choice from the Source of Value list:
   - Subtract Variable

- Subtract Literal

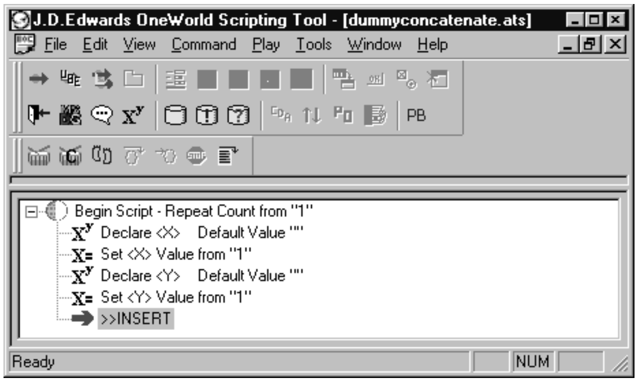
4. If you want to subtract a variable value, choose the name of a variable from the value selection list; if you want to subtract a literal value, enter a literal value in the value selection list.
5. Click the Insert button.

Concatenating a Variable

Concatenating a variable enables you to create alphanumeric strings from two or more variables. You can construct a concatenated variable from other variables or from literal values. The table below demonstrates in simple terms the principle behind variable concatenation:

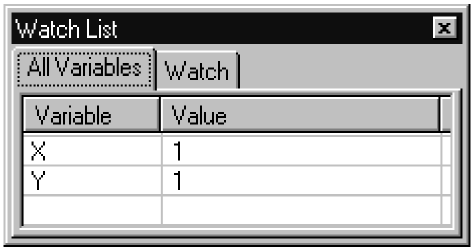
Variable Name	**Variable Value**
X	1
Y	1
X concatenated from Y	11

In this example, you declare and set the values of two variables, X and Y.


J.D.Edwards OneWorld Scripting Tool - [dummyconcatenate.ats]
File
Edit
View
Command
Play
Tools
Window
Help
PB
Ready
NUM

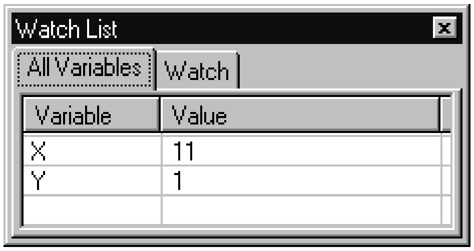
```
Begin Script - Repeat Count from "1"
    Declare <X>    Default Value ""
    Set <X> Value from "1"
    Declare <Y>    Default Value ""
    Set <Y> Value from "1"
    >>INSERT
```

The variable watch list displays the names and values of the two variables.



Variable	Value
X	1
Y	1

If you concatenate the X and Y values to create a new value for X, the watch list displays the new value for X:



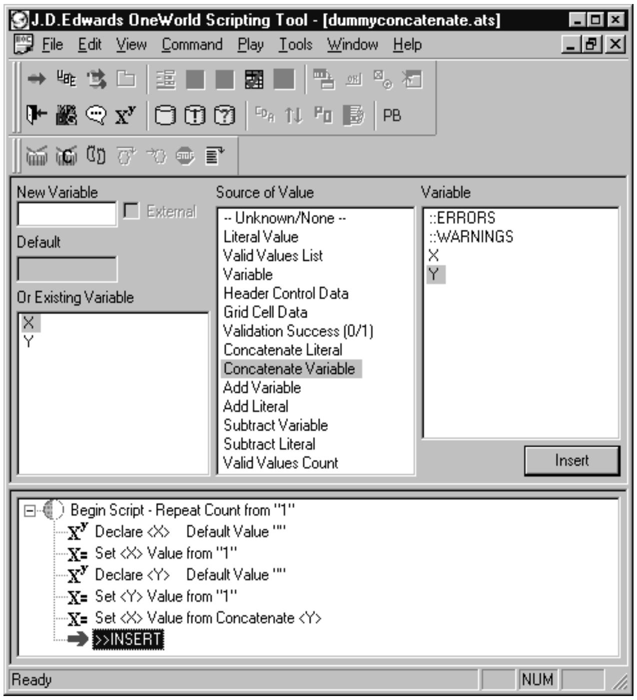
Variable	Value
X	11
Y	1

Before You Begin

❑ Before you can concatenate a variable, you must declare at least one variable and either set its value or enter a default value.

▶ To concatenate a variable

1. In the command menu of the OneWorld Scripting Tool form, choose Variables.
2. Choose a variable from the Existing Variable list.
3. Make a choice from the Source of Value list:
   - Concatenate Literal
   - Concatenate Variable
4. If you want to concatenate a literal value, enter a literal value to the value selection list. If you want to concatenate a variable, choose a variable from the value selection list.
5. Click the Insert button.



```
Begin Script - Repeat Count from "1"
    Declare <X>    Default Value ""
    Set <X> Value from "1"
    Declare <Y>    Default Value ""
    Set <Y> Value from "1"
    Set <X> Value from Concatenate <Y>
    >>INSERT
```

OneWorld Scripting Tool creates a concatenated value for the variable you chose from the Existing Variable list.

Creating a Variable to Confirm Validation Success

To confirm the success or failure of a validation, you declare a variable and choose Validation Success as a source of value. When you choose Validation Success, OneWorld Scripting Tool displays in the value selection list the names of the validations that you have declared. When you run the validation, OneWorld Scripting Tool sets the value of 1 for the variable if the validation succeeds. If the validation fails, OneWorld Scripting Tool sets the value of the variable as 0. If you declare the validation and assign values, but do not run it, OneWorld Scripting Tool sets the value of the validation variable as 2.

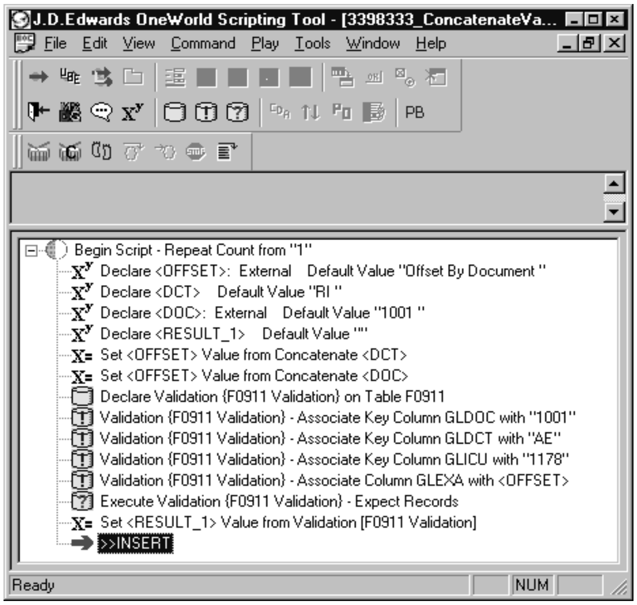
With Watch List selected from the OneWorld Scripting Tool View menu, you can easily determine following playback whether your data validation was successful. The watch list displays the name of the validation variable in the Variable column. If the validation was successful, OneWorld Scripting Tool displays a 1 in the Value column.

Before You Begin

- ❑ To use the Validation Success variable as a source of input, you must first declare and associate a validation. For OneWorld Scripting Tool to return a value of 1 or 0, indicating success or failure, you must also run the validation. See *Scripting the Database Validation Command*.

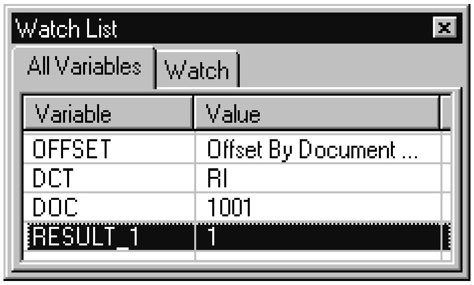
▶ To create a variable to confirm validation success

1. In the command menu of the OneWorld Scripting Tool form, choose Variables.
2. Enter the name of a variable to the New Variable list or choose the name of a variable from the Existing Variable list.
3. Choose Validation Success (0/1) from the Source of Value list.
4. Choose the name of a validation from the value selection list.
5. Click the Insert button.


J.D.Edwards OneWorld Scripting Tool - [3398333_ConcatenateVa...
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
    Declare <OFFSET>:  External    Default Value "Offset By Document "
    Declare <DCT>    Default Value "RI "
    Declare <DOC>:  External    Default Value "1001 "
    Declare <RESULT_1>    Default Value ""
    Set <OFFSET> Value from Concatenate <DCT>
    Set <OFFSET> Value from Concatenate <DOC>
    Declare Validation {F0911 Validation} on Table F0911
    Validation {F0911 Validation} - Associate Key Column GLDOC with "1001"
    Validation {F0911 Validation} - Associate Key Column GLDCT with "AE"
    Validation {F0911 Validation} - Associate Key Column GLICU with "1178"
    Validation {F0911 Validation} - Associate Column GLEXA with <OFFSET>
    Execute Validation {F0911 Validation} - Expect Records
    Set <RESULT_1> Value from Validation [F0911 Validation]
    >>INSERT
```

If the validation runs successfully, OneWorld Scripting Tool displays in the watch list a value of 1 for the validation variable.



Variable	Value
OFFSET	Offset By Document ...
DCT	RI
DOC	1001
RESULT_1	1

Creating a Variable to Store a Valid Values List Count

To capture and store the value that equates to the number of items in a valid values list, you choose Valid Values Count as a source of value for a new or existing variable. After you choose this source of value, you choose a valid values list. OneWorld Scripting Tool stores as the variable's value the number of items in the list.

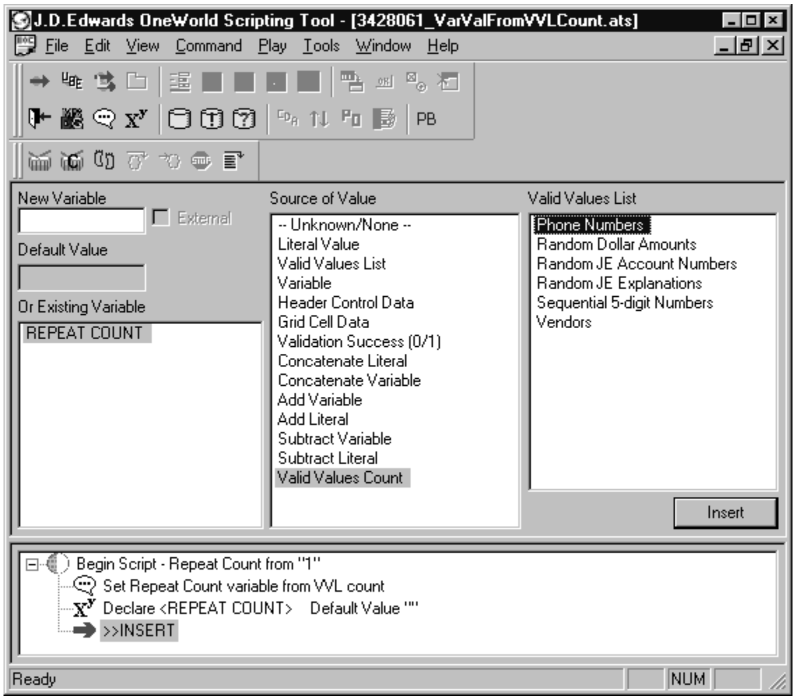
To use the Valid Values Count as a source of value, you must create a valid values list, either by generating a simple database query or by creating a list of literal values.

See Also

- *Using a Valid Values List as a Source of Input*

▶ To create a variable to store a valid valid values list count

1. In the command menu of the OneWorld Scripting Tool form, choose Variables.
2. Enter the name of a variable to the New Variable list or choose the name of a variable from the Existing Variable list.
3. Choose Valid Values Count from the Source of Value list.
4. Choose the name of a valid values list from the value selection list.


J.D.Edwards OneWorld Scripting Tool - [3428061_VarValFromVVLCount.ats]
File Edit View Command Play Tools Window Help
PB
External
Insert
Ready
NUM

New Variable

Default Value

Or Existing Variable: REPEAT COUNT

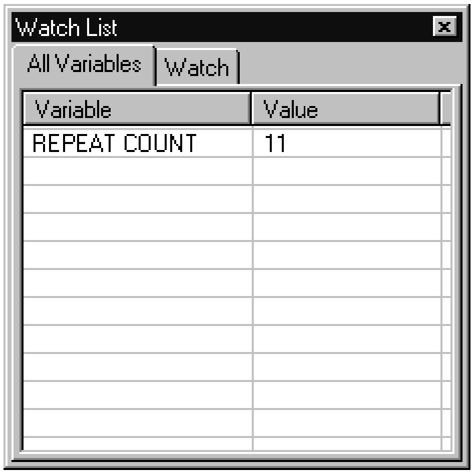
Source of Value:
-- Unknown/None --
Literal Value
Valid Values List
Variable
Header Control Data
Grid Cell Data
Validation Success (0/1)
Concatenate Literal
Concatenate Variable
Add Variable
Add Literal
Subtract Variable
Subtract Literal
Valid Values Count

Valid Values List:
Phone Numbers
Random Dollar Amounts
Random JE Account Numbers
Random JE Explanations
Sequential 5-digit Numbers
Vendors

```
Begin Script - Repeat Count from "1"
    Set Repeat Count variable from VVL count
    Declare <REPEAT COUNT>    Default Value ""
    >>INSERT
```

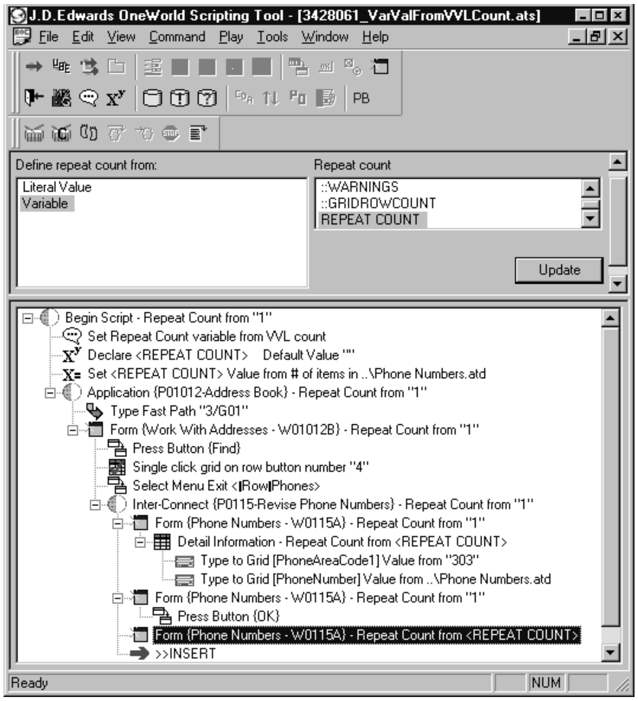
5. Click the Insert button.

   The watch list displays the number of items in the valid values list as the value of the variable.



Variable	Value
REPEAT COUNT	11

6. To use the valid values count value to set the repeat count for a node, select the node in the script pane that contains the valid values list you used to determine the value of the Valid Values Count variable.
7. In the Define repeat count list, choose Variable.
8. In the Repeat Count list, choose the variable that stores the valid values count value.
9. Click the Update button.


J.D.Edwards OneWorld Scripting Tool - [3428061_VarValFromVVLCount.ats]
File
Edit
View
Command
Play
Tools
Window
Help
PB
Update
Ready
NUM

Define repeat count from:
Literal Value
Variable

Repeat count
::WARNINGS
::GRIDROWCOUNT
REPEAT COUNT

```
Begin Script - Repeat Count from "1"
    Set Repeat Count variable from VVL count
    Declare <REPEAT COUNT>    Default Value ""
    Set <REPEAT COUNT> Value from # of items in ..\Phone Numbers.atd
    Application {P01012-Address Book} - Repeat Count from "1"
        Type Fast Path "3/G01"
        Form {Work With Addresses - W01012B} - Repeat Count from "1"
            Press Button {Find}
            Single click grid on row button number "4"
            Select Menu Exit <|Row|Phones>
            Inter-Connect {P0115-Revise Phone Numbers} - Repeat Count from "1"
                Form {Phone Numbers - W0115A} - Repeat Count from "1"
                    Detail Information - Repeat Count from <REPEAT COUNT>
                        Type to Grid [PhoneAreaCode1] Value from "303"
                        Type to Grid [PhoneNumber] Value from ..\Phone Numbers.atd
                Form {Phone Numbers - W0115A} - Repeat Count from "1"
                    Press Button {OK}
                Form {Phone Numbers - W0115A} - Repeat Count from <REPEAT COUNT>
                >>INSERT
```

OneWorld Scripting Tool updates the repeat count value, which now matches the item number value in the valid values list. When you play back the script, OneWorld Scripting Tool enters a value from the valid values list to a grid cell, one grid row at a time until it has used each value in the valid values list.

Using a UDC Visual Assist Value as a Source of Input

Some header controls, grid columns, and QBE lines in OneWorld forms contain UDC visual assists. When you choose UDC Visual Assist Value as a source of input, the UDCs in the value selection list in the command pane mirror the codes in the OneWorld visual assist forms.

The picture of a flashlight identifies the UDC visual assist, but OneWorld identifies other visual assists in this way. UDC Visual Assist Value appears in the Source of Input list only if a header control or grid column in the active OneWorld form has a UDC visual assist.

▶ To use a UDC Visual Assist value as a source of input

1. In the command pane of the OneWorld Scripting Tool form, choose a header control or grid column from the Header Control or Grid Column list.
2. Choose UDC Visual Assist Value from the Source of Input list.
3. In the UDC Visual Assist Value list, choose a UDC value.

   Note: These values mirror the UDC values you find if you click the flashlight in the corresponding control or column in a OneWorld form.

4. Click the Insert button.

   OneWorld Scripting Tool runs the code path in OneWorld and inserts the UDC value to the script.

Using a Form Interconnect Visual Assist as a Source of Input

Some header controls and grid columns contain visual assists that require you to exit from the current application to a new one. If you have chosen a header control or grid column that has this type of visual assist, such as Company Master Search, OneWorld Scripting Tool displays Form Interconnect Visual Assist in the Source of Input list. If you choose this source of input, OneWorld Scripting Tool clicks the flashlight button in the OneWorld form and writes an Application Interconnect command and Form command. You can then write any additional commands that you need.

Note that you do not use the value selection list when you make this choice. Once you have exited to a new application and form, you can write the additional commands that you need as part of your script.

▶ To use a form interconnect visual assist as a source of input

1. In the command pane of the OneWorld Scripting Tool form, choose a header control or grid column from the Header Control or Grid Column list.
2. Choose Form Interconnect Visual Assist from the Source of Input list.

   OneWorld Scripting Tool automatically writes a command to click the flashlight button in the active OneWorld form, then runs an Application Interconnect command and a Form command.

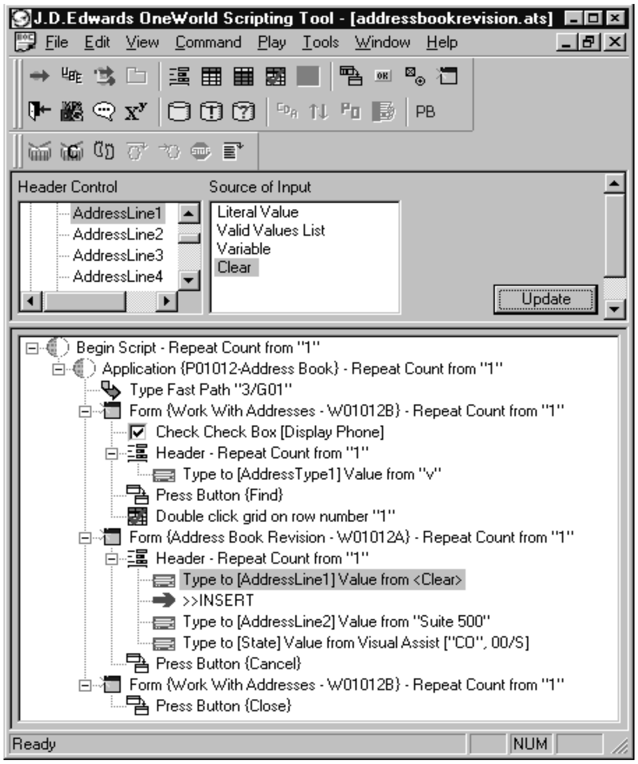
Clearing an Input to a Header Control or Grid Column

After you have entered a value to a header control or grid column, you might want to delete it, or you might want to clear an existing value, such as a date, from a control or column in a OneWorld form. OneWorld Scripting Tool allows you to do this by writing a Delete command. You choose Delete from the Source of Input list in the command pane.

▶ To clear an input to a header control or grid column

1. In the command menu of the OneWorld Scripting Tool form, click either Set Header Control Value, Set Grid Cell Value, or Set QBE Cell Value.
2. In the OneWorld Scripting Tool command pane, choose the name of a header control or grid column from the Header Control or Grid Column List.
3. In the Source of Input list, choose Delete.
4. Click the Insert button.

   OneWorld Scripting Tool clears the chosen header control or grid column of its value.


J.D.Edwards OneWorld Scripting Tool - [addressbookrevision.ats]
File Edit View Command Play Tools Window Help
PB
Header Control
Source of Input
Update
Ready
NUM

Header Control: AddressLine1, AddressLine2, AddressLine3, AddressLine4

Source of Input: Literal Value, Valid Values List, Variable, Clear

```
Begin Script - Repeat Count from "1"
  Application {P01012-Address Book} - Repeat Count from "1"
    Type Fast Path "3/G01"
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
      Check Check Box [Display Phone]
      Header - Repeat Count from "1"
        Type to [AddressType1] Value from "v"
      Press Button {Find}
      Double click grid on row number "1"
    Form {Address Book Revision - W01012A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [AddressLine1] Value from <Clear>
        >>INSERT
        Type to [AddressLine2] Value from "Suite 500"
        Type to [State] Value from Visual Assist ["CO", 00/S]
      Press Button {Cancel}
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
      Press Button {Close}
```

Using the Value Selection List

Once you have chosen a header control or grid column and a source of input to it, you complete the Type to command by choosing a value from the value selection list. OneWorld Scripting Tool displays this list when you choose a source of input and captions it according to the source of input that you choose.

If you want to input a literal value to the header control or grid column, you type that value into the unpopulated value selection list. If you want to input the values you have assigned to a valid values list or variable, you choose the name of the list or variable from the value selection list. If you want to input a UDC visual assist value, you choose one from the value selection list. Finally, you might want to clear an entry to a control in a header or a processing option form or to a grid column. OneWorld Scripting Tool allows you to do this by choosing Clear from the Source of Input list.

This topic discusses using the value selection list to perform the following tasks:

- Assigning a literal value
- Assigning a valid values list value
- Assigning a variable value
- Assigning a UDC visual assist value

- Assigning a form interconnect visual assist value

Assigning a Literal Value

The literal value you assign as an input to a header control or grid column can be numbers, letters, special characters, or a combination thereof. Be sure to verify that the literal value you assign is a valid input.

▶ To assign a literal value

1. In the command pane of the OneWorld Scripting Tool form, choose a header control or grid column name from the Header Control or Grid Column list.
2. Choose Literal Value from the Source of Input list.
3. In the value selection list, labeled Literal Value, input the value you want to assign to the control or grid column

Assigning a Valid Values List Value

You can assign a value from a valid values list as an input to a header control or grid column. OneWorld Scripting Tool chooses the first value in the list the first time you play back the script. If you choose to play back more than once, OneWorld Scripting Tool chooses the second value in the list on the second loop. This pattern continues until the loop ends or the item list is exhausted.

▶ To assign a valid values list value

1. In the command pane of the OneWorld Scripting Tool form, choose a header control or grid column name from the Header Control or Grid Column list.
2. Choose Valid Values List from the Source of Input list.
3. In the value selection list, labeled Valid Values List, choose the name of a valid values list you previously created.

Assigning a Variable Value

You can declare a variable, set its value, and assign that value to a header control or grid column. Once you have set the value, OneWorld Scripting Tool stores it, and it is available for you to use at any point in the script.

▶ To assign a variable value

1. In the command pane of the OneWorld Scripting Tool form, choose a header control or grid column name from the Header Control or Grid Column list.
2. Choose Variable from the Source of Input list.
3. In the value selection list, labeled Variable, choose the name of a variable that you previously declared and whose value you set.

Assigning a UDC Visual Assist Value

OneWorld Scripting Tool enables you to choose a UDC value to those header controls or grid columns in OneWorld forms that contain a UDC visual assist. The value selection form displays the valid UDC values for the control or column you chose. If the control or column that you chose does not contain a UDC value, UDC Visual Assist Value does not appear as a choice in the Source of Input list.

▶ To assign a UDC visual assist value

1. In the command pane of the OneWorld Scripting Tool form, choose a header control or grid column name from the Header Control or Grid Column list.
2. Choose UDC Visual Assist Value from the Source of Input list.
3. In the value selection list, labeled UDC Visual Assist Value, choose the name of a UDC value.

Assigning a Form Interconnect Visual Assist Value

OneWorld Scripting Tool enables you to choose a form interconnect visual assist value to those header controls or grid columns in OneWorld forms that contain a visual assist that requires an exit to a new application. You do not use the value selection list when you choose Form Interconnect Visual Assist as a source of input. Instead, after you exit to a new application, you script an input to a header control or grid column in the active form of that application by using the Press Toolbar Button command to find and select a value.

▶ To assign a form interconnect visual assist value

1. In the command pane of the OneWorld Scripting Tool form, choose a header control or grid column name from the Header Control or Grid Column list.
2. Choose Form Interconnect Visual Assist from the Source of Input list.
3. Click the Insert button.

OneWorld Scripting Tool automatically presses the visual assist button on the header control or grid column that you have chosen, then writes an Application Interconnect command and a Form command to the script pane.

4. In the OneWorld Scripting Tool command menu, choose QBE.
5. In the OneWorld Scripting Tool command pane, choose the name of a grid column from the Grid Column list.
6. Choose a source of input from the Source of Input list.
7. Enter or choose a value from the value selection list.
8. Click the Insert button.
9. In the command menu, choose Press Toolbar Button.
10. In the command pane, choose Standard Button.
11. Choose Find.
12. Click the Insert button.
13. In the command menu, choose Press Toolbar Button.
14. In the command pane, choose Standard Button.
15. Choose Select.

    OneWorld Scripting Tool enters the value from the form interconnect visual assist to the header control in the form you had originally chosen.

16. In the command menu, choose Press Toolbar Button.
17. In the command pane, choose Standard Button.
18. Choose Close or Cancel.
19. In the command menu, choose Form.
20. In the command pane, choose from the Form list the name of the form from which you exited when you scripted pressing the form interconnect visual assist button.

    Note: This command confirms for OneWorld Scripting Tool that you have returned to the previous form. If you do not confirm the form, you cannot continue scripting.

21. Click the Insert button.

Running the Type to Command

Once you have made choices from each of the three command pane lists, you run the Type to command simply by clicking the Insert button. OneWorld Scripting Tool uses the information you provided in your choices to write two command lines in the script pane. One command line contains the context, either header, grid, or QBE, and the repeat count for the node. The other contains the name of the header control or grid column you chose; a symbol that indicates whether you chose a literal value, a valid values list, a variable, or a UDC visual assist value as your source of input; and the value that you assigned.

This topic covers the following tasks involved in executing the Type to command:

- Typing data to a header control
- Clicking options in a header
- Typing data to a grid column
- Typing data to a QBE line

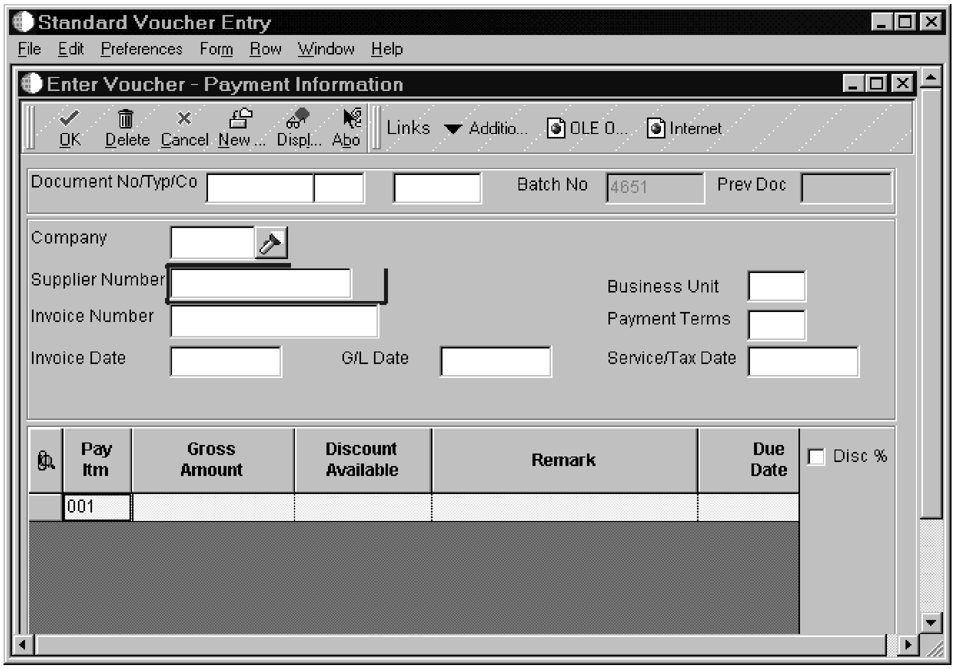
Typing Data to a Header Control

You use the command pane lists to script inputs to the header portion of a OneWorld form. Clicking Set Header Control Value in the command menu establishes the header as the context in which you type data. The selections that you make from the lists in the command pane create the Type to action command. When you click the Insert button, the command line specifying the header control as the context appears as a node. The command line identifies the control that you choose and the value you type to it. OneWorld Scripting Tool inserts subsequent Type to commands as leaves connected to the node.

▶ To type data to a header control

1. In the command menu of the OneWorld Scripting Tool form, click the Set Header Control Value in the cool bar.
2. click the name of a control from the Header Control list.

   When OneWorld is running, a BlueCue highlights the control in the OneWorld form that corresponds to the control chosen in OneWorld Scripting Tool.



Batch No 4651

Pay Itm 001

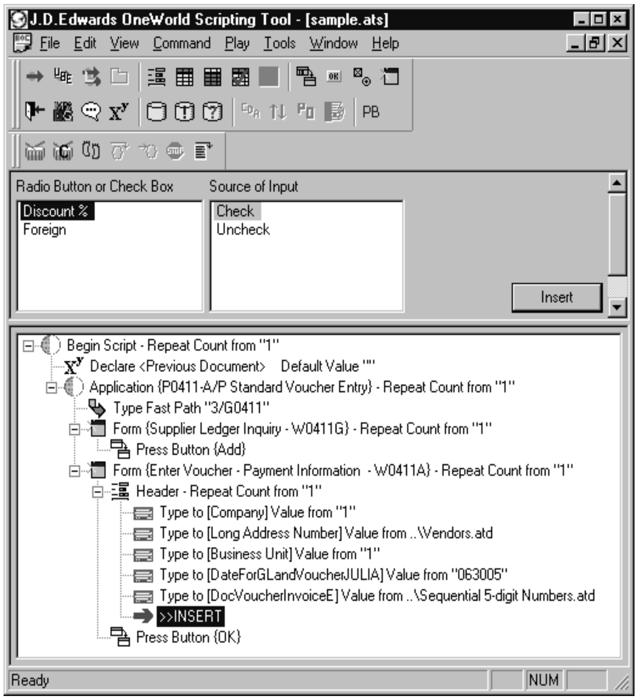
3. Choose a source of input from the Source of Input list.
4. In the value selection list, enter a literal value or choose the name of a valid values list, variable, or UDC visual assist value.
5. Click the Insert button.
6. Script inputs to additional header controls by clicking Set Header Control Value in the command menu and repeating steps 1-5.

Clicking Options in a Header

Some OneWorld forms also contain options that you can click. You click Check Box/Radio Button in the command menu if you want to write commands for these options in your script. The command to click an option is a different command than the Type to command, which is used to type data to header controls, grid columns, or QBE lines. However, when you work with a form that contains these options in its header, OneWorld Scripting Tool inserts the command in the Header node along with any Type to commands you write.

▶ To script clicking options in a header

1. In the command menu of the OneWorld Scripting Tool form, click Show Radio Button/Check Box.
2. In the command pane, choose an option from the Radio Button or Check Box list.
3. Click Check or Uncheck in the Source of Input list if the option is a check box.



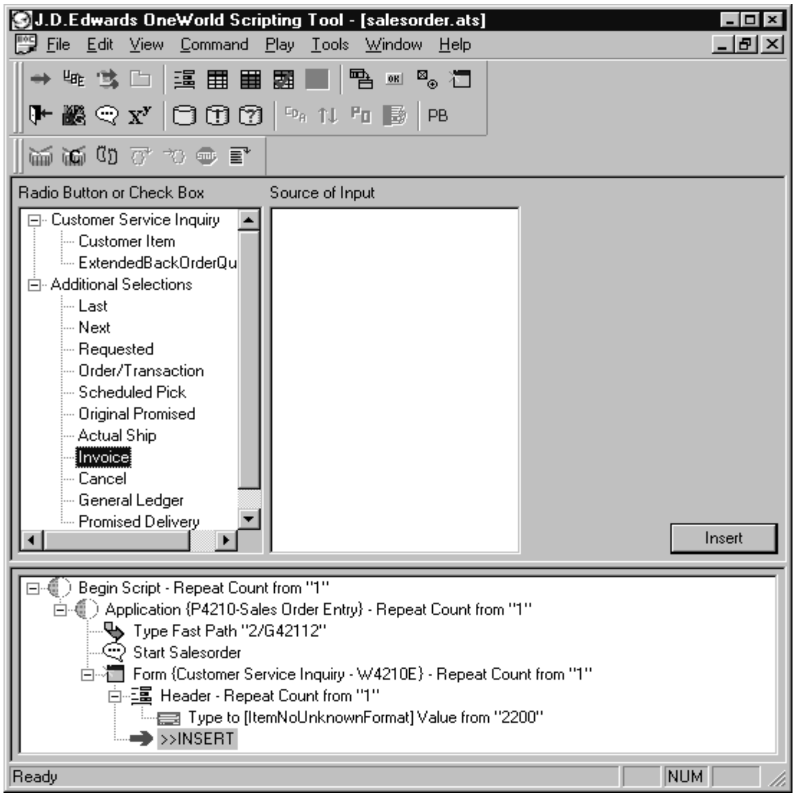
J.D.Edwards OneWorld Scripting Tool - [sample.ats]

Radio Button or Check Box: Discount %, Foreign

Source of Input: Check, Uncheck

```
Begin Script - Repeat Count from "1"
  Declare <Previous Document>    Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        Type to [Long Address Number] Value from ..\Vendors.atd
        Type to [Business Unit] Value from "1"
        Type to [DateForGLandVoucherJULIA] Value from "063005"
        Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
        >>INSERT
      Press Button {OK}
```

If the option is a radio button, the Source of Input list is unpopulated. OneWorld Scripting Tool clicks the radio button when you insert the command.



J.D.Edwards OneWorld Scripting Tool - [salesorder.ats]

Radio Button or Check Box:

- Customer Service Inquiry
  - Customer Item
  - ExtendedBackOrderQu
- Additional Selections
  - Last
  - Next
  - Requested
  - Order/Transaction
  - Scheduled Pick
  - Original Promised
  - Actual Ship
  - Invoice
  - Cancel
  - General Ledger
  - Promised Delivery

```
Begin Script - Repeat Count from "1"
  Application {P4210-Sales Order Entry} - Repeat Count from "1"
    Type Fast Path "2/G42112"
    Start Salesorder
    Form {Customer Service Inquiry - W4210E} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [ItemNoUnknownFormat] Value from "2200"
      >>INSERT
```

Typing Data to a Grid Column

You use the command pane lists to script inputs to the grid detail portion of a OneWorld form. Clicking Set Grid Cell Value in the command menu establishes the grid detail area as the context in which you type data. The selections that you make from the lists in the command pane create the Type to action command. When you click the Insert button, the command line specifying the grid detail area as the context appears as a node. The Type to command is indented beneath and attached to the node. The command line identifies the column that you chose and the value you typed to it. OneWorld Scripting Tool inserts subsequent Type to commands as leaves connected to the node. You can script different inputs to multiple rows of the grid, or you can script the input to one row multiple times, using a playback loop in OneWorld Scripting Tool.

To script commands to type data to a grid column of a OneWorld form, you might first have to script a Press Toolbar Button command in OneWorld Scripting Tool in order to move to a form that has a grid.

This topic covers the following tasks:

- Typing grid column inputs
- Typing inputs to additional grid rows
- Scripting a playback loop

▶ To type grid column inputs

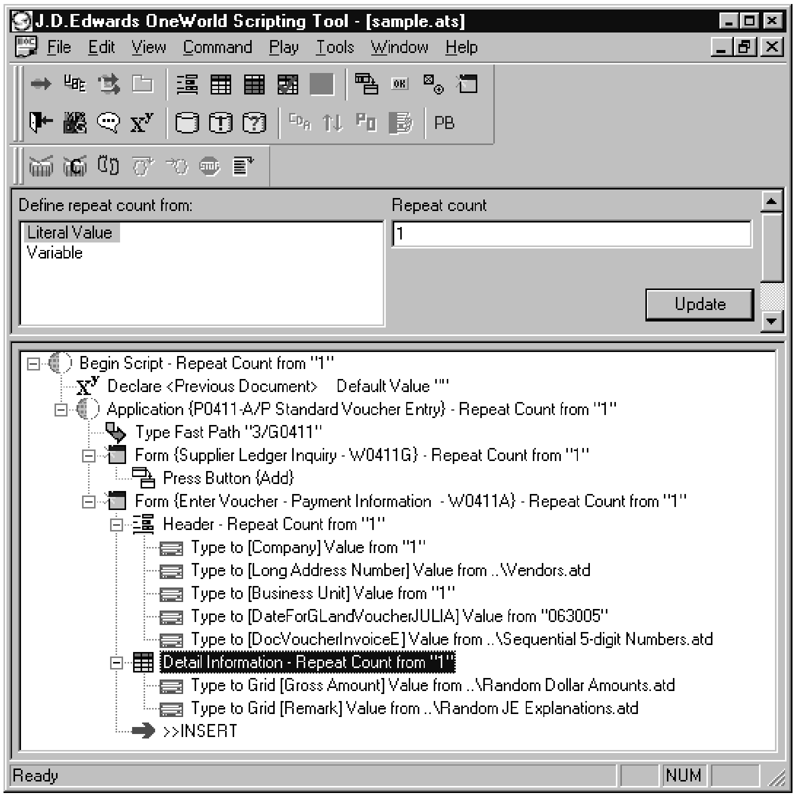
1. In the command menu of the OneWorld Scripting Tool form, click Set Grid Cell Value.
2. In the command pane, choose a grid column from the Grid Column list.

   A BlueCue appears in the appropriate grid column in the OneWorld form.
3. Choose a source of input.
4. Enter a literal value or choose the name of a valid values list, variable, or UDC visual assist value.
5. Click the Insert button.
6. Script inputs to additional grid columns by clicking the name of a different column in the Grid Column list and repeating steps 1-5.

▶ To type inputs to additional grid rows

1. From the script pane in OneWorld Scripting Tool, click a Detail Information command line you have inserted to the script.

   The insertion cursor is connected to the node you chose.

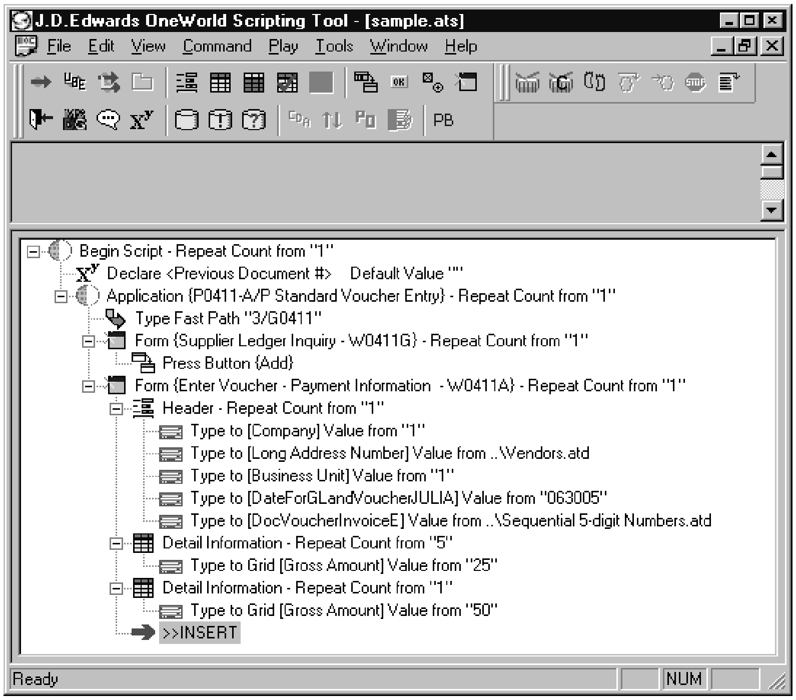


Define repeat count from:
Literal Value
Variable

Repeat count
1

```
Begin Script - Repeat Count from "1"
    Declare <Previous Document>    Default Value ""
    Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
        Type Fast Path "3/G0411"
        Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
            Press Button {Add}
        Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
            Header - Repeat Count from "1"
                Type to [Company] Value from "1"
                Type to [Long Address Number] Value from ..\Vendors.atd
                Type to [Business Unit] Value from "1"
                Type to [DateForGLandVoucherJULIA] Value from "063005"
                Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
            Detail Information - Repeat Count from "1"
                Type to Grid [Gross Amount] Value from ..\Random Dollar Amounts.atd
                Type to Grid [Remark] Value from ..\Random JE Explanations.atd
            >>INSERT
```

2. In the command menu of the OneWorld Scripting Tool form, click Set Grid Cell Value.
3. In the command pane, choose a grid column from the Grid Column list.
4. Choose a source of input.
5. Enter a literal value or choose the name of a valid values list, variable, or UDC visual assist value.
6. Click the Insert button.

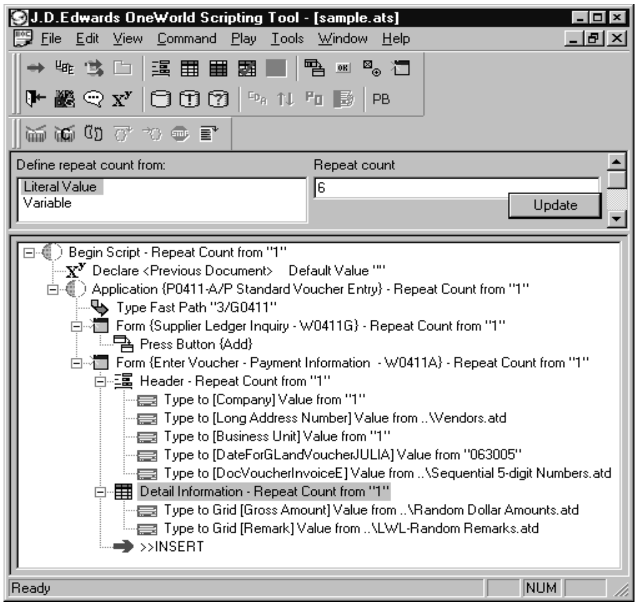

J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
    Declare <Previous Document #>   Default Value ""
    Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
        Type Fast Path "3/G0411"
        Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
            Press Button {Add}
        Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
            Header - Repeat Count from "1"
                Type to [Company] Value from "1"
                Type to [Long Address Number] Value from ..\Vendors.atd
                Type to [Business Unit] Value from "1"
                Type to [DateForGLandVoucherJULIA] Value from "063005"
                Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
            Detail Information - Repeat Count from "5"
                Type to Grid [Gross Amount] Value from "25"
            Detail Information - Repeat Count from "1"
                Type to Grid [Gross Amount] Value from "50"
            >>INSERT
```

7. Repeat steps 1-6 each time you want to script commands to a new row.

▶ To script a playback loop

1. Follow the steps for typing inputs to a grid row.
2. Turn off the Playback button in the cool bar (optional).
3. In the script pane, click the Detail Information command line that you want to play multiple times.
4. In the command pane, choose a source of input, such as literal value or variable, in the Define repeat count from list.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Update
Ready
NUM

Define repeat count from:
Literal Value
Variable

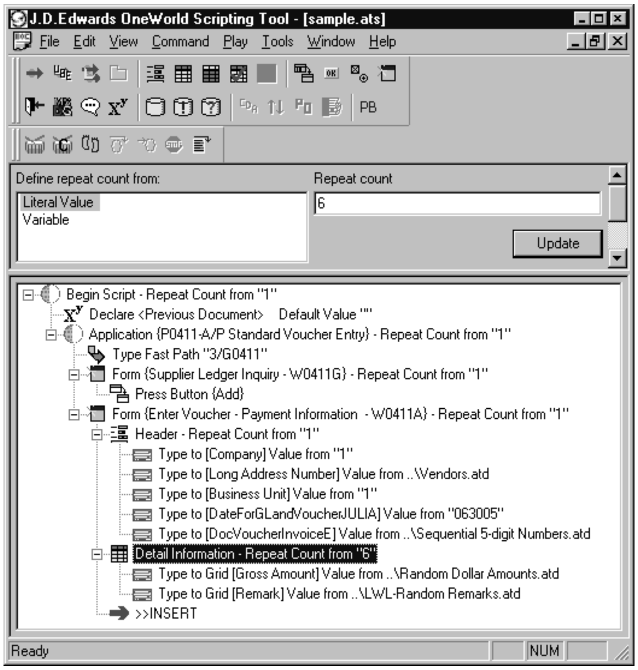
Repeat count
6

```
Begin Script - Repeat Count from "1"
  Declare <Previous Document>    Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        Type to [Long Address Number] Value from ..\Vendors.atd
        Type to [Business Unit] Value from "1"
        Type to [DateForGLandVoucherJULIA] Value from "063005"
        Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
      Detail Information - Repeat Count from "1"
        Type to Grid [Gross Amount] Value from ..\Random Dollar Amounts.atd
        Type to Grid [Remark] Value from ..\LWL-Random Remarks.atd
      >>INSERT
```

5. Enter a literal value or choose the name of a variable you created.

   The value specifies the number of times you want the inputs to loop, or to be entered in successive grid rows.

6. Click the Update button.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Update
Ready
NUM

Define repeat count from:
Literal Value
Variable

Repeat count
6

```
Begin Script - Repeat Count from "1"
  Declare <Previous Document>    Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        Type to [Long Address Number] Value from ..\Vendors.atd
        Type to [Business Unit] Value from "1"
        Type to [DateForGLandVoucherJULIA] Value from "063005"
        Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
      Detail Information - Repeat Count from "6"
        Type to Grid [Gross Amount] Value from ..\Random Dollar Amounts.atd
        Type to Grid [Remark] Value from ..\LWL-Random Remarks.atd
      >>INSERT
```

7. Click the Playback button (optional).

   During script playback, OneWorld Scripting Tool enters the inputs for a single row of the grid as many times as you assigned in the Repeat Count list. For example, if the repeat count for Row 1 was 3, Rows 1 through 3 fills with the inputs originally scripted for Row 1.

Typing Data to a QBE Line

You can script commands to enter data to the QBE line of a detail area in a Find/Browse form. You use the command pane lists to script inputs to the QBE line of a OneWorld form. Clicking Set QBE Cell Value in the command menu establishes the QBE line as the context in which you type data. The selections that you make from the lists in the command pane create the Type to action command. When you click the Insert button, the command line specifying the QBE line of the grid as the context appears as a node. The Type to command is indented beneath and attached to the node. The command line identifies the grid column that you chose and the value you typed to it. OneWorld Scripting Tool inserts subsequent Type to commands as leaves connected to the node. You can script different inputs to multiple rows of the grid, or you can script the input to one row multiple times, using a playback loop in OneWorld Scripting Tool.

▶ To type data to a QBE line

1. In the OneWorld Scripting Tool form, if you are not already in a Find/Browse form, script the commands to make one active in OneWorld.
2. From the command menu, click Set QBE Cell Value.
3. In the Grid Column list in the command pane, choose the name of a grid column.
4. Choose a source of input.
5. Enter a literal value, or the name of a valid values list, variable, or UDC visual assist value.
6. Click the Insert button.
7. To script an input to the QBE line of another grid column, choose another name in the Grid Column list and repeat steps 2-6.
8. In the command menu, click Press Toolbar Button.
9. Choose Press Standard Button.
10. Choose Find.
11. Click the Insert button.

The Select Grid Line Command

The Select Grid Line command allows you to perform and to test several important OneWorld functions. You use it to work within a detail area of a OneWorld form. With the grid filled, you can select records, delete them, add to a grid row, or edit the entries to a grid row. You can also script an exit to another form in another application. You use this command, therefore, in conjunction with several other action and context commands, including Press Toolbar Button, Application Interconnect, and Grid Data.

OneWorld Scripting Tool allows you to click either a row or a grid cell, specify the row number or grid column and perform a specific action, such as double clicking the row or editing the content of the cell.

This chapter covers the four key components of the command pane that appear when you click the Select Grid Line command:

- ❑ Operation Type list
- ❑ Action on Grid Row list
- ❑ Grid Columns list
- ❑ Source of Row Number list

Note that when you choose the Select Grid Line command, OneWorld Scripting Tool also populates the command pane with a value selection list that allows you to enter a literal value or choose the name of a previously created valid values list or variable. The caption of this list changes to reflect the choice you make in the Source of Row Number list.

Operation Type List

The Operation Type List in the OneWorld Scripting Tool command pane allows you to choose a row in a grid detail area either by specifying a row number or by specifying the value of a particular cell in a particular grid column. The Operations Type list contains two options:

- ❑ Click by Row Number option
- ❑ Click by Cell Content option

Click by Row Number Option

The Click by Row Number option allows you to choose a grid row by number. Because you specify a single number, this option works particularly well for grid detail areas with a large number of filled rows. OneWorld Scripting Tool simply finds the designated row and performs the action you have chosen in the Action on Grid Row list. You can designate the grid row either by entering a literal value, a value from a valid values list, or a value from a variable to the value selection list.

Click by Cell Content Option

You might want to choose a grid line that contains a particular value, such as an item number. In that case, you can use the Click by Cell Content option and choose a grid column and an action that you want to perform on the cell. OneWorld Scripting Tool selects the grid cell rather than the entire grid row.

Action on Grid Row List

The Operations Options list allows you to specify what you want to do in choosing the row. There are four types of grid row operations that you can script in OneWorld Scripting Tool:

❑ Single click a grid row

❑ Single click a grid row button

❑ Double click a grid row

❑ Double click a grid row button

❑ Position grid row for add/edit

Single Click a Grid Row

You write a command to single click a grid row when the form that is active in OneWorld does not have a row button. You can write this command when the active form has a row button, but it is not advisable in that situation because sometimes single clicking the grid row selects only a cell. After you have selected the row, you can write a command to press the Select or Delete button.

Single Click a Grid Row Button

You write a command to single click the grid row button when the form that is active in OneWorld contains a media object button. You do not script this grid row operation in forms that do not contain a button. After you have selected the row, you can write a command to press the Select or Delete button.

Double Click a Grid Row

If you want to move from the grid detail area of one form to another OneWorld form, or exit to a new application, you write a command to double click the grid row.

Double Click a Grid Row Button

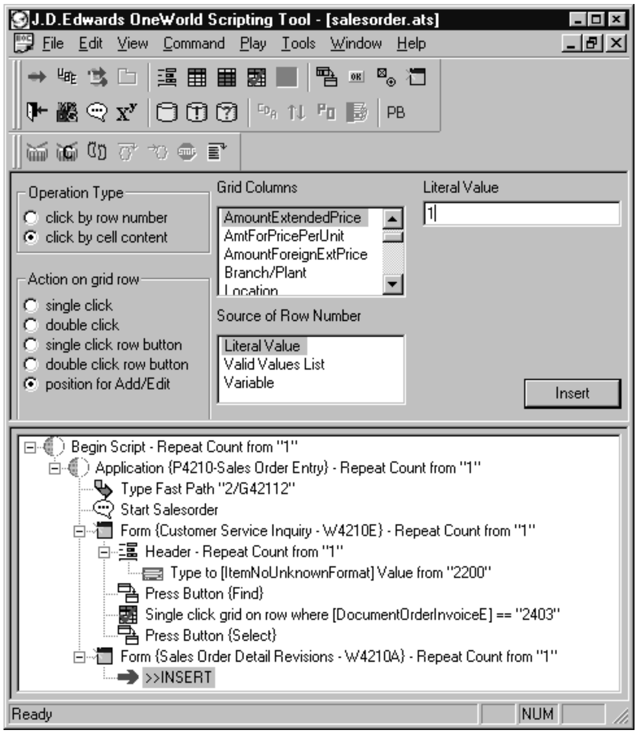
You write a command to double click the grid row button when the form that is active in OneWorld contains a media object button. You do not script this grid row operation in forms that do not contain a button.

Position Grid Row for Add/Edit

To enter changes to the grid detail area of a form, you write a command to position the grid row for an add or an edit. You use the command to add an entry to a grid line or to edit an existing value in a grid cell.

Grid Columns List

You use the Grid Columns list only when you have chosen the Click by Cell Content option. When you choose this option, OneWorld Scripting Tool populates the Grid Columns list with the names of all the columns of the grid that is active in OneWorld. You can scroll through this list to find the name of a column.


J.D.Edwards OneWorld Scripting Tool - [salesorder.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

Operation Type
○ click by row number
● click by cell content

Action on grid row
○ single click
○ double click
○ single click row button
○ double click row button
● position for Add/Edit

Grid Columns
AmountExtendedPrice
AmtForPricePerUnit
AmountForeignExtPrice
Branch/Plant
Location

Literal Value
1

Source of Row Number
Literal Value
Valid Values List
Variable

```
Begin Script - Repeat Count from "1"
    Application {P4210-Sales Order Entry} - Repeat Count from "1"
        Type Fast Path "2/G42112"
        Start Salesorder
        Form {Customer Service Inquiry - W4210E} - Repeat Count from "1"
            Header - Repeat Count from "1"
                Type to [ItemNoUnknownFormat] Value from "2200"
            Press Button {Find}
            Single click grid on row where [DocumentOrderInvoiceE] == "2403"
            Press Button {Select}
        Form {Sales Order Detail Revisions - W4210A} - Repeat Count from "1"
            >>INSERT
```

Once you choose it, you can specify in the Source of Row Number list a value that might exist in a particular cell of that column. OneWorld Scripting Tool searches for the value and selects the first row that it finds that contains that value in the specified column.

Source of Row Number List

In the Source of Row Number list, you choose the source of the value you use to select the grid row. You can choose Literal Value, in which case you simply type a row number or a grid cell's value into the value selection list. You can choose Valid Values List, in which case you choose from the value selection list a valid values list you previously created. OneWorld Scripting Tool uses the first value in the list to select a grid row either by row number or by cell content. You can choose Variable, in which case you choose from the Value Selection list a variable whose name you have declared and whose value you have set. OneWorld Scripting Tool uses this value to select a grid row either by row number or by cell content.

Scripting the Select Grid Line Command

To begin the scripting, you should have a Find/Browse form active in OneWorld. You fill the detail area by writing a command to click Find. With the grid filled, you choose in the OneWorld Scripting Tool command pane the row that you want to target for your script and the operation that you want to perform on that row. With the grid filled, you can select a grid row, either by row number or by cell content. If you choose to click a grid cell, you must choose a grid column.

In either case, you choose the type of operation you want to perform on the grid row. There are four options: single click the row, single click the row button, single click and perform an add/edit, or double click the row.

To complete scripting the Select Grid Line command, you choose a source of value for row selection. These sources include a literal value, a valid values list, or a variable. You then either enter a literal value to the value selection list or choose from the value selection list a valid values list or variable.

This chapter covers the following tasks:

❑ Clicking by row number

❑ Clicking by cell content

❑ Performing grid row operations

Clicking by Row Number

After you have filled the grid in a OneWorld form, you can use OneWorld Scripting Tool to select a row by searching for a row number you designate. To complete this command, you must also choose an action that you want OneWorld Scripting Tool to perform on the grid row, choose a source of value that OneWorld Scripting Tool uses to select the row, and choose or enter the value of the row.

When you choose the Select Grid Line command, OneWorld Scripting Tool populates the command pane with five lists:

- Operation Type
- Action on Grid Row
- Grid Columns

- Source of Row Number
- Value selection list

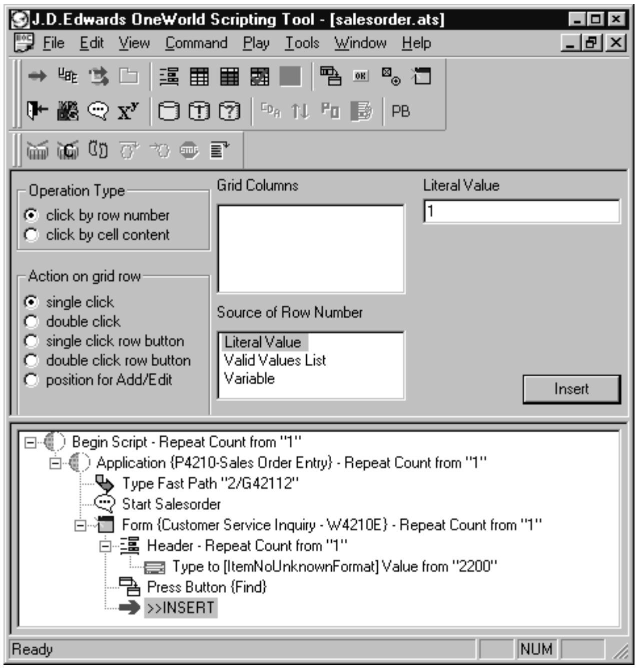
You make choices from these lists to write a command to click a grid row by row number.

▶ To click a grid row by row number

1. In the command menu of the OneWorld Scripting Tool form, click Press Button.
2. Choose Press Standard Button.
3. Choose Find.

   OneWorld Scripting Tool fills the detail area in the active OneWorld form.

4. In the command menu of the OneWorld Scripting Tool form, choose Select Grid Line.
5. In the OneWorld Scripting Tool command pane, choose Click by Row Number from the Operation Type list.
6. Choose a grid row action from the Action on Grid Row list.
7. Choose a value source from the Source of Row Number list.
8. In the value selection list, enter a literal value or choose the name of a valid values list or variable.



Operation Type
● click by row number
○ click by cell content

Action on grid row
● single click
○ double click
○ single click row button
○ double click row button
○ position for Add/Edit

Grid Columns

Literal Value: 1

Source of Row Number
Literal Value
Valid Values List
Variable

```
⊟ Begin Script - Repeat Count from "1"
  ⊟ Application {P4210-Sales Order Entry} - Repeat Count from "1"
      Type Fast Path "2/G42112"
      Start Salesorder
    ⊟ Form {Customer Service Inquiry - W4210E} - Repeat Count from "1"
      ⊟ Header - Repeat Count from "1"
          Type to [ItemNoUnknownFormat] Value from "2200"
        Press Button {Find}
        >>INSERT
```

9. Click the Insert button.

Clicking by Cell Content

You might want OneWorld Scripting Tool to search the grid detail area for a particular value, then select the row once it has found a cell that contains that value. In this case, you can write a command to click the grid row using cell content, rather than row number, as the clicking criterion.

Writing a command to select a grid line based on cell content requires that you take the same steps that you took in writing a command to select the line based on row number. However, when you select based on cell content, you must also choose a grid column as a search criterion.

▶ To click a grid row by cell content

1. In the command menu of the OneWorld Scripting Tool form, click Press Button.
2. Choose Press Standard Button.
3. Choose Find.

   OneWorld Scripting Tool fills the detail area in the active OneWorld form.

4. In the command menu of the OneWorld Scripting Tool form, choose Select Grid Line.
5. In the OneWorld Scripting Tool command pane, choose Click by Cell Content from the Operation Type list.
6. Choose a grid row action from the Action on Grid Row list.
7. In the Grid Columns list, choose a grid column on which you want OneWorld Scripting Tool to search for a value.
8. Choose a value source from the Source of Row Number list.
9. In the value selection list, enter a literal value or choose the name of a valid values list or variable.
10. Click the Insert button.

Performing Grid Row Operations

Once you have selected a row, you can perform five different operations on it:

- Single click a grid row
- Single click a grid row button
- Double click a grid row
- Double click a grid row button

- Position a grid row for add/edit

Note that you choose the single click and double click grid row operations especially when you are writing commands to test a OneWorld form containing a grid detail area that does not have row buttons.

You can use these options as preludes to writing other OneWorld Scripting Tool commands. For example, single clicking on a grid row or grid row button enables you to write a command to press the OneWorld Select or Delete button. double clicking a row enables you to move to another OneWorld form or application. Clicking on a row and positioning for add or edit allows you to make changes to the grid row that you have selected in a Fix/Inspect form.

▶ To single click a grid row

1. In the command menu of the OneWorld Scripting Tool form, click Select Grid Line.
2. Choose an operation from the Operation Type list.
3. Click the single click option in the Action on Grid Row list.
4. Choose a value source from the Source of Row Number list
5. Enter a literal value or choose a valid values list or variable in the value selection list.
6. Click the Insert button.

▶ To single click a grid row button

1. In the command menu of the OneWorld Scripting Tool form, click Select Grid Line.
2. Choose an operation from the Operation Type list.
3. Click the single click row button option in the Action on Grid Row list.
4. Choose a value source from the Source of Row Number list
5. Enter a literal value or choose a valid values list or variable in the value selection list.
6. Click the Insert button.

▶ To double click a grid row

1. In the command menu of the OneWorld Scripting Tool form, click Select Grid Line.
2. Choose an operation from the Operation Type list.
3. Click the double click option in the Action on Grid Row list.

4. Choose a value source from the Source of Row Number list
5. Enter a literal value or choose a valid values list or variable in the value selection list.
6. Click the Insert button.
7. In the command menu, click Form or Application Interconnect.
8. In the command pane, choose the next form or application that resulted from double clicking the row in the OneWorld form.
9. Click the Insert button.

▶ To double click a grid row button

1. In the command menu of the OneWorld Scripting Tool form, click Select Grid Line.
2. Choose an operation from the Operation Type list.
3. Choose the double click row button option in the Action on Grid Row list.
4. Choose a value source from the Source of Row Number list
5. Enter a literal value or choose a valid values list or variable in the value selection list.
6. Click the Insert button.
7. In the command menu, click Form or Application Interconnect.
8. In the OneWorld Scripting Tool command pane, choose the next form or application that resulted from double clicking the row button in the OneWorld form.
9. Click the Insert button.

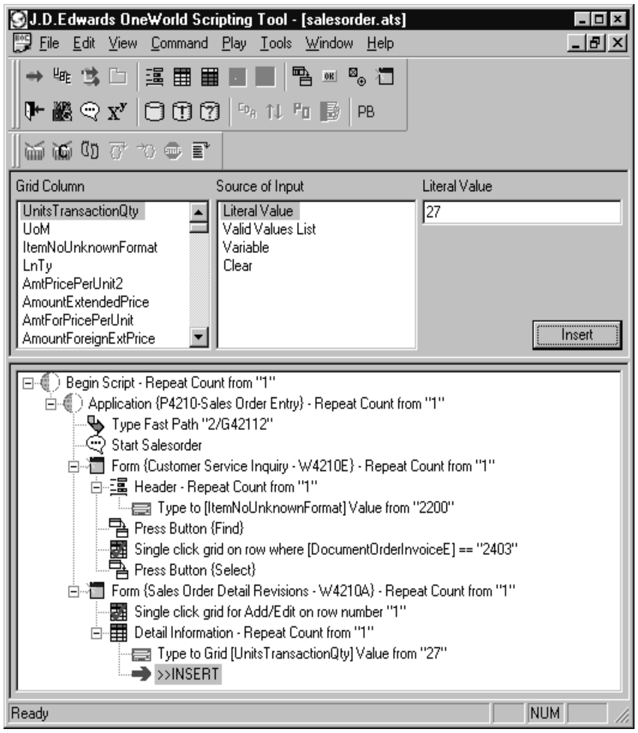
▶ To position a grid row add or edit

1. In the command menu of the OneWorld Scripting Tool form, click Select Grid Line.
2. Choose an operation from the Operation Type list.
3. Choose Position for Add/Edit in the Action on Grid Row list.
4. Choose a value source from the Source of Row Number list
5. Enter a literal value or choose a valid values list or variable in the value selection list.

   Note: If you are adding a row, enter a high row number, such as 999, to ensure that you reach the bottom of the grid when OneWorld Scripting Tool inserts the command.

6. In the OneWorld Scripting Tool command menu, click Set Grid Cell Value.

7. Choose a grid column.
8. Choose a source of input.
9. Enter a literal value or UDC value, or choose a valid values list or a variable from the value selection list.
10. Click the Insert button.


J.D.Edwards OneWorld Scripting Tool - [salesorder.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

Grid Column: UnitsTransactionQty, UoM, ItemNoUnknownFormat, LnTy, AmtPricePerUnit2, AmountExtendedPrice, AmtForPricePerUnit, AmountForeignExtPrice

Source of Input: Literal Value, Valid Values List, Variable, Clear

Literal Value: 27

```
Begin Script - Repeat Count from "1"
  Application {P4210-Sales Order Entry} - Repeat Count from "1"
    Type Fast Path "2/G42112"
    Start Salesorder
    Form {Customer Service Inquiry - W4210E} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [ItemNoUnknownFormat] Value from "2200"
      Press Button {Find}
      Single click grid on row where [DocumentOrderInvoiceE] == "2403"
      Press Button {Select}
    Form {Sales Order Detail Revisions - W4210A} - Repeat Count from "1"
      Single click grid for Add/Edit on row number "1"
      Detail Information - Repeat Count from "1"
        Type to Grid [UnitsTransactionQty] Value from "27"
        >>INSERT
```

OneWorld Scripting Tool adds to or edits the column in the grid row that you selected.

The Press Toolbar Button Command

The Press Toolbar Button command allows you to script many of the important actions that you can take in a OneWorld form, including:

- Moving from one form to another within the same application
- Moving to a form in a new application either through a form or a row exit
- Filling a grid in a form
- Choosing a row in a grid
- Updating the OneWorld database
- Deleting a record
- Exiting a form
- Choosing a grid tab
- Clicking the grid scroll button
- Submitting UBEs that are not hard-coded to run automatically

This chapter discusses the four choices for writing button-clicking commands with OneWorld Scripting Tool:

❑ The Standard Button option

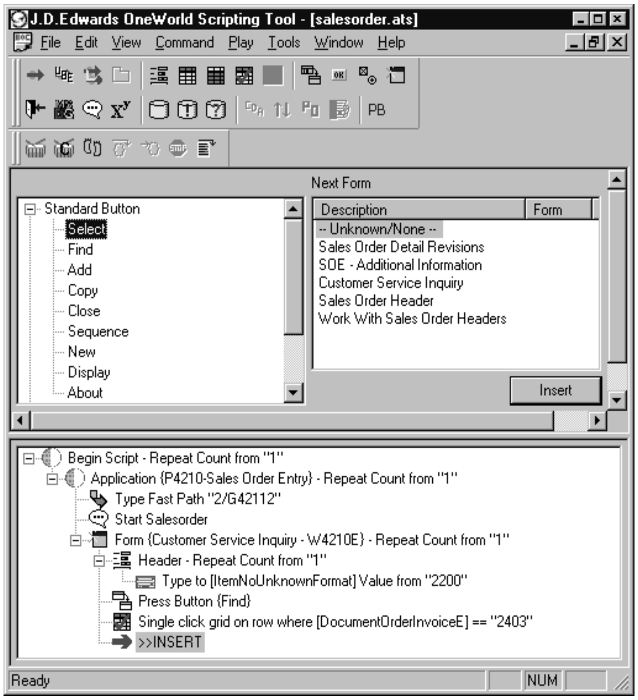
❑ The Custom Button option

❑ The Select Grid Tab option

❑ The Grid Scroll Button option

The Standard Button Option

The Standard Button option in OneWorld Scripting Tool contains button-clicking choices that mirror the buttons in the cool bar of OneWorld forms. For example, the cool bar in some OneWorld forms, such as Find/Browse forms, features 10 button-clicking options. When such a form is active in OneWorld, the Standard Button tree in the Button list in OneWorld Scripting Tool contains the same options.



J.D.Edwards OneWorld Scripting Tool - [salesorder.ats]
File Edit View Command Play Tools Window Help
PB
Next Form
Insert
Ready
NUM

- Standard Button
  - Select
  - Find
  - Add
  - Copy
  - Close
  - Sequence
  - New
  - Display
  - About

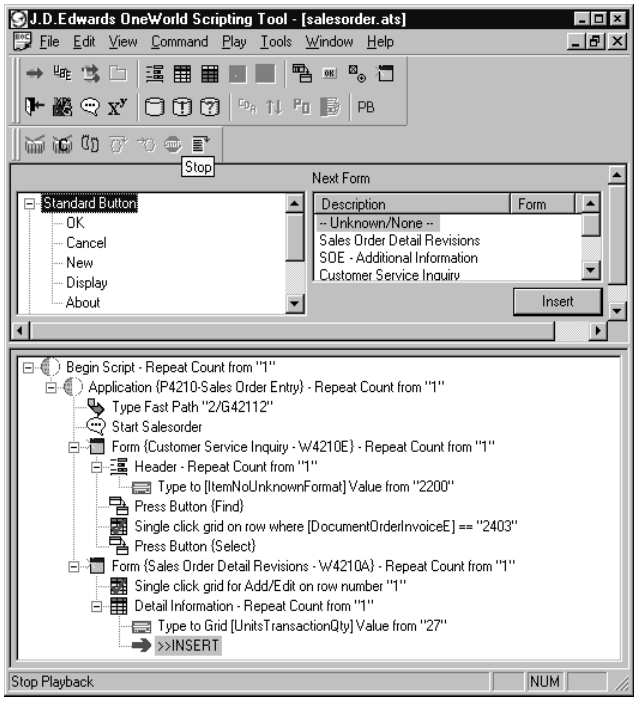
Description	Form
-- Unknown/None --	
Sales Order Detail Revisions	
SOE - Additional Information	
Customer Service Inquiry	
Sales Order Header	
Work With Sales Order Headers	

```
Begin Script - Repeat Count from "1"
  Application {P4210-Sales Order Entry} - Repeat Count from "1"
    Type Fast Path "2/G42112"
    Start Salesorder
    Form {Customer Service Inquiry - W4210E} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [ItemNoUnknownFormat] Value from "2200"
      Press Button {Find}
      Single click grid on row where [DocumentOrderInvoiceE] == "2403"
      >>INSERT
```

The cool bar in other OneWorld forms features fewer button-clicking options. When such a form is active in OneWorld, the Standard Button tree in the Button list in OneWorld Scripting Tool again contains the same options that you find in the OneWorld form.

When you script one of these button-clicking options, OneWorld Scripting Tool runs the command exactly as it would be run in OneWorld. For example, You might script clicking OK to update the OneWorld database after you have entered new data to a form.

When you choose the Standard Button option, the Next Form list also appears. The choice you make from this list specifies the OneWorld form that becomes or remains active when you click the Insert button.


J.D.Edwards OneWorld Scripting Tool - [salesorder.ats]
File Edit View Command Play Tools Window Help
PB
Stop
Next Form
Description
Form
Insert
NUM

```
Standard Button
    OK
    Cancel
    New
    Display
    About
```

Description	Form
-- Unknown/None --	
Sales Order Detail Revisions	
SOE - Additional Information	
Customer Service Inquiry	

```
Begin Script - Repeat Count from "1"
  Application {P4210-Sales Order Entry} - Repeat Count from "1"
    Type Fast Path "2/G42112"
    Start Salesorder
    Form {Customer Service Inquiry - W4210E} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [ItemNoUnknownFormat] Value from "2200"
      Press Button {Find}
      Single click grid on row where [DocumentOrderInvoiceE] == "2403"
      Press Button {Select}
    Form {Sales Order Detail Revisions - W4210A} - Repeat Count from "1"
      Single click grid for Add/Edit on row number "1"
      Detail Information - Repeat Count from "1"
        Type to Grid [UnitsTransactionQty] Value from "27"
        >>INSERT
```

Stop Playback

Note that you use the Standard Button option when your script includes forms that are in the same application. However, occasionally clicking a standard button, such as Add, might take you to another OneWorld application. In this case, you must write an Application Interconnect command.

When you launch a UBE, OneWorld Scripting Tool's standard button choices again mirror the buttons in the menu bar of the OneWorld form. For example, you might launch and need to submit a UBE using the Version Prompting form. In OneWorld Scripting Tool, you simply choose Submit from the Standard Button options.

See Also

- *UBE Submission*
- *Submitting a UBE*

The Custom Button Option

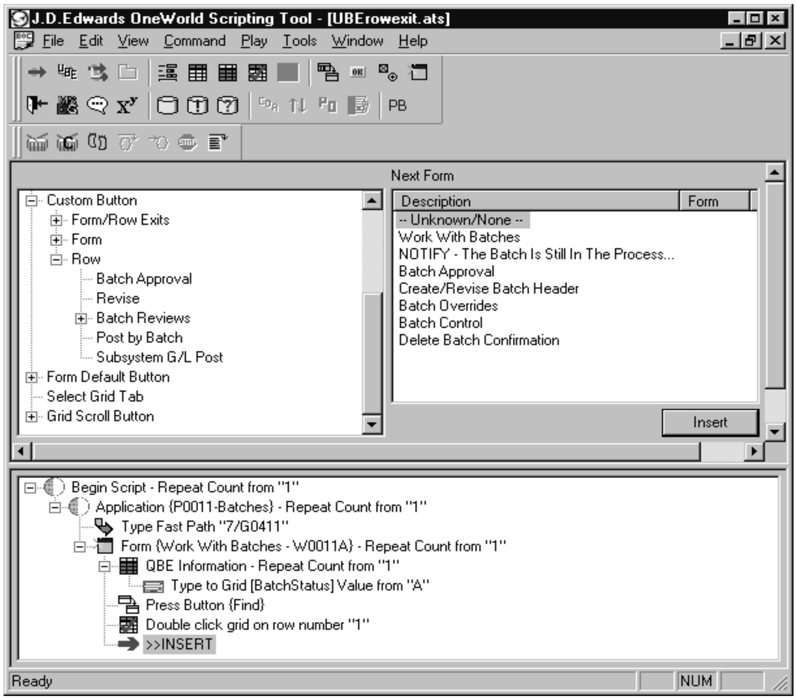
While you use the Standard Button option if you want to script moving from form to form within the same application, you use the Custom Button option if you want to script the interconnecting of applications. You use the Custom Button option to script the exit from one application or form in OneWorld to another. The exits can be either form exits or row exits.

You should also distinguish the Custom Button option from the Application Interconnect command. You use both to interconnect applications. However, you script an Application Interconnect command *after* you have exited to a new application in OneWorld, perhaps by clicking the Add button. The Custom Button option can be used *before* you have exited to a new application. The option allows you to choose the application and form in the command pane and insert the commands. OneWorld Scripting Tool interconnects the applications, and the form in the new application in OneWorld appears.

When you choose Custom Button from the Button list in the command pane, two options appear in the tree:

- Form exit
- Row exit

Clicking either or both further expands the tree, revealing the OneWorld forms to which you can elect to exit:


J.D.Edwards OneWorld Scripting Tool - [UBErowexit.ats]
File Edit View Command Play Tools Window Help
PB
Next Form
Insert
Ready
NUM

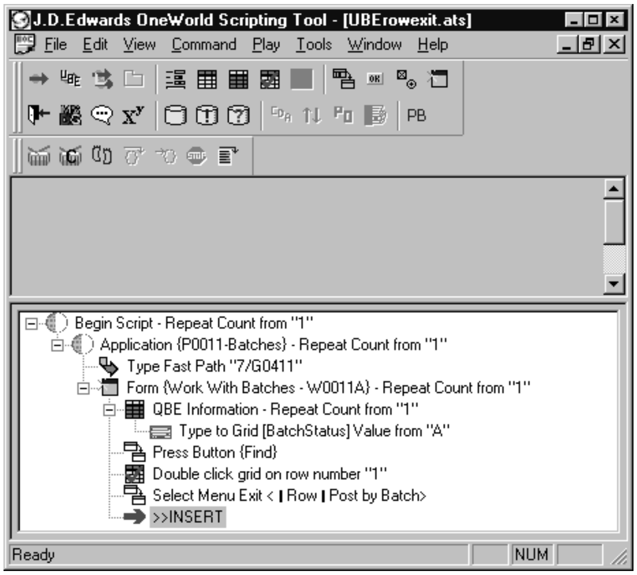
- Custom Button
  - Form/Row Exits
  - Form
  - Row
    - Batch Approval
    - Revise
    - Batch Reviews
    - Post by Batch
    - Subsystem G/L Post
- Form Default Button
- Select Grid Tab
- Grid Scroll Button

Description	Form
-- Unknown/None --	
Work With Batches	
NOTIFY - The Batch Is Still In The Process...	
Batch Approval	
Create/Revise Batch Header	
Batch Overrides	
Batch Control	
Delete Batch Confirmation	

```
Begin Script - Repeat Count from "1"
  Application {P0011-Batches} - Repeat Count from "1"
    Type Fast Path "7/G0411"
    Form {Work With Batches - W0011A} - Repeat Count from "1"
      QBE Information - Repeat Count from "1"
        Type to Grid [BatchStatus] Value from "A"
      Press Button {Find}
      Double click grid on row number "1"
      >>INSERT
```

Note that the options under Form Exit and Row Exit mirror the choices found in the drop-down menu if you click Form or Row in the menu bar of the active OneWorld form. Note that when you insert a command to press a standard button and to press a custom button, the same symbol appears in the command lines of the script. However, the command line for clicking a standard button describes the command as Press Toolbar Button, whereas the command line for clicking a custom button describes the command as Select Menu Exit. This shows that when you press a standard button, you travel between or within forms in the same application. When you press a custom button, you travel through the OneWorld menu and generally change applications.

Note also that the command line for clicking a custom button contains the type of menu exit, either form or row, and the name of the menu item that you select. The Select Menu Exit command line is followed by the Application Interconnect command line, which records the application to which you have exited. OneWorld Scripting Tool also inserts a Form command line, which records the form that is now active in OneWorld.



```
Begin Script - Repeat Count from "1"
    Application {P0011-Batches} - Repeat Count from "1"
        Type Fast Path "7/G0411"
        Form {Work With Batches - W0011A} - Repeat Count from "1"
            QBE Information - Repeat Count from "1"
                Type to Grid [BatchStatus] Value from "A"
            Press Button {Find}
            Double click grid on row number "1"
            Select Menu Exit < | Row | Post by Batch>
            >>INSERT
```

Form Exit

OneWorld Scripting Tool allows you to run a form exit when you want to move to a OneWorld form that is related to the one on which you have been working. However, that form may be in a different application and the exit represents a change in the standard sequence of forms encountered in working on a OneWorld transaction.

Row Exit

You can use OneWorld Scripting Tool to run a row exit when you want to move from a chosen grid row in a OneWorld form to a related OneWorld form. That form may be in a different application, and the exit again represents a change in the standard sequence of forms encountered in working on a OneWorld transaction. In OneWorld Scripting Tool, you choose Row Exit from the Custom Button list only when you fill the detail area of a Find/Browse form in OneWorld and choose a row.

You might also use the Custom Button option to run a row exit that launches a UBE version. In this case, once you script the row exit, you write a UBE command, and OneWorld Scripting Tool automatically submits the UBE.

See Also

- *Launching a UBE from a Row Exit*

The Select Grid Tab Option

You use the Select Grid Tab option in OneWorld Scripting Tool to test OneWorld's ability to move to customized grid tabs that you have created. In OneWorld, you customize the grid by choosing Preferences, Grid, and New Format. Your customization of fonts, number of grid columns, width of grid columns, and so on creates a new tab. Each grid customization creates a new tab. After you have created as many tabs as you like in OneWorld, you can use OneWorld Scripting Tool to script selecting a tab or tabs. Not only can you see if the tab is chosen, you can also see if your customized changes are in place.

The Grid Scroll Button Option

The Grid Scroll Button option can be used in OneWorld Scripting Tool to script clicking the up and down arrows in a OneWorld grid. OneWorld Scripting Tool moves the arrows up or down by line or by page.

Scripting the Press Toolbar Button Command

You use the Press Toolbar Button command in OneWorld Scripting Tool to script many important functions. For example, clicking Add allows you to move to a new form, either in the same application or a new one. By choosing the Standard Button option of the Press Toolbar command, you can write a script command to click Add. Other Press Toolbar Button options allow you to perform form and row exits, select a grid tab or scroll through a grid.

This chapter discusses using the button-clicking options in OneWorld Scripting Tool to complete the following tasks:

- ❑ Clicking a standard button
- ❑ Clicking a custom button
- ❑ Selecting a grid tab
- ❑ Clicking the grid scroll button

Clicking a Standard Button

In general, you choose the Standard Button option when you want to click a toolbar button in a OneWorld form. When you choose this option, the choices in the command pane match the toolbar buttons in the form that is active in OneWorld.

This topic covers the following tasks:

- Moving from one form to another using the Add button
- Moving from one form to another using the Select button
- Updating the database
- Filling a grid
- Deleting a record
- Exiting a form

See Also

- *Submitting a UBE*

- *Selecting Data for a UBE*

▶ To move from one form to another using the Add button

1. In the command menu of the OneWorld Scripting Tool form, click Press Toolbar Button.
2. Choose Standard Button.
3. Choose Add.
4. Choose the name of a OneWorld form from the Next Form list or choose Unknown/None for another application.

   Caution: If you choose Unknown/None, then click the Insert button, you need to ensure that the Form command line in OneWorld Scripting Tool mirrors the form that is now active in OneWorld. Click Form in the command menu, then choose the name of the active OneWorld form and click the Insert button.

   Once you have selected the row, you can write a command to click Select or Delete.

5. Click the Insert button.

   The next form appears in OneWorld.

   If you have chosen Unknown/None and the form that appears in OneWorld is part of a different application, be sure to script an Application Interconnect command, following these steps:

6. In OneWorld Scripting Tool, click Application Interconnect in the command menu.
7. From the Application list in the command pane, choose the name of the application active in OneWorld.
8. From the Form list in the command pane, choose the name of the form active in OneWorld.
9. Click the Insert button.

   You can now script commands to be run in a new OneWorld form.

▶ To move from one form to another using the Select button

1. In the command menu of the OneWorld Scripting Tool form, click Select Grid Line.
2. In the Source of Row Number list in the command pane, choose literal value, valid values list, or variable.

3. Enter or choose a value from the value selection list.

   This specifies the row number to select.

4. Click the Insert button.
5. From the command menu, click Press Toolbar Button.
6. Choose Standard Button.
7. Choose Select.
8. In the Next Form list, choose the OneWorld form to appear next.
9. Click the Insert button.

   Note: Remember that if you choose Unknown/None from the Next Form list and the form that appears in OneWorld is part of a different application, you must script an Application Interconnect command.

▶ To update the database

1. In the command menu of the OneWorld Scripting Tool form, click Press Toolbar Button.
2. Choose Standard Button.
3. Choose OK.
4. If you want to move to another form, choose one from the Next Form list. If you want to stay on the same form, make no choice from the list.
5. Click the Insert button.

   See *Updating the Database* for more on writing commands to update the database.

▶ To fill a grid

1. In the command menu of the OneWorld Scripting Tool form, click Press Toolbar Button.
2. Choose Standard Button.
3. Choose Find.
4. Click the Insert button.

   See *Finding Records.* for more on querying the database.

▶ To delete a record

1. In the command menu of the OneWorld Scripting Tool form, click Select Grid Line.
2. In the Source of Row Number list in the command pane, choose literal value, valid values list, or variable.
3. Enter or choose a value from the value selection list to specify the row number to be highlighted.

   The row number should contain the record that you want to delete.

4. Click the Insert button.
5. With the row highlighted in OneWorld, click Press Toolbar Button in the OneWorld Scripting Tool command menu.
6. Choose Standard Button.
7. Choose Delete.
8. Click the Insert button.

   See *Selecting Records and Deleting Them from the Database.* for more on deleting records from the database.

▶ To exit a form

1. In the command menu of the OneWorld Scripting Tool form, click Press Toolbar Button.
2. Choose Standard button.
3. Choose Cancel. (In some forms, you choose Close.)
4. Click the Insert button.

Clicking a Custom Button

You use the Custom Button option to script row and form exits. These are commands that you cannot write using the Standard Button option. When you choose this option, the choices in the command pane match the form and row exits that are available in the OneWorld form that is active.

Remember that a row exit might result in an application interconnect. You can handle the application interconnect in the script by choosing from the command pane the row exit, the next form, and the application before you click the Insert button.

This topic covers the following tasks:

- Scripting a form exit
- Scripting a row exit

▶ To script a form exit

1. In the command menu of the OneWorld Scripting Tool form, click Press Toolbar Button.
2. Choose Custom Button.
3. Choose Form/Row Exits.
4. Click Form.
5. Choose a form exit.
6. Choose an application from the Application list.
7. From the Next Form list, choose a form, if necessary.
8. Click the Insert button.

▶ To script a row exit

1. In the command menu of the OneWorld Scripting Tool form, click Press Toolbar Button.
2. If the grid in the active OneWorld form is unfilled, choose Find.
3. Click the Insert button.

   The grid detail area fills in the OneWorld form.

4. In OneWorld Scripting Tool command menu, click Select Grid Line.

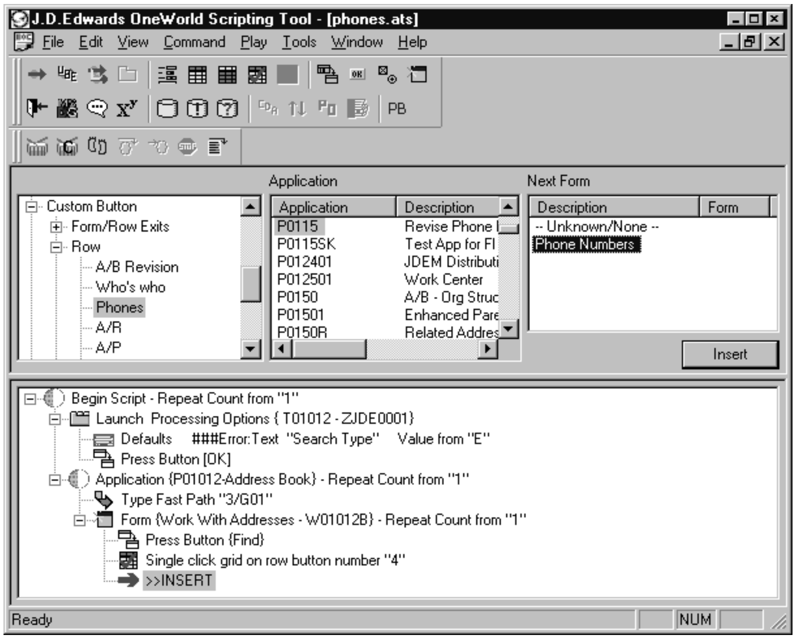
   Note: Here you can choose Row from the menu bar in OneWorld, click a form that is listed, or click Select, and bring up the new form. You can check the name of the newly active OneWorld application and form by clicking on About OneWorld in the menu bar.

5. In the Source of Row Number list in the command pane, choose literal value, valid values list, or variable.
6. Enter or choose a value from the value selection list to specify the row number to be highlighted.
7. Choose single click row button from the Operations options.
8. Click the Insert button.
9. In the OneWorld Scripting Tool command menu, click Press Toolbar Button.
10. From the Button list in the OneWorld Scripting Tool command pane, click Custom Button.

11. Click Form/Row Exits.
12. Click Row Exit.

    Note: If you run a row exit to launch a UBE, you do not not choose from the Application or Next Form lists that appear. You choose the row exit only, click the Insert button, and then write a UBE command.

13. If you are running an interactive application, choose an application from the Application list.
14. Choose a form name from the Next Form list.


J.D.Edwards OneWorld Scripting Tool - [phones.ats]
File Edit View Command Play Tools Window Help
PB
Application
Next Form
Insert
Ready
NUM

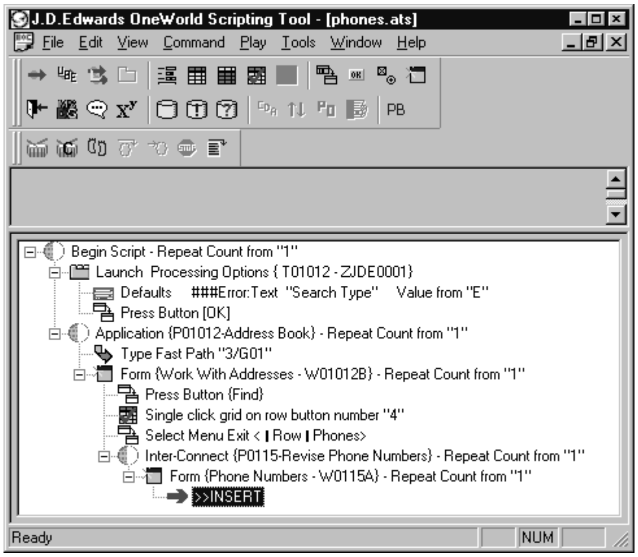
- Custom Button
  - Form/Row Exits
  - Row
    - A/B Revision
    - Who's who
    - Phones
    - A/R
    - A/P

Application	Description
P0115	Revise Phone I
P0115SK	Test App for FI
P012401	JDEM Distributi
P012501	Work Center
P0150	A/B - Org Struc
P01501	Enhanced Pare
P0150R	Related Addres

Description	Form
-- Unknown/None --	
Phone Numbers	

```
Begin Script - Repeat Count from "1"
    Launch  Processing Options { T01012 - ZJDE0001}
        Defaults    ###Error:Text  "Search Type"    Value from "E"
        Press Button [OK]
    Application {P01012-Address Book} - Repeat Count from "1"
        Type Fast Path "3/G01"
        Form {Work With Addresses - W01012B} - Repeat Count from "1"
            Press Button {Find}
            Single click grid on row button number "4"
            >>INSERT
```

15. Click the Insert button.


J.D.Edwards OneWorld Scripting Tool - [phones.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
    Launch  Processing Options { T01012 - ZJDE0001}
        Defaults    ###Error:Text  "Search Type"    Value from "E"
        Press Button [OK]
    Application {P01012-Address Book} - Repeat Count from "1"
        Type Fast Path "3/G01"
        Form {Work With Addresses - W01012B} - Repeat Count from "1"
            Press Button {Find}
            Single click grid on row button number "4"
            Select Menu Exit < | Row | Phones>
            Inter-Connect {P0115-Revise Phone Numbers} - Repeat Count from "1"
                Form {Phone Numbers - W0115A} - Repeat Count from "1"
                    >>INSERT
```

Selecting a Grid Tab

You can choose customized grid tabs that you have created in OneWorld by using the Select Grid Tab command. OneWorld Scripting Tool chooses the grid tab number you script and in playback mode displays the grid with your customized changes.

▶ To script the Select Grid Tab option

1. In a OneWorld form, create as many tabs as you desire.
2. In the OneWorld Scripting Tool command menu, click Press Toolbar Button.
3. In the Button list, click Select Grid Tab.
4. In the Literal Value list, enter the number of the grid tab you want to choose.
5. Click the Insert button.

Clicking the Grid Scroll Button

When you are working in a OneWorld form with a filled grid, you may want to scroll through the grid, from top to bottom or from page to page. You might script the Grid Scroll Button option to test OneWorld's ability to run the command, or you might include it as an integral part of a larger script.

▶ To script the Grid Scroll Button option

1. In the command menu of the OneWorld Scripting Tool form, click Press Toolbar Button
2. If you have not filled the grid in the active form in OneWorld, choose Standard Button.
3. Choose Find.
4. From the Button list of the command pane, click Grid Scroll Button.
5. In the command menu, click Select Grid Row.
6. In the command pane, under Operation Type, choose click by row number.
7. Under Action on grid row, choose single click.
8. Choose a source of row number and enter a literal value or choose a variable or valid values list.
9. In the command menu, click Press Toolbar button.
10. Choose Grid Scroll Button.

11. Choose one of the two scrolling options in the tree:
   - Page Up
   - Page Down
12. Click the Insert button.

The Press Push Button Command

You can use the Press Toolbar Button command to script clicking standard OneWorld toolbar buttons, running custom functions such as form exits and row exits, selecting grid tabs, and grid scroll buttons. However, some OneWorld applications utilize special buttons that do not reside on the OneWorld toolbar or menu bar.

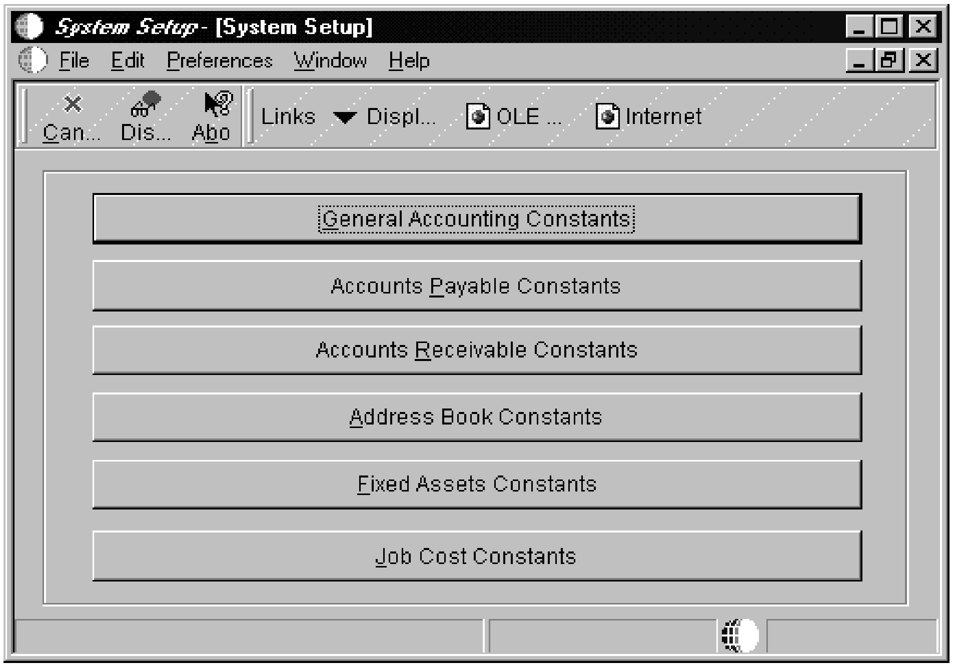
You use the Press Push Button command to script clicking these oversized push buttons and clickable bitmaps in OneWorld forms that use them. OneWorld Scripting Tool displays in the command pane the push button options and clickable bitmaps in the form that is active in OneWorld.

This chapter discusses the following options for pressing buttons that do not reside on the cool bar of OneWorld forms:

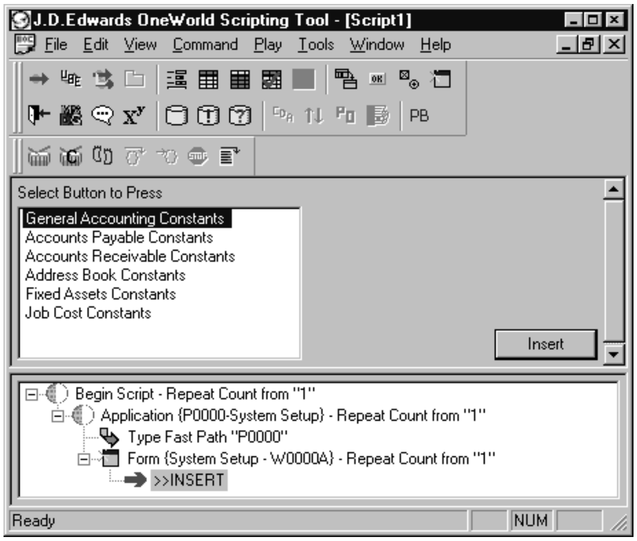
- ❏ Push button options
- ❏ Clickable bitmap options

Push Button Options

OneWorld applications such as System Setup (P0000) contain forms that utilize oversized push buttons. Pushing these buttons allows you to choose forms where you can set up, for example, constants for general accounting, accounts payable, accounts receivable, and so on.


System Setup - [System Setup]
File Edit Preferences Window Help
Can... Dis... Abo
Links Displ... OLE ... Internet
General Accounting Constants
Accounts Payable Constants
Accounts Receivable Constants
Address Book Constants
Fixed Assets Constants
Job Cost Constants

Note that you cannot click these buttons using a Press Toolbar Button command because they do not reside on the toolbar. Therefore, if you click Press Toolbar Button in the OneWorld Scripting Tool command menu, you find that none of these push button options populate the command pane. If you click Press Push Button, on the other hand, the command pane displays in the Select Button to Press list the push button options that reside in the OneWorld form:


J.D.Edwards OneWorld Scripting Tool - [Script1]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

Select Button to Press

General Accounting Constants
Accounts Payable Constants
Accounts Receivable Constants
Address Book Constants
Fixed Assets Constants
Job Cost Constants

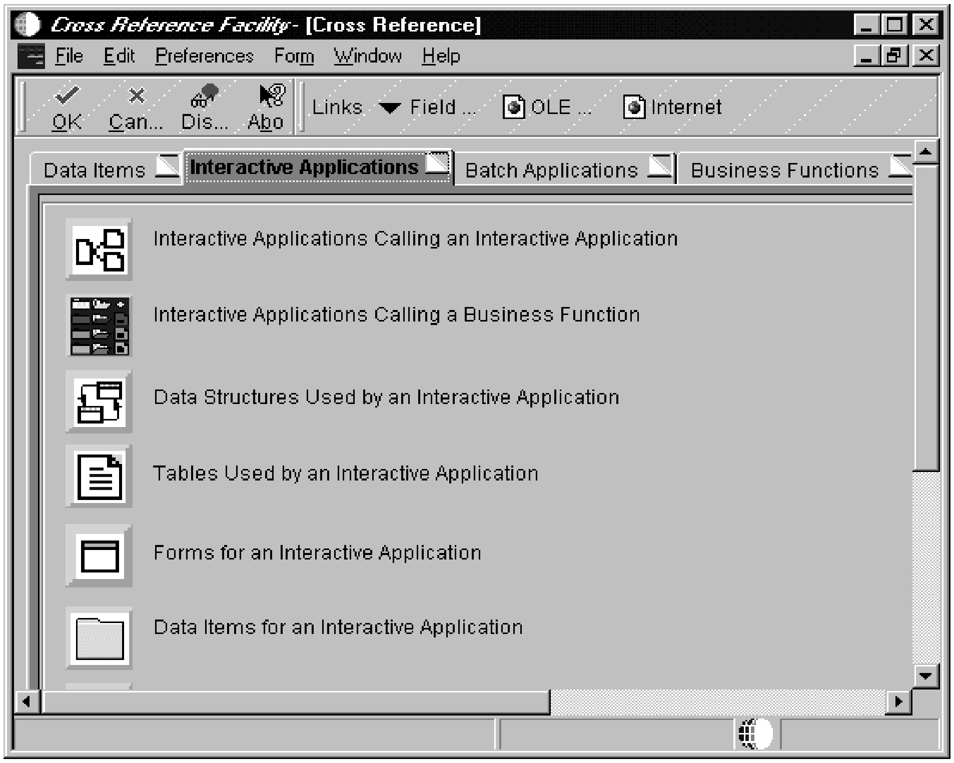
```
Begin Script - Repeat Count from "1"
  Application {P0000-System Setup} - Repeat Count from "1"
    Type Fast Path "P0000"
    Form {System Setup - W0000A} - Repeat Count from "1"
      >>INSERT
```

When you write a command to press a push button, you very often move to a new OneWorld form. Therefore, you often need to write a Form command for the new OneWorld form so that your OneWorld Scripting Tool script matches the actions you have taken in OneWorld. Sometimes by pressing a push button, you exit to a new OneWorld application. In that case, you must write an Application

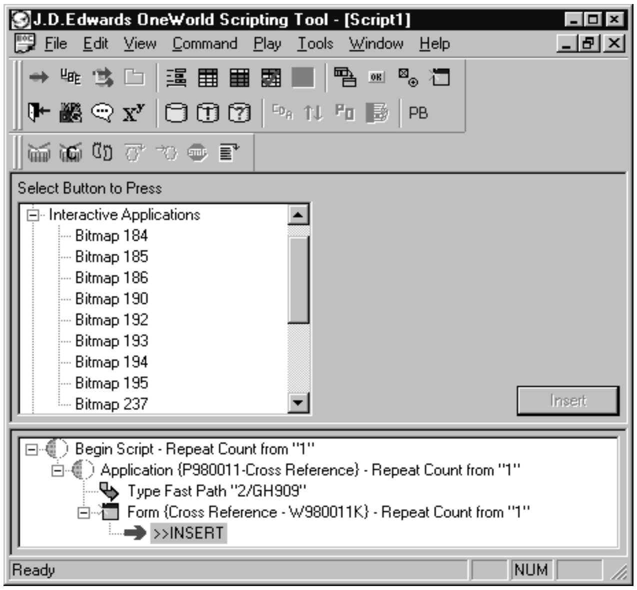
Interconnect command in OneWorld Scripting Tool. Remember that you can check to see if you have exited to a new application by clicking Help and About OneWorld in the cool bar of the active OneWorld form.

Clickable Bitmap Options

OneWorld applications such as Cross Reference (P980011) utilize clickable bitmaps that enable you to move to a new form. Because you cannot click these options by using a button on the OneWorld cool bar, you cannot use a Press Button command to script pressing one of them.


Cross Reference Facility - [Cross Reference]
File Edit Preferences Form Window Help
OK Can... Dis... Abo
Links Field ... OLE ... Internet
Data Items
Interactive Applications
Batch Applications
Business Functions
Interactive Applications Calling an Interactive Application
Interactive Applications Calling a Business Function
Data Structures Used by an Interactive Application
Tables Used by an Interactive Application
Forms for an Interactive Application
Data Items for an Interactive Application

Instead, to click a bitmap option, you again use the OneWorld Scripting Tool Press Push Button command. If you have a OneWorld form active that contains clickable bitmaps, the OneWorld Scripting Tool command pane displays the available bitmaps.


J.D.Edwards OneWorld Scripting Tool - [Script1]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

Select Button to Press

- Interactive Applications
  - Bitmap 184
  - Bitmap 185
  - Bitmap 186
  - Bitmap 190
  - Bitmap 192
  - Bitmap 193
  - Bitmap 194
  - Bitmap 195
  - Bitmap 237

```
Begin Script - Repeat Count from "1"
  Application {P980011-Cross Reference} - Repeat Count from "1"
    Type Fast Path "2/GH909"
    Form {Cross Reference - W980011K} - Repeat Count from "1"
      >>INSERT
```

Note that OneWorld Scripting Tool displays the OneWorld-assigned name, such as Bitmap 184, for each clickable bitmap, rather than the descriptive label that appears next to each one in the OneWorld form. In addition, the control properties for the bitmaps do not contain clues to their identities. These facts might make it important that you take care to identify the particular bitmap you want OneWorld Scripting Tool to click, a task that is made easier by the BlueCue that appears around the OneWorld-assigned name when you click the corresponding name in the OneWorld Scripting Tool command pane.

As was true with push button options, when you script pressing a clickable bitmap, you likely move to another OneWorld form and must write a form command in OneWorld Scripting Tool to match the actions you have taken in OneWorld. If by pressing a clickable bitmap you exit to a new application, you must write an Application Interconnect command in OneWorld Scripting Tool so that the current application in your script matches the application that is active in OneWorld.

Scripting the Press Push Button Command

Writing a Press Push Button command allows your script to perform actions in OneWorld that you cannot script with the Press Button command. You need to script a Press Push Button command when you are working on a script that tests OneWorld applications and forms that contain push buttons and clickable bitmaps that do not reside on the cool bar of the OneWorld form. Writing a command to press a push button or to click a bitmap enables you to move to another form in the application. In some cases this command enables you to exit to another application.

This chapter discusses using the Press Push Button command in OneWorld Scripting Tool to complete the following tasks:

❑ Pressing a push button

❑ Clicking a bitmap

Pressing a Push Button

You might want to test a OneWorld application, such as System Setup (P0000), which contains forms that display oversized push buttons that do not reside on the OneWorld cool bar. In most cases, you push these buttons to move to a new form in the same application or to exit to a new application.

Scripting a Press Push Button command in OneWorld Scripting Tool enables you to push the button in the OneWorld form. Remember that you cannot write a command to perform this action by writing a Press Button command because the Press Button command applies only to buttons on the OneWorld toolbar.

▶ To press a push button in a OneWorld form

1. In the command menu of the OneWorld Scripting Tool form, click Press Push Button.

   OneWorld Scripting Tool populates the command pane with push button choices only when the active OneWorld form contains push buttons that you cannot access from the OneWorld cool bar.

2. In the OneWorld Scripting Tool command pane, choose a push button from the Select Button to Press list.

3. Click the Insert button.
4. If pressing the push button results in a move to another form in the same application, click Form in the OneWorld Scripting Tool command menu.
5. In the Form list of the OneWorld Scripting Tool command pane, choose the name of the form that is active in OneWorld.
6. Click the Insert button.
7. If pressing the push button results in OneWorld exiting to a new application, click Application Interconnect in the OneWorld Scripting Tool command menu.
8. In the command pane, choose from the lists that appear:
   - Application (choose the application that is active in OneWorld)
   - Menu (choose the form that is active in OneWorld)
9. Click the Insert button.

Clicking a Bitmap

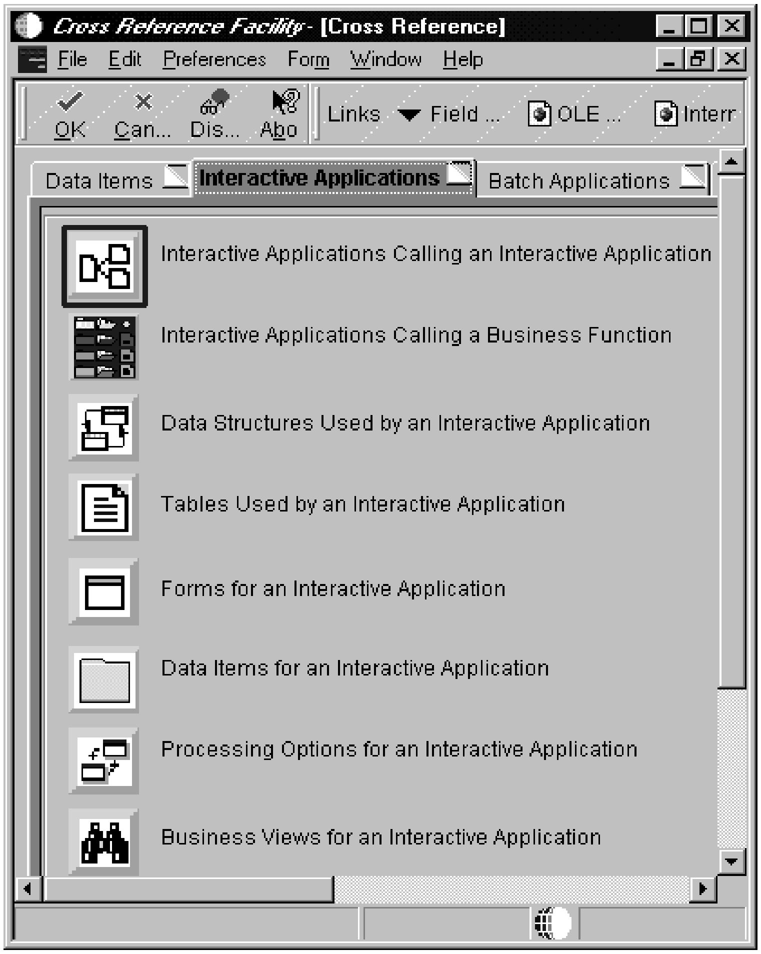
Some OneWorld applications contain forms that use clickable bitmaps that do not reside on the OneWorld cool bar. If you are using OneWorld Scripting Tool to test one of these applications, you must script a Press Push Button command if you want to click one of the bitmaps. Remember that OneWorld Scripting Tool displays in the command pane the OneWorld-assigned name for each bitmap, rather than the descriptive text that appears next to each bitmap in OneWorld. If you click the OneWorld-assigned name in the command pane, OneWorld Scripting Tool helps to identify the corresponding clickable bitmap in the OneWorld form by enclosing it in a BlueCue.

Scripting a Press Push Button command to click a bitmap enables you to move to another OneWorld form or to exit to another application. You cannot click the bitmap by writing a Press Button command because the bitmaps do not reside on the OneWorld cool bar.

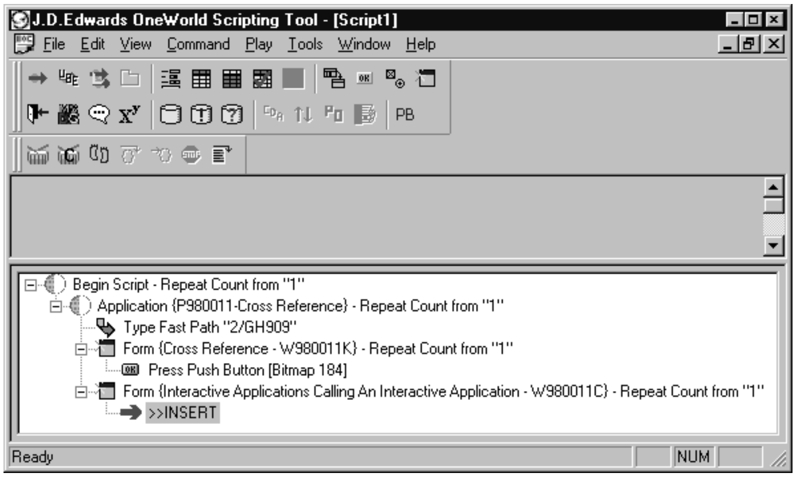
▶ To click a bitmap in a OneWorld form

1. In the command menu of the OneWorld Scripting Tool form, click Press Push Button.
2. In the OneWorld Scripting Tool command pane, choose from the Select Button to Press list the name of a tab in the OneWorld form.
3. Click the node next to the tab name.
4. From the drop-down menu, choose the OneWorld-assigned name of a clickable bitmap.

   Note: When you click on the bitmap name, the BlueCue that appears in the OneWorld form identifies the corresponding bitmap.


Cross Reference Facility - [Cross Reference]
File Edit Preferences Form Window Help
OK Can... Dis... Abo
Links Field ... OLE ... Interr
Data Items
Interactive Applications
Batch Applications
Interactive Applications Calling an Interactive Application
Interactive Applications Calling a Business Function
Data Structures Used by an Interactive Application
Tables Used by an Interactive Application
Forms for an Interactive Application
Data Items for an Interactive Application
Processing Options for an Interactive Application
Business Views for an Interactive Application

5. Click the Insert button.
6. If pressing the push button results in OneWorld moving to another form in the same application, click Form in the OneWorld Scripting Tool command menu.
7. In the Form list of the OneWorld Scripting Tool command pane, choose the name of the form that is active in OneWorld.
8. Click the Insert button.



```
Begin Script - Repeat Count from "1"
  Application {P980011-Cross Reference} - Repeat Count from "1"
    Type Fast Path "2/GH909"
    Form {Cross Reference - W980011K} - Repeat Count from "1"
      Press Push Button [Bitmap 184]
    Form {Interactive Applications Calling An Interactive Application - W980011C} - Repeat Count from "1"
      >>INSERT
```

9. If pressing the push button results in OneWorld exiting to a new application, click Application Interconnect in the OneWorld Scripting Tool command menu.

10. In the command pane, choose from the lists that appear

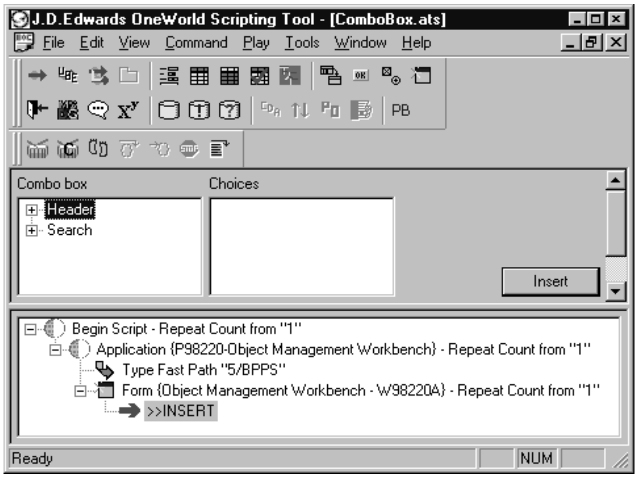
    - Application (choose the application that is active in OneWorld)
    - Menu (choose the form that is active in OneWorld)

11. Click the Insert button.

The Select Item in Combo Box Command

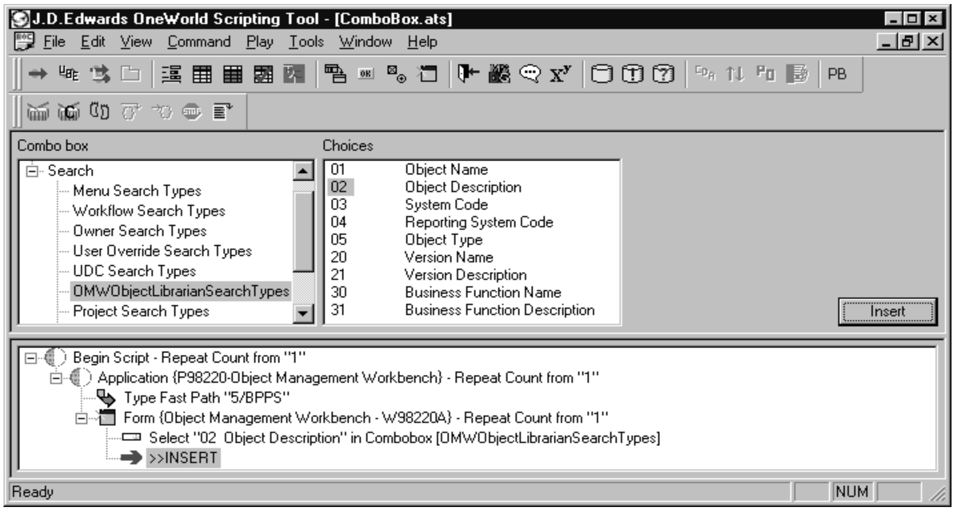
Some OneWorld applications, such as Object Management Workbench (P98220) and Expense Report (P20120), utilize combo box controls. These controls can appear in forms as edit text fields, pop-up menus, or scrolling lists. OneWorld Scripting Tool provides the text from the combo box in the command pane, which allows you to script choosing items in a combo box. After you write a Select Item in Combo Box command and play back the script, OneWorld Scripting Tool locates the combo box and sends a message to the OneWorld form to select the text string that you specified in the command pane.

You choose the Select Item in Combo Box command from the Command menu only; there is no cool bar button for the command. When you choose the command, OneWorld Scripting Tool populates the Combo Box list in the command pane with either a list or a tree control. A tree control appears in the command pane only when the combo box is under tab controls in the OneWorld form that is active.


J.D.Edwards OneWorld Scripting Tool - [ComboBox.ats]
File Edit View Command Play Tools Window Help
PB
Combo box
Header
Search
Choices
Insert
Ready
NUM

```
Begin Script - Repeat Count from "1"
  Application {P98220-Object Management Workbench} - Repeat Count from "1"
    Type Fast Path "5/BPPS"
    Form {Object Management Workbench - W98220A} - Repeat Count from "1"
      >>INSERT
```

When you click in the Combo Box list the name of a control or of an item in a scrolling list or pop-up menu, OneWorld Scripting Tool populates the Choices list with the text names contained in the combo box, along with the user defined system codes, which it retrieves from the OneWorld UDC table (F0005).


J.D.Edwards OneWorld Scripting Tool - [ComboBox.ats]
File Edit View Command Play Tools Window Help
PB
Combo box
Choices
Insert
Ready
NUM

- Search
  - Menu Search Types
  - Workflow Search Types
  - Owner Search Types
  - User Override Search Types
  - UDC Search Types
  - OMWObjectLibrarianSearchTypes
  - Project Search Types

01	Object Name
02	Object Description
03	System Code
04	Reporting System Code
05	Object Type
20	Version Name
21	Version Description
30	Business Function Name
31	Business Function Description

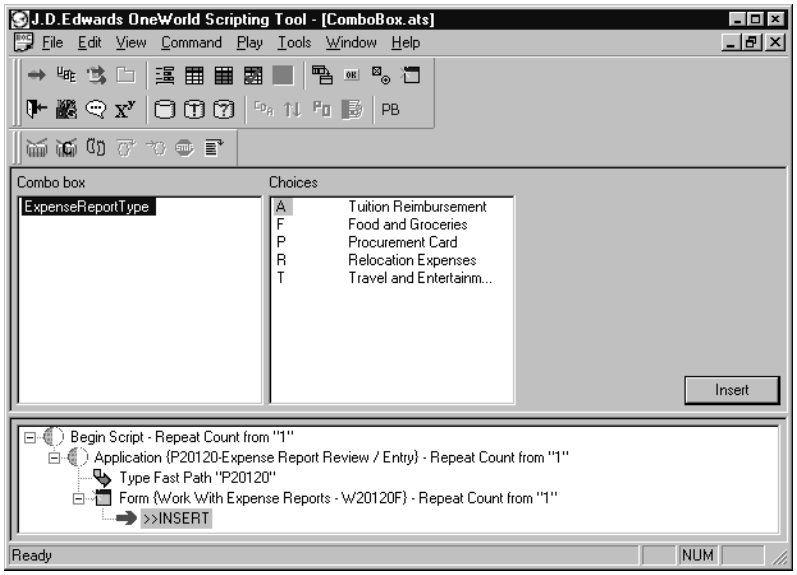
- Begin Script - Repeat Count from "1"
  - Application {P98220-Object Management Workbench} - Repeat Count from "1"
    - Type Fast Path "5/BPPS"
    - Form {Object Management Workbench - W98220A} - Repeat Count from "1"
      - Select "02 Object Description" in Combobox [OMWObjectLibrarianSearchTypes]
      - >>INSERT

You can add as many combo box commands as you need to complete your script. Forms in some OneWorld applications have hidden combo box controls that are not used. Nonetheless, OneWorld Scripting Tool displays these controls in the Choices list, just as it displays hidden header controls and grid columns in the command pane. You cannot choose a default entry to the combo box, such as None.

Scripting the Select Item in Combo Box Command

OneWorld utilizes combo boxes in form header controls and under form tabs. When you write scripts that use forms that contain combo box lists, you choose the Select Item in Combo Box command from the command menu of the OneWorld Scripting Tool form and choose a combo box from the Combo Box list in the command pane.

After you make a choice from the Combo Box list, OneWorld Scripting Tool populates the Choices list with the available items in the combo box.

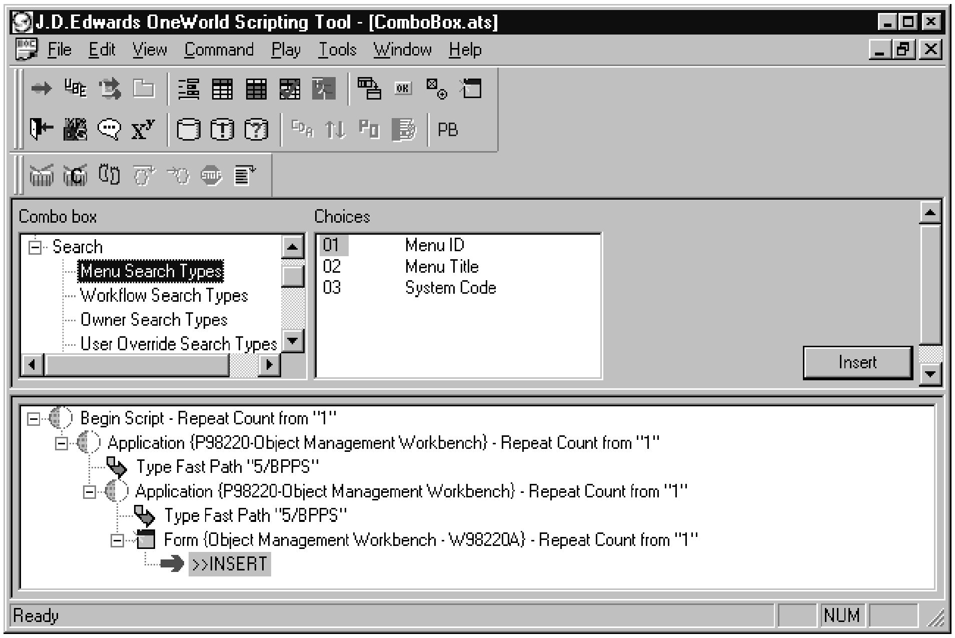

J.D.Edwards OneWorld Scripting Tool - [ComboBox.ats]
File Edit View Command Play Tools Window Help
PB
Combo box
Choices
Insert
Ready
NUM

ExpenseReportType

A	Tuition Reimbursement
F	Food and Groceries
P	Procurement Card
R	Relocation Expenses
T	Travel and Entertainm...

```
Begin Script - Repeat Count from "1"
    Application {P20120-Expense Report Review / Entry} - Repeat Count from "1"
        Type Fast Path "P20120"
        Form {Work With Expense Reports - W20120F} - Repeat Count from "1"
            >>INSERT
```

Some applications, such as Object Management Workbench (P98220) use more than one combo box list, and the lists are dependent on one another to establish, for example, search criteria.

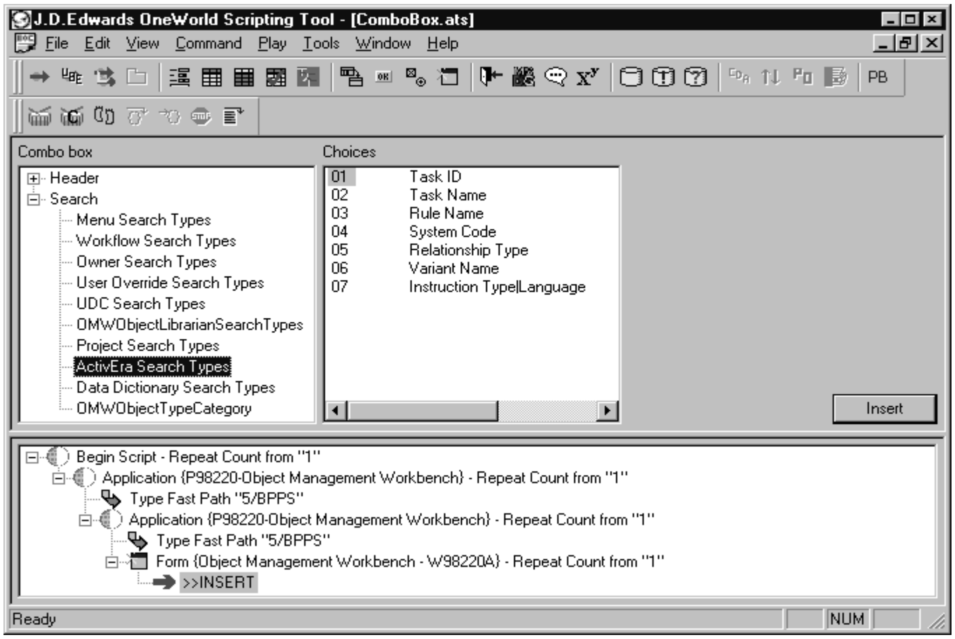

J.D.Edwards OneWorld Scripting Tool - [ComboBox.ats]
File Edit View Command Play Tools Window Help
PB
Combo box
Choices
Insert
Ready
NUM

Search
- Menu Search Types
- Workflow Search Types
- Owner Search Types
- User Override Search Types

01	Menu ID
02	Menu Title
03	System Code

```
Begin Script - Repeat Count from "1"
  Application {P98220-Object Management Workbench} - Repeat Count from "1"
    Type Fast Path "5/BPPS"
    Application {P98220-Object Management Workbench} - Repeat Count from "1"
      Type Fast Path "5/BPPS"
      Form {Object Management Workbench - W98220A} - Repeat Count from "1"
        >>INSERT
```

In this case, the choice that you make in the Combo Box list in the OneWorld Scripting Tool command pane changes the items in the Choices list.


J.D.Edwards OneWorld Scripting Tool - [ComboBox.ats]
File Edit View Command Play Tools Window Help
PB
Combo box
Choices
Insert
Ready
NUM

Header

Search
- Menu Search Types
- Workflow Search Types
- Owner Search Types
- User Override Search Types
- UDC Search Types
- OMWObjectLibrarianSearchTypes
- Project Search Types
- ActivEra Search Types
- Data Dictionary Search Types
- OMWObjectTypeCategory

01	Task ID
02	Task Name
03	Rule Name
04	System Code
05	Relationship Type
06	Variant Name
07	Instruction Type|Language

```
Begin Script - Repeat Count from "1"
  Application {P98220-Object Management Workbench} - Repeat Count from "1"
    Type Fast Path "5/BPPS"
    Application {P98220-Object Management Workbench} - Repeat Count from "1"
      Type Fast Path "5/BPPS"
      Form {Object Management Workbench - W98220A} - Repeat Count from "1"
        >>INSERT
```

Before You Begin

- ❑ Select Item in Combo Box is active in the command menu of the OneWorld Scripting Tool form only if you launch a OneWorld application and form that uses combo boxes.

To access English documentation updates, see https://knowledge.jdedwards.com/JDEContent/documentationcbt/overview/about_documentation_updates.pdf

▶ To script the Select Item in Combo Box command

1. In the Command menu of the OneWorld Scripting Tool form, choose Application.
2. In the command pane, choose an application and Fast Path and click the Insert button.
3. Choose Select Item in Combo Box from the command menu.
4. In the command pane of the OneWorld Scripting Tool form, choose a combo box from the Combo Box list.

   OneWorld Scripting Tool populates the Choices list with the items in the combo box.

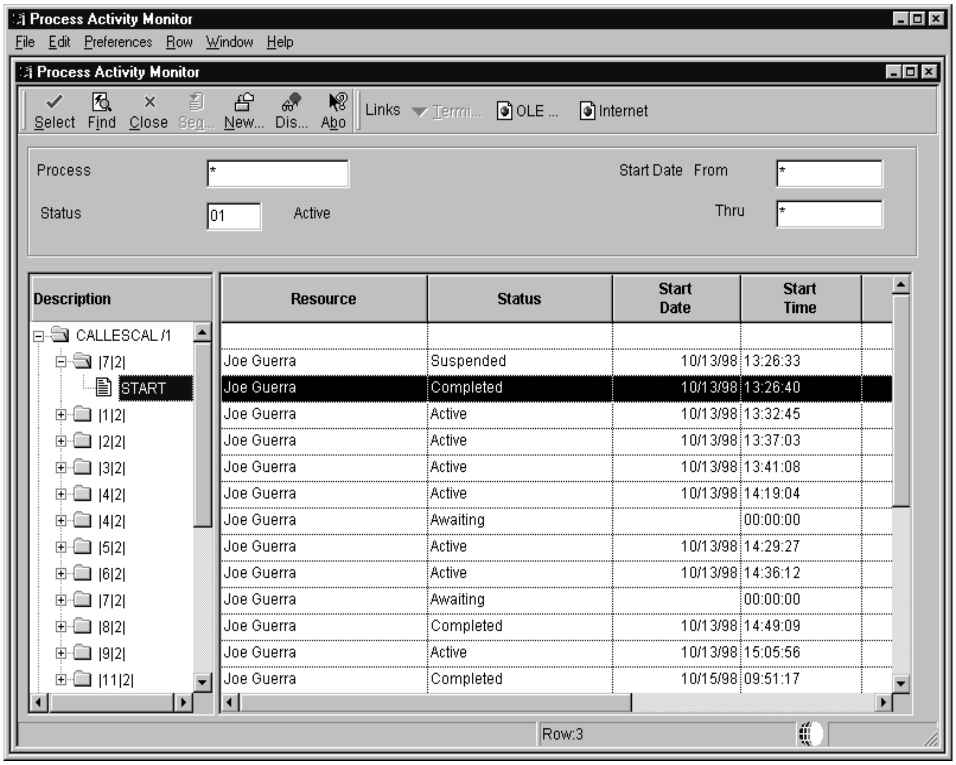
5. Choose a combo box item from the Choices list and click the Insert button.

   OneWorld Scripting Tool enters the item from the combo box list to the control in the OneWorld form.

The Build Tree Path Command

Some OneWorld applications use tree path controls rather than header controls and grid detail areas. To write script commands for forms in these applications, you must use the Build Tree Path Control command to create a unique path that traverses the tree path in a OneWorld form.

You use the Build Tree Path Control command using any combination of literal text or variables. For example, the first node in a tree might consist of a parent, one child, and one grandchild.



Process *

Status 01 Active

Start Date From *

Thru *

Description

CALLESCAL /1
|7|2|
START
1	2
2	2
3	2
4	2
4	2
5	2
6	2
7	2
8	2
9	2
11	2

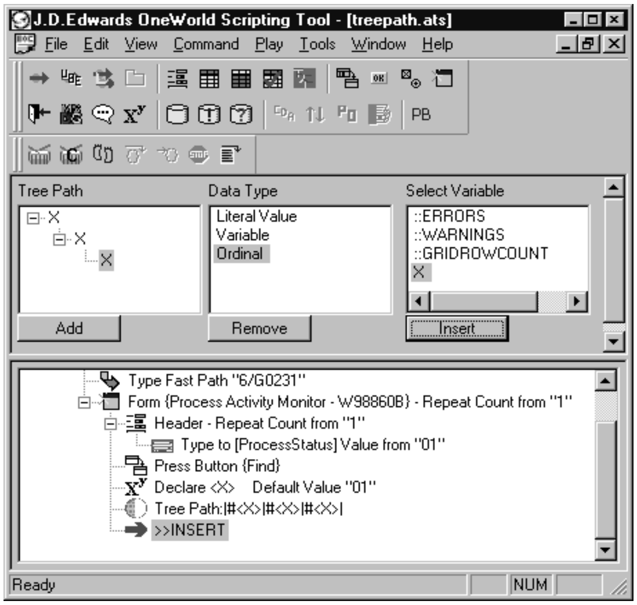
Resource	Status	Start Date	Start Time
Joe Guerra	Suspended	10/13/98	13:26:33
Joe Guerra	Completed	10/13/98	13:26:40
Joe Guerra	Active	10/13/98	13:32:45
Joe Guerra	Active	10/13/98	13:37:03
Joe Guerra	Active	10/13/98	13:41:08
Joe Guerra	Active	10/13/98	14:19:04
Joe Guerra	Awaiting		00:00:00
Joe Guerra	Active	10/13/98	14:29:27
Joe Guerra	Active	10/13/98	14:36:12
Joe Guerra	Awaiting		00:00:00
Joe Guerra	Completed	10/13/98	14:49:09
Joe Guerra	Active	10/13/98	15:05:56
Joe Guerra	Completed	10/15/98	09:51:17

To use OneWorld Scripting Tool to write the Build Tree Path Control command, you first designate the data type that represents the first node in the tree path. Available data types are:

- Literal values, which are the precise text that designates a node in a tree control
- Variable values, which you set as the text that designates a node in a tree control

- Ordinal values, which represent the order that a node is placed in a tree path: first, second, third, and so on

In this example, you choose ordinal as the data type: you want to build a path to the first node in the tree (the parent), the first child of the parent, and the first grandchild of the parent. Clicking the Add button populates the tree path list with a leaf node: one without children. You can create parent-child relationships by clicking the Add button to add nodes.



Tree Path
X
X
X

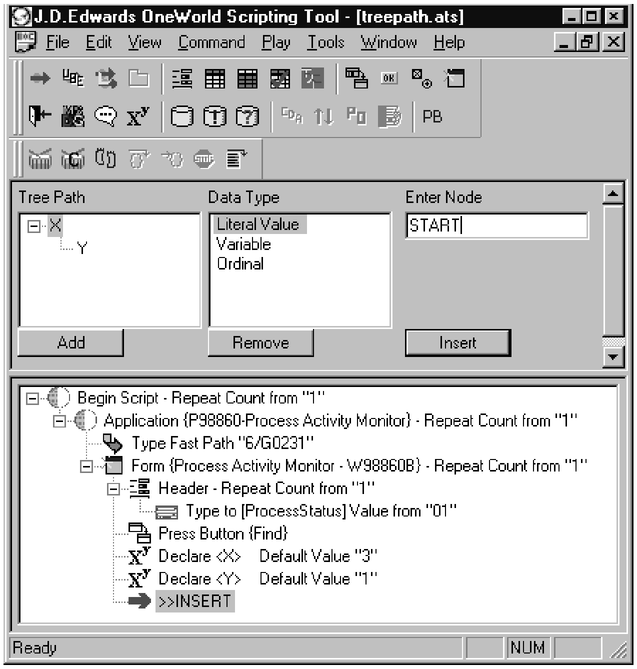
Data Type
Literal Value
Variable
Ordinal

Select Variable
::ERRORS
::WARNINGS
::GRIDROWCOUNT
X

```
Type Fast Path "6/G0231"
Form {Process Activity Monitor - W98860B} - Repeat Count from "1"
  Header - Repeat Count from "1"
    Type to [ProcessStatus] Value from "01"
  Press Button {Find}
  Declare <X>    Default Value "01"
  Tree Path:|#<X>|#<X>|#<X>|
  >>INSERT
```

During playback, OneWorld Scripting Tool uses the search string you composed to identify the coordinates of each node in the tree in the OneWorld form that is active.

You can modify the tree path as necessary using the Add and Remove buttons. Removing the parent node also removes all children from the path. If you want to add a node, you must add it to a leaf node.


J.D.Edwards OneWorld Scripting Tool - [treepath.ats]
File Edit View Command Play Tools Window Help
PB
Add
Remove
Insert
Ready
NUM

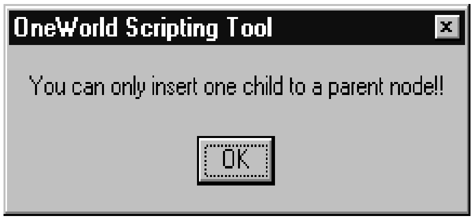
Tree Path
X
Y

Data Type
Literal Value
Variable
Ordinal

Enter Node
START

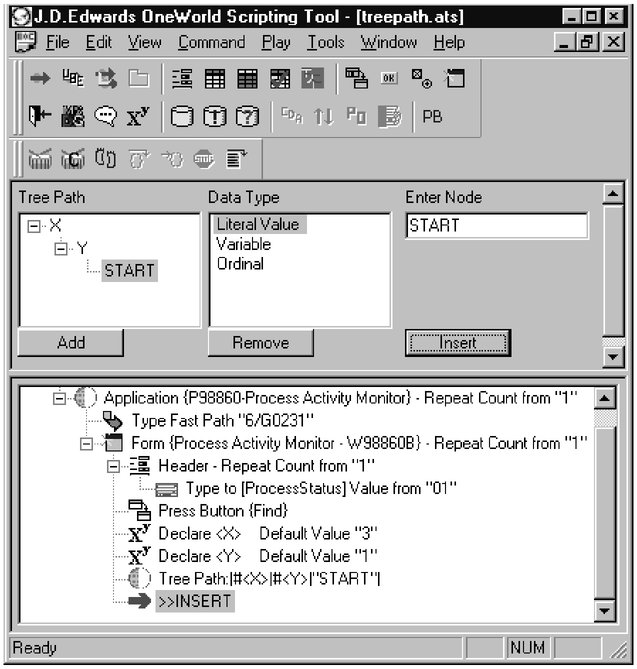
```
Begin Script - Repeat Count from "1"
  Application {P98860-Process Activity Monitor} - Repeat Count from "1"
    Type Fast Path "6/G0231"
    Form {Process Activity Monitor - W98860B} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [ProcessStatus] Value from "01"
      Press Button {Find}
      Declare <X>    Default Value "3"
      Declare <Y>    Default Value "1"
      >>INSERT
```

If you attempt to add a child node to a parent that already has a child, OneWorld Scripting Tool displays a dialogue box notifying you that cannot add a child to the node.



You can only insert one child to a parent node!!

You must click a leaf node to create a new node with a child.



J.D.Edwards OneWorld Scripting Tool - [treepath.ats]

Tree Path:
- X
  - Y
    - START

Data Type: Literal Value, Variable, Ordinal

Enter Node: START

```
Application {P98860-Process Activity Monitor} - Repeat Count from "1"
    Type Fast Path "6/G0231"
    Form {Process Activity Monitor - W98860B} - Repeat Count from "1"
        Header - Repeat Count from "1"
            Type to [ProcessStatus] Value from "01"
        Press Button {Find}
        Declare <X>    Default Value "3"
        Declare <Y>    Default Value "1"
        Tree Path:|#<X>|#<Y>|"START"|
        >>INSERT
```

Ready NUM

Scripting the Build Tree Path Command

You use the Build Tree Path command to write scripts that test OneWorld applications that use tree controls rather than header controls and grid detail areas. You can use any combination of variables or literal text to build a unique path to nodes that exist in the tree path in an active OneWorld form. You can also modify the tree path by adding and removing nodes from the path that you build.

This chapter discusses the steps you take to run the following tree path tasks:

- ❑ Building a tree path using variable values
- ❑ Building a tree path using literal values
- ❑ Adding a parent node or child to a tree path
- ❑ Removing a parent node or a child from a tree path

Building a Tree Path Using Variable Values

There are two ways to use variable values to build a tree path. You can declare a variable and set its value as the text that represents a node in the tree. In this case, you choose Variable as the data type for the node. You can also set a numeric value for the variable. If you set the value as 3, OneWorld Scripting Tool chooses the third node in the tree or the third child of a parent. In this case, you choose Ordinal as the data type for the node.

Whether you choose Variable or Ordinal as the data type for the node, the names of the variables you declared appear in the Select Variable list of the command pane.

Before You Begin

- ❑ Declare a variable and set its value. See *Using a Variable as a Source of Input.*

▶ To build a tree path using variable values

1. In the OneWorld Scripting Tool form, launch an application and form that utilize a tree control.

2. In the Command menu, click Variable.
3. Declare a variable and set its value.

   Note: You can assign any value to the variable. However, remember that if you assign a numeric value, that value represents the position of a parent node or a child in the tree path. For example, if you set the variable's value as 3, the value represents the third node in the OneWorld tree control.

4. With a form that utilizes a tree control active in OneWorld, click Build Tree Path in the command menu.

   The OneWorld Scripting Tool command pane contains a Tree Path list and a Data Type list.

5. Choose either Variable or Ordinal from the Data Type list.
   - Choose Variable if the variable you want to use has a text value
   - Choose Ordinal if the variable you want to use has a numeric value
6. Choose from the Select Variable list a variable whose value you have set.
7. Click the Add button.

   OneWorld Scripting Tool inserts the variable as a node in the Tree Path list.

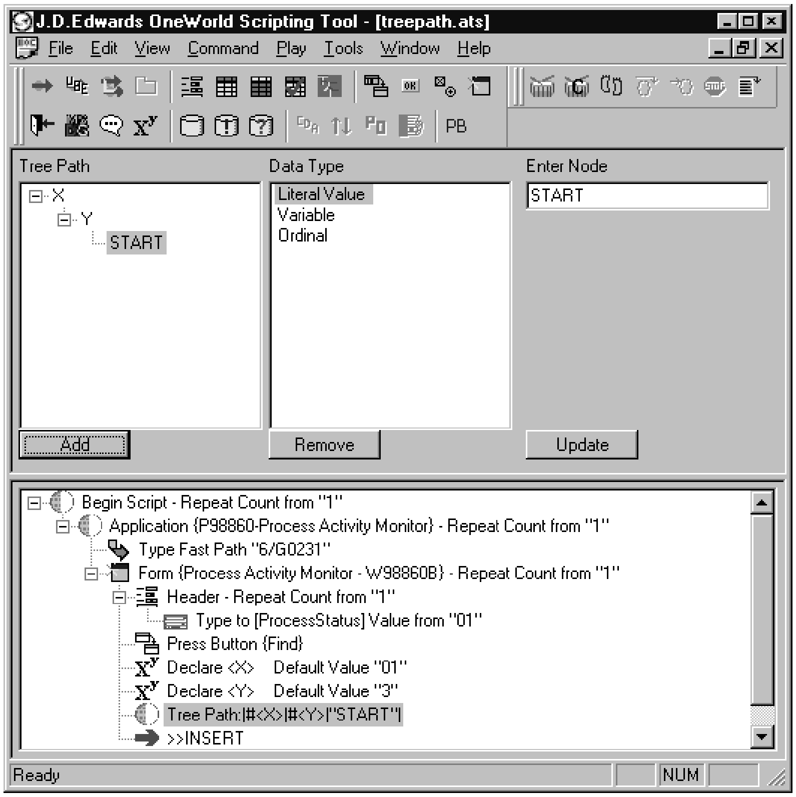
Building a Tree Path Using Literal Values

You can also build a tree path using a literal value as the data type to represent a node. You type the name of the node in the Enter Node list exactly as it appears in a OneWorld form. Remember that you can use literal values as a data type in combination with variable and ordinal values to create the tree path.

▶ To build a tree path using literal values

1. In the OneWorld Scripting Tool form, launch an application and form that utilize tree controls.
2. In the command menu, click Build Tree Path.
3. In the Data Type list, click Literal Value.
4. In the Enter Node list, type a literal value.
5. Click the Add button.

   OneWorld Scripting Tool inserts the literal value as a node in the Tree Path list.


J.D.Edwards OneWorld Scripting Tool - [treepath.ats]
File Edit View Command Play Tools Window Help
PB
Add
Remove
Update
Ready
NUM

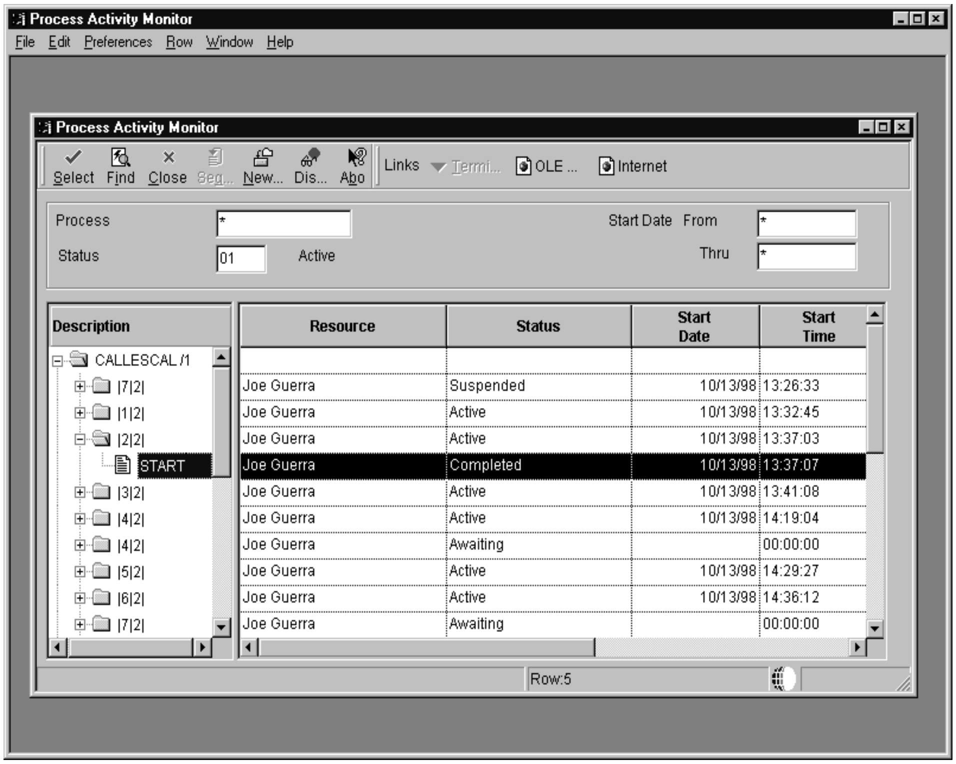
Tree Path
- X
  - Y
    - START

Data Type
- Literal Value
- Variable
- Ordinal

Enter Node: START

```
Begin Script - Repeat Count from "1"
  Application {P98860-Process Activity Monitor} - Repeat Count from "1"
    Type Fast Path "6/G0231"
    Form {Process Activity Monitor - W98860B} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [ProcessStatus] Value from "01"
      Press Button {Find}
      Declare <X>    Default Value "01"
      Declare <Y>    Default Value "3"
      Tree Path:|#<X>|#<Y>|"START"|
      >>INSERT
```

When you play back the script, OneWorld Scripting Tool finds the node with the literal value based on the tree path you built.


Process Activity Monitor
File Edit Preferences Row Window Help
Process Activity Monitor
Select Find Close Seq... New... Dis... Abo
Links Termi... OLE ... Internet
Row:5

Process: *
Status: 01 Active
Start Date From: *
Thru: *

Description
- CALLESCAL /1
  - |7|2|
  - |1|2|
  - |2|2|
    - START
  - |3|2|
  - |4|2|
  - |4|2|
  - |5|2|
  - |6|2|
  - |7|2|

Resource	Status	Start Date	Start Time
Joe Guerra	Suspended	10/13/98	13:26:33
Joe Guerra	Active	10/13/98	13:32:45
Joe Guerra	Active	10/13/98	13:37:03
Joe Guerra	Completed	10/13/98	13:37:07
Joe Guerra	Active	10/13/98	13:41:08
Joe Guerra	Active	10/13/98	14:19:04
Joe Guerra	Awaiting		00:00:00
Joe Guerra	Active	10/13/98	14:29:27
Joe Guerra	Active	10/13/98	14:36:12
Joe Guerra	Awaiting		00:00:00

Adding a Parent Node or Child to a Tree Path

After you have chosen a data type and entered a literal value or selected a variable, you add a tree path node by:

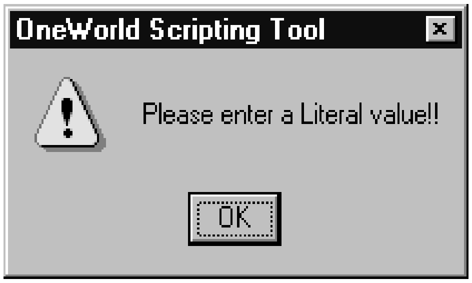
- Clicking the Add button with the Tree Path list in the command pane unpopulated
- Choosing a node in the tree path list that does not have a child and clicking the Add button.

Remember that you can not add to a node that already has a child.

▶ To add a parent node or child to a tree path

1. In the Tree Path list of the OneWorld Scripting Tool form command pane, click a node that does not have a child.
2. In the Data Type list, click Literal Value, Variable, or Ordinal.
3. Choose a variable from the Value Selection list or enter a literal value to the Enter Node list.
4. Click the Add button.

   Note: If you click the Add button without entering a literal value, choosing a variable, or choosing an ordinal, a form appears, asking you to enter or choose a value.



OneWorld Scripting Tool

Please enter a Literal value!!

Removing a Parent Node or a Child from a Tree Path

You can modify your tree path simply by selecting a node or a child and clicking the Remove button. Remember that removing a node also removes the child of the parent.

▶ To remove a parent node or a child from a tree path

1. In the Tree Path list of the OneWorld Scripting Tool form command pane, click a parent node or a child.
2. To remove a parent node and its child, click the parent node and the Remove button.
3. To remove the child only, click the child and the Remove button.

The Database Validation Command

To ensure that the data you have entered to the OneWorld database through commands in your OneWorld Scripting Tool script, you use the data validation action commands.

OneWorld Scripting Tool's data validation process consists of three stages:

- Declaration, in which you give the validation a name.
- Association, in which you pair the values that you enter to header controls or grid columns with columns in a database table.
- Execution, in which you use record selection criteria to check the database to see if records you entered during scripting were, in fact, entered to the OneWorld database.

You can also verify that OneWorld has deleted records from the database.

This chapter covers the essential components of database validation:

❑ Validation definition

❑ Validation declaration

❑ Validation association

❑ Validation execution

❑ Expect No Matching Records option

Validation Definition

You script validation commands to compare a data set you created in OneWorld Scripting Tool and ran in OneWorld with a data set written to the database. These commands confirm that OneWorld has entered records to the database in the form that you expected. You also use validation for process testing, to make sure that data have moved as you intended through a sequence of applications related to a transaction cycle in OneWorld. Finally, you use validation to validate values that can't be accessed through OneWorld Scripting Tool, such as century. In most transaction fields, you make no entry for century. Using validation commands, you can check to make sure that century data are actually getting to the database.

Validation Declaration

You script a new database validation command each time you declare a validation. You can declare a validation at any point in the script.

The declared validation includes:

- A validation name of your choosing
- A table to be validated, which you choose from a list

Declaring the validation is much like declaring a variable in that it provides only a name for the validation. In essence, it provides the room in which you later store values.

Validation Association

Associating a variable enables you to store data. You can gather these data from many different points in the script. No action is taken when you associate a validation. Rather, you open the room that you created when you declared the validation and store in that room data of your choosing.

In associating a validation, you define the values to be validated against chosen columns in the database. You choose a column, which cannot be associated more than once in a script, such as ABALPH (NameAlpha in Address Book Revisions) and associate that column with a value, which can be derived from a literal value, variable, header control, or grid column. You then choose a database value type, against which OneWorld Scripting Tool validates the data you enter to the database when you run your script.

In summary, you must specify four items in order to script the validation association:

- Validation name (declared validation)
- Database column identification, chosen from a list
- Source of expected data, such as a header control
- Database value type, either key selection or validation value

Validation association can occur at as many different points in the script as you desire, but you must use both a key selection value and a validation value.

This topic discusses the distinction between the two types of database values you use in associating a validation:

❑ Key selection value

❑ Validation value

Key Selection Value

The key selection value specifies the database column that contains the specific records that you want. When you mark a validation association as a key selection value, a database record matching an associated column must exist or the validation fails. When OneWorld Scripting Tool runs the validation, it uses the key selection value to check to see if OneWorld has written to the correct database column the values that you stored during association.

In choosing a key selection value, you select a value that all records that you are validating have in common. For example, in performing an Address Book Revision validation, you might choose the database column A5AN8 or ABAN8 as a key selection value because all customers have an Address Book Number.

Validation Value

When you mark a validation association value, you again choose a specific record set to be validated. However, while the key selection value tells the database where to go to find the records that are to be validated, the validation value specifies what values, such as names, you expect to find in the database.

Validation Execution

You run the validation using the record selection criteria you established in validation association. OneWorld Scripting Tool retrieves the specified data from the database through structured query language (SQL). You can then compare the retrieved data with the data you expected to return. Running the validation answers the question: Did the data that I entered and stored during association make it to the database in the condition that I specified?

Three actions occur when you script running the declared and associated validation:

- Generation of an SQL statement
- Query of the database for the specified data
- Comparison of the returned data to the expected data

The generated SQL statement contains the table that was chosen when the variable was declared, the validation value columns chosen during association, and the key selection chosen during association.

Thus, an SQL statement generated in running the validation might contain the identity of the table (F0101) that contains the expected data, the columns (ABALPH, ABAT1 and ABAN8) that contain the expected data, and the column (ABAN8) that is the key that unites all the columns.

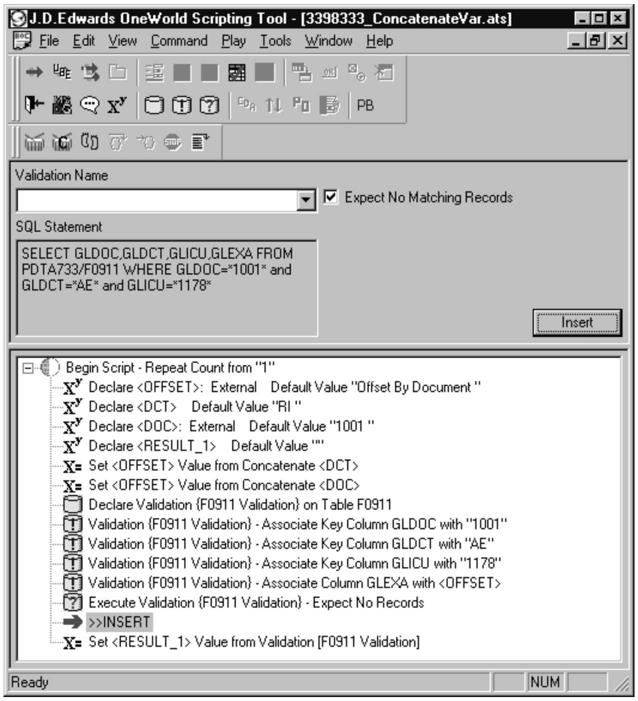
The SQL statement queries the database and retrieves data that conform to the statement elements. You compare the results of that query against the results that you expected based on the data that you stored during validation association.

If you have made an error in associating the data to be validated, OneWorld Scripting Tool presents an error message. For example, if you do not choose a key column during association, OneWorld Scripting Tool notifies you that no record selection criteria have been chosen. If you associate the key column with an incorrect header control, grid column or variable, OneWorld Scripting Tool notifies you that the SQL statement contains an error.

In either of these cases, you can go back in the script and make necessary corrections. In addition, running a validation that fails does not stop the script from playing through to completion. The test results compare the data you expected to return with the data that were returned.

Expect No Matching Records Option

If you enter a record, successfully validate it, then write a command to delete the record, you might want to validate that OneWorld successfully deleted the record from the database. To do so, you run the validation again and use the Expect No Matching Records Checkbox option. When you run the script, OneWorld Scripting Tool again checks the database. The validation runs successfully if OneWorld Scripting Tool finds that the record that you deleted from the database no longer exists. Put another way, in choosing this option you tell OneWorld Scripting Tool to expect to find no records that match the criteria in the SQL statement.



J.D.Edwards OneWorld Scripting Tool - [3398333_ConcatenateVar.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

Validation Name

☑ Expect No Matching Records

SQL Statement

```
SELECT GLDOC,GLDCT,GLICU,GLEXA FROM
PDTA733/F0911 WHERE GLDOC="1001" and
GLDCT="AE" and GLICU="1178"
```

```
Begin Script - Repeat Count from "1"
    Declare <OFFSET>:  External   Default Value "Offset By Document "
    Declare <DCT>   Default Value "RI "
    Declare <DOC>:  External   Default Value "1001 "
    Declare <RESULT_1>    Default Value ""
    Set <OFFSET> Value from Concatenate <DCT>
    Set <OFFSET> Value from Concatenate <DOC>
    Declare Validation {F0911 Validation} on Table F0911
    Validation {F0911 Validation} - Associate Key Column GLDOC with "1001"
    Validation {F0911 Validation} - Associate Key Column GLDCT with "AE"
    Validation {F0911 Validation} - Associate Key Column GLICU with "1178"
    Validation {F0911 Validation} - Associate Column GLEXA with <OFFSET>
    Execute Validation {F0911 Validation} - Expect No Records
    >>INSERT
    Set <RESULT_1> Value from Validation [F0911 Validation]
```

Scripting the Database Validation Command

You enter database validation action commands as you write your script. There is no formula that determines where the commands must occur. However, each of the three phases, declaration, assignment, and execution, must occur for the validation to take place.

This chapter covers the following tasks:

- ❑ Declaring a validation
- ❑ Associating a validation
- ❑ Executing a validation

Declaring a Validation

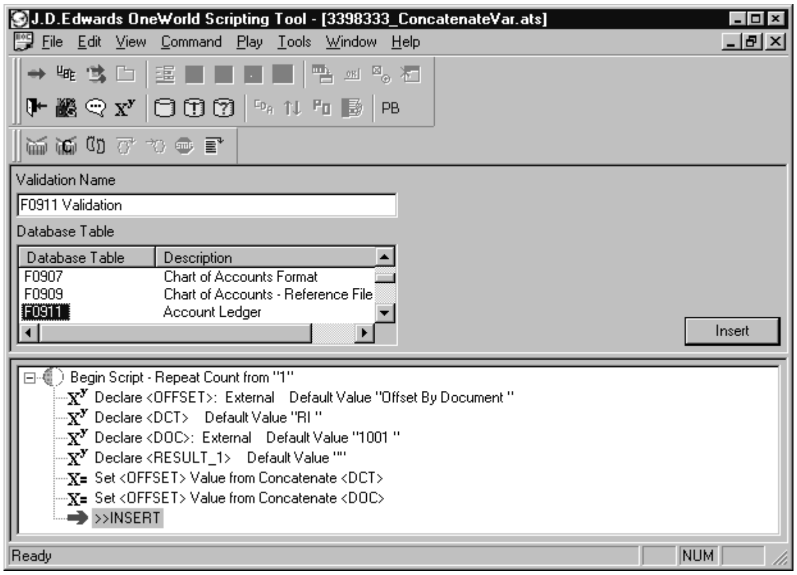
You can declare one or more validations as soon as you begin a script. You do not have to place the validation declaration command line at the top of the script if you want the validation to be effective within any node in the script, as you do when you declare a variable. However, declaring the validation early enables you to easily store data through association as you write the script.

▶ To declare a validation

1. In the command menu of the OneWorld Scripting Tool form, click Declare New Validation.
2. In the command pane, assign a name to the validation by typing one in the Validation Name list.
3. Choose a selection in the Database Table list.

   This identifies the database table against which you validate data from OneWorld Scripting Tool and OneWorld that you store during association.

4. Click the Insert button.



J.D.Edwards OneWorld Scripting Tool - [3398333_ConcatenateVar.ats]

Validation Name: F0911 Validation

Database Table

Database Table	Description
F0907	Chart of Accounts Format
F0909	Chart of Accounts - Reference File
F0911	Account Ledger

```
Begin Script - Repeat Count from "1"
    Declare <OFFSET>:  External    Default Value "Offset By Document "
    Declare <DCT>    Default Value "RI "
    Declare <DOC>:  External    Default Value "1001 "
    Declare <RESULT_1>    Default Value ""
    Set <OFFSET> Value from Concatenate <DCT>
    Set <OFFSET> Value from Concatenate <DOC>
    >>INSERT
```

Ready   NUM

Associating a Validation

Once you declare the validation, you have a place in which you can store values that you have entered to header controls and grid columns during scripting. During validation association, you choose values that you want to validate. Then you pair or associate those values with columns in the database.

▶ To associate a validation

1. In the command menu of the OneWorld Scripting Tool form, click Associate a Validation Column.
2. Choose a validation name that you created in the Declare New Validation phase. Click the scroll arrow and highlight the name to do so.
3. Choose a column name from the Database Column list.
4. In the Value Type list, choose Validation Value.

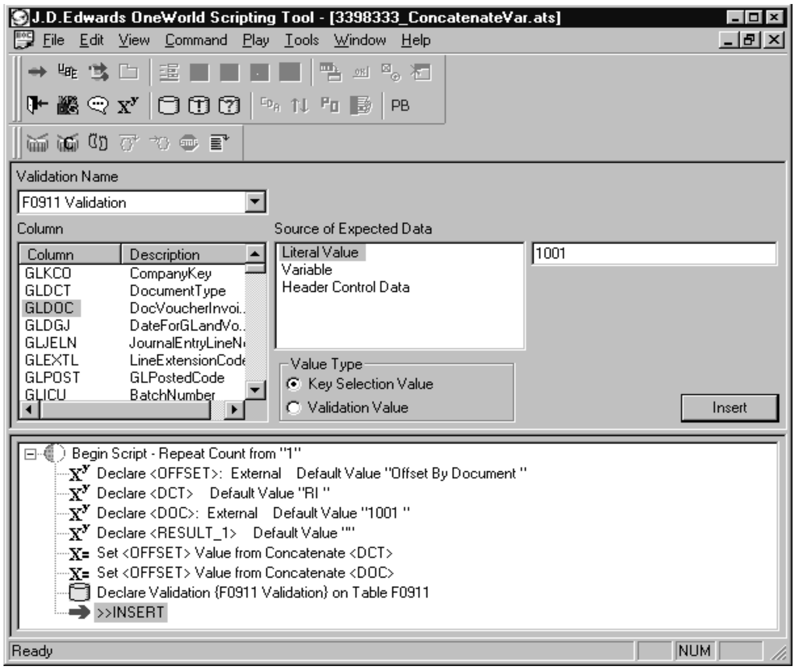
   This associates the chosen column with a particular value.

5. Click on a selection from the Source of Expected Data list.

   The source can be a literal value, variable, header control, or grid column.

6. In the value selection list enter a literal value or choose the name of a variable, header control or grid column.

   This step specifies the value that you associate with the database column.


J.D.Edwards OneWorld Scripting Tool - [3398333_ConcatenateVar.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

Validation Name: F0911 Validation

Column

Column	Description
GLKCO	CompanyKey
GLDCT	DocumentType
GLDOC	DocVoucherInvoi...
GLDGJ	DateForGLandVo...
GLJELN	JournalEntryLineN
GLEXTL	LineExtensionCode
GLPOST	GLPostedCode
GLICU	BatchNumber

Source of Expected Data: Literal Value / Variable / Header Control Data — 1001

Value Type: ● Key Selection Value ○ Validation Value

```
Begin Script - Repeat Count from "1"
    Declare <OFFSET>: External   Default Value "Offset By Document "
    Declare <DCT>   Default Value "RI "
    Declare <DOC>: External   Default Value "1001 "
    Declare <RESULT_1>   Default Value ""
    Set <OFFSET> Value from Concatenate <DCT>
    Set <OFFSET> Value from Concatenate <DOC>
    Declare Validation {F0911 Validation} on Table F0911
    >>INSERT
```

Note: With playback turned on, if you choose a header control or grid column, OneWorld Scripting Tool highlights with a BlueCue the designated control or column in the OneWorld form. Be sure to choose a header control or grid column to which you have previously entered a value. Likewise, if you have chosen the name of a variable, be sure that you have followed the steps outlined previously for setting the variable's value.

7. Click the Insert button.
8. Before you finish the association, make sure you write a command that follows steps 1-7, but choose a Key Selection Value.

   This step associates data from all the individual columns from which you expect to have data returned with a single key column.

Executing a Validation

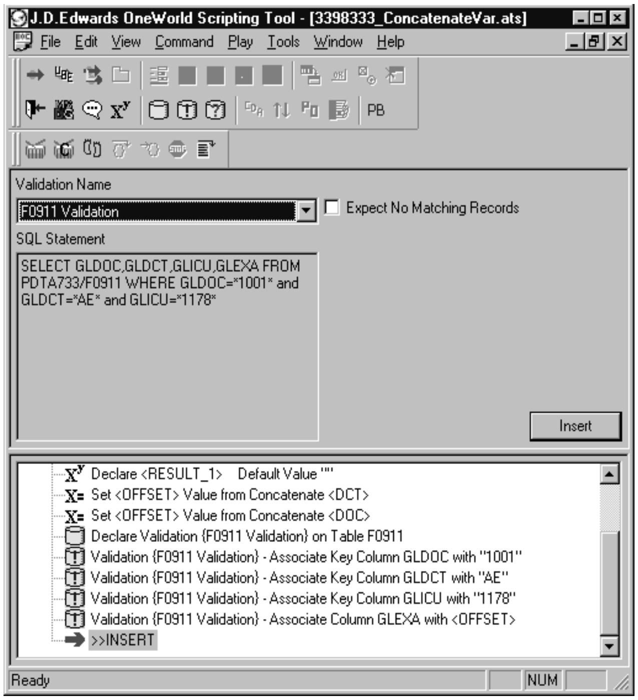
After you have declared the validation and associated the data with database tables and columns, you are ready to execute the validation.

▶ To execute a validation

1. In the command menu of the OneWorld Scripting Tool form, click Execute Validation.
2. Click the scroll button in the Validation Name list.

3. Highlight the name of the validation you declared and now want to execute.

   Note: OneWorld Scripting Tool populates the SQL Statement list. The statement contains the data dictionary aliases of the tables and columns you associated with the data you entered, the name of the validation, and the key selection value.


J.D.Edwards OneWorld Scripting Tool - [3398333_ConcatenateVar.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

Validation Name: F0911 Validation

□ Expect No Matching Records

SQL Statement:

```
SELECT GLDOC,GLDCT,GLICU,GLEXA FROM
PDTA733/F0911 WHERE GLDOC=*1001* and
GLDCT=*AE* and GLICU=*1178*
```

```
Declare <RESULT_1>    Default Value ""
Set <OFFSET> Value from Concatenate <DCT>
Set <OFFSET> Value from Concatenate <DOC>
Declare Validation {F0911 Validation} on Table F0911
Validation {F0911 Validation} - Associate Key Column GLDOC with "1001"
Validation {F0911 Validation} - Associate Key Column GLDCT with "AE"
Validation {F0911 Validation} - Associate Key Column GLICU with "1178"
Validation {F0911 Validation} - Associate Column GLEXA with <OFFSET>
>>INSERT
```

4. Click the Insert button.

   Later in the script you might delete the records. In turn, you might want to validate that the deleted records are no longer in the database.

5. Repeat steps 1-3.
6. Choose the Expect No Match Records checkbox option.
7. Click the Insert button.

   Note: You can declare and set the value of a variable to test validation success. For further discussion of how to accomplish this, see *Validation Success*.

The Command Line Message Command

You might want to encapsulate running another program or programs within a OneWorld Scripting Tool script. For example, you might prepare a PowerPoint or Excel presentation that you want to include within the script. Once you have run that presentation, you might decide to close the program, then return to OneWorld Scripting Tool to continue scripting inputs in OneWorld.

You are able to complete these tasks by clicking Command Line in the command menu. Using this command, you type the path to the program you want to run, much as you use the Run function in Windows. OneWorld Scripting Tool opens the program and the document or presentation you have created.

You can also send a command line message to make captures of OneWorld forms that are current at designated points in the script playback process. You can make these captures in a particular OneWorld language version, and you can store them in a directory and file that you establish.

Scripting a Command Line Message Command

Scripting the command line message requires that you know the path to the program you want to run and, for example, a document in the program that you want to open. You can also send a command line message that captures the OneWorld window that is currently open. If you have established a folder within a directory for your screen captures, OneWorld Scripting Tool automatically sends them to the folder you created in the path that you have set up.

This chapter covers the following tasks:

- ❑ Sending a command line message
- ❑ Capturing a current OneWorld window

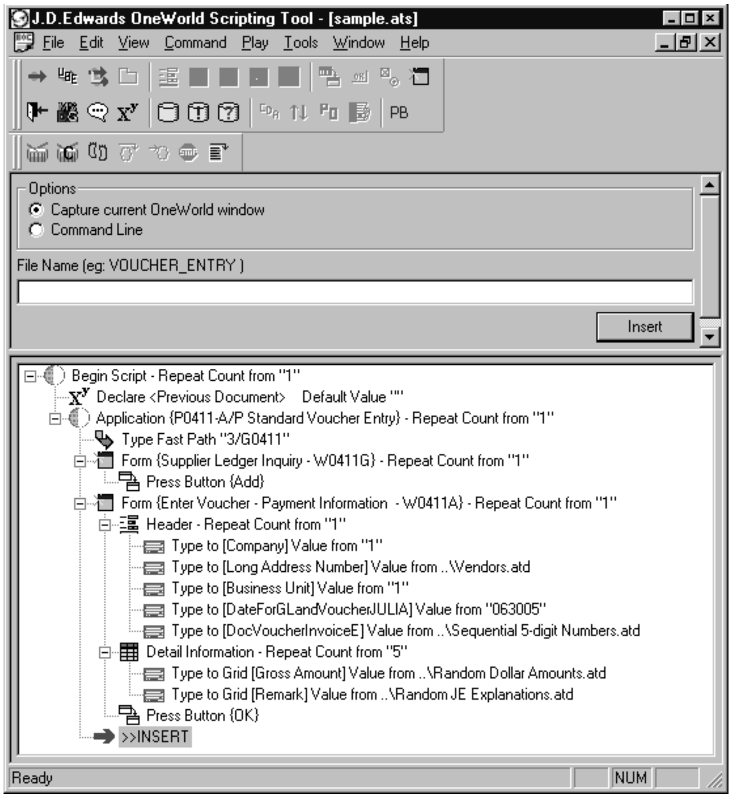
Sending a Command Line Message

To open a program using the Command Line Message command, you type the program's path into the unpopulated command line list. OneWorld Scripting Tool reads the path and opens the program.

▶ To send a command line message

1. In the command menu of the OneWorld Scripting Tool form, click Command Line.

   The command pane displays the unpopulated Command Line list and radio button options.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

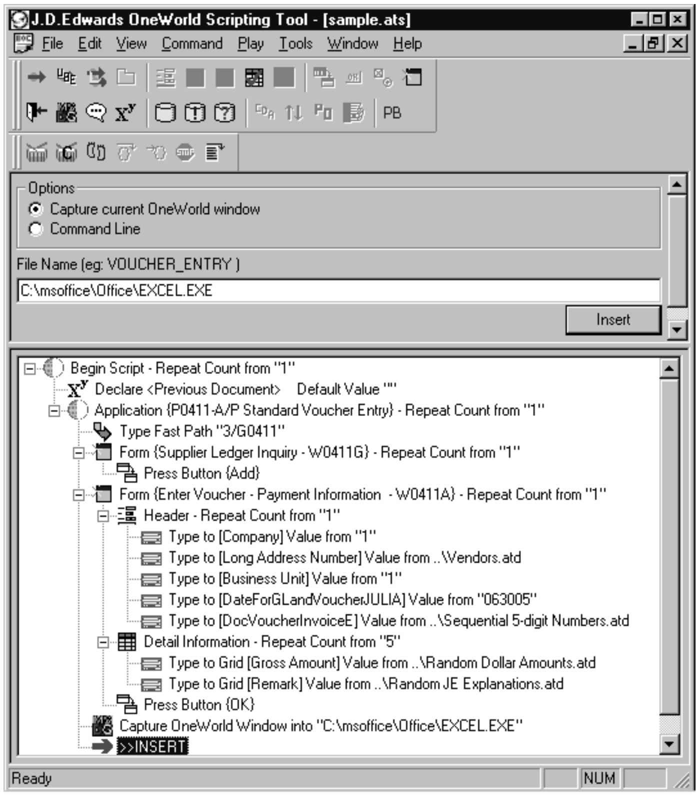
Options
● Capture current OneWorld window
○ Command Line

File Name (eg: VOUCHER_ENTRY)

```
Begin Script - Repeat Count from "1"
  Declare <Previous Document>    Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        Type to [Long Address Number] Value from ..\Vendors.atd
        Type to [Business Unit] Value from "1"
        Type to [DateForGLandVoucherJULIA] Value from "063005"
        Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
      Detail Information - Repeat Count from "5"
        Type to Grid [Gross Amount] Value from ..\Random Dollar Amounts.atd
        Type to Grid [Remark] Value from ..\Random JE Explanations.atd
      Press Button {OK}
    >>INSERT
```

2. Choose the Command Line option.
3. In the Command Line list, type in the path to the program you want to open.

   Note: Turn playback off if you do not want the program to open while you are writing the script.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

Options
● Capture current OneWorld window
○ Command Line

File Name (eg: VOUCHER_ENTRY) C:\msoffice\Office\EXCEL.EXE

```
Begin Script - Repeat Count from "1"
  Declare <Previous Document>    Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        Type to [Long Address Number] Value from ..\Vendors.atd
        Type to [Business Unit] Value from "1"
        Type to [DateForGLandVoucherJULIA] Value from "063005"
        Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
      Detail Information - Repeat Count from "5"
        Type to Grid [Gross Amount] Value from ..\Random Dollar Amounts.atd
        Type to Grid [Remark] Value from ..\Random JE Explanations.atd
      Press Button {OK}
    Capture OneWorld Window into "C:\msoffice\Office\EXCEL.EXE"
    >>INSERT
```

4. Click the Insert button.

Capturing a Current OneWorld Window

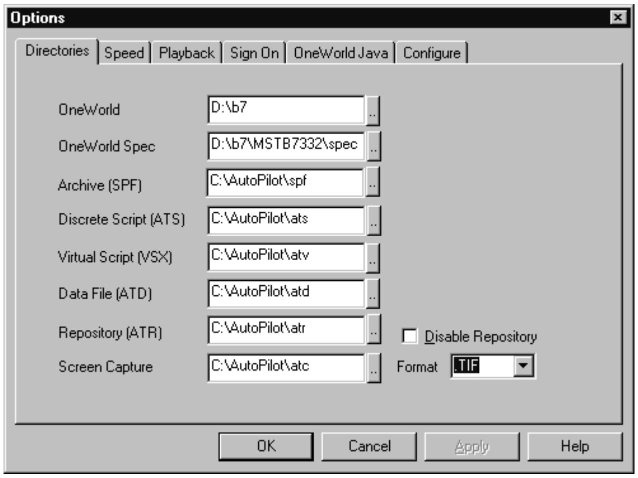
You can set up in advance the path and file extension for any OneWorld window captures that you make using the Capture command. You do so by clicking Tools, then choosing Options from the drop-down menu. You then click the Directories tab in the Options form.

In the Screen Capture field under the Directories tab, you type the path that OneWorld Scripting Tool uses to send your OneWorld form captures. You then choose the format in which you want the images to be saved. You can click the scroll button to browse for the format you desire.

Once you have made these designations and you want to use the Command Line command to capture a OneWorld window, you need only to type in a name for the image in the Path/File field, and OneWorld Scripting Tool immediately routes the image to the drive and directory you have designated. OneWorld Scripting Tool also automatically converts the captured window to the format you specified.

▶ To use the Tools option to set up OneWorld window captures

1. In the menu bar of the OneWorld Scripting Tool form, click Tools.
2. Click Options.
3. In the Options form, click the Directories tab.
4. In the Screen Capture control, type the path that establishes where you want OneWorld Scripting Tool to send OneWorld window captures you make.
5. In the Format control, click the scroll button to choose the file extension, such as .tif, you want to use for the window capture.



OneWorld: D:\b7

OneWorld Spec: D:\b7\MSTB7332\spec

Archive (SPF): C:\AutoPilot\spf

Discrete Script (ATS): C:\AutoPilot\ats

Virtual Script (VSX): C:\AutoPilot\atv

Data File (ATD): C:\AutoPilot\atd

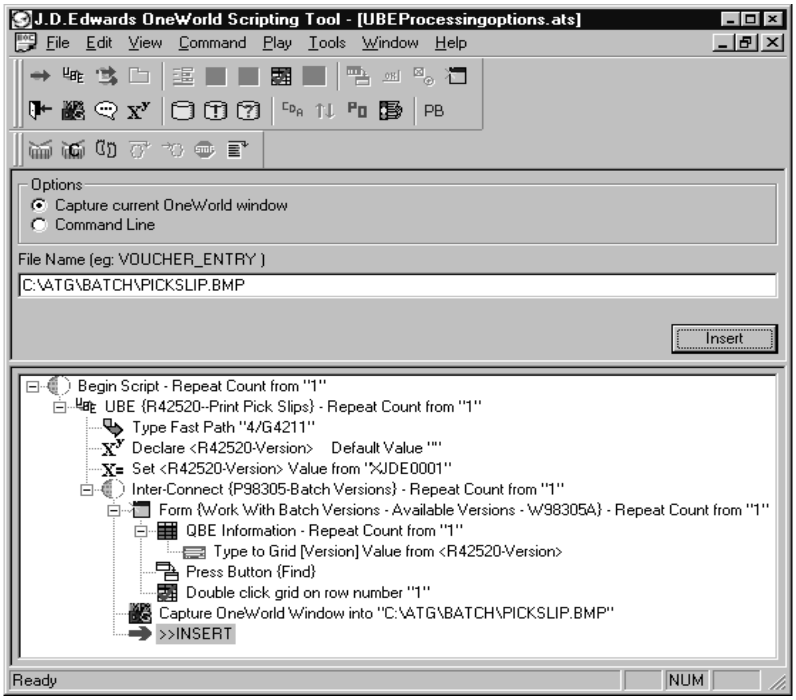
Repository (ATR): C:\AutoPilot\atr

□ Disable Repository

Screen Capture: C:\AutoPilot\atc

Format: TIF

6. Click OK.
7. With a script open, in the command menu of the OneWorld Scripting Tool form, click Command Line.
8. Choose the Capture current OneWorld window radio button option.
9. With a OneWorld window active, place the cursor in the Path/File field of the OneWorld Scripting Tool command pane and enter a name for the image you are going to capture.



J.D.Edwards OneWorld Scripting Tool - [UBEProcessingoptions.ats]

Options
● Capture current OneWorld window
○ Command Line

File Name (eg: VOUCHER_ENTRY)

C:\ATG\BATCH\PICKSLIP.BMP

```
Begin Script - Repeat Count from "1"
  UBE {R42520--Print Pick Slips} - Repeat Count from "1"
    Type Fast Path "4/G4211"
    Declare <R42520-Version>    Default Value ""
    Set <R42520-Version> Value from "XJDE0001"
    Inter-Connect {P98305-Batch Versions} - Repeat Count from "1"
      Form {Work With Batch Versions - Available Versions - W98305A} - Repeat Count from "1"
        QBE Information - Repeat Count from "1"
          Type to Grid [Version] Value from <R42520-Version>
        Press Button {Find}
        Double click grid on row number "1"
      Capture OneWorld Window into "C:\ATG\BATCH\PICKSLIP.BMP"
      >>INSERT
```

10. Click the Insert button.

    OneWorld Scripting Tool automatically sends the captured image of the OneWorld window to the drive and directory that you specified in the Options form.

Script Pane

Working with the Script Pane

As you are working on a OneWorld Scripting Tool script, or after you have completed a script, you can work in the script pane to delete, modify, and move the commands that you have created. Working with the script pane requires that you understand the structure of the script tree that you build as you insert commands. You must also learn how to work with the tree to change its structure.

Parent-child relationships make up the script tree. Every script begins with the Begin Script command, from which any number of commands descend. Subsequent context commands are the parents of action commands and sometimes of other context commands. These parent context commands and their children make up nodes in the script pane, which are identified with plus signs. OneWorld Scripting Tool indents any command that is the child of another command. You can change the sequence of commands and the relationship between commands by dragging and dropping. For example, you can make one context command the child of another. This means that any changes you make to the parent command affect the child command.

Whatever the modifications are that you make to the script, the important thing to remember is that these modifications change the way that the script is run. The changes that you make should be based on what you want to accomplish by running the script. For example, you might drag a declared variable command line to the top of the script to make it global because you need all the commands in the script to have access to the value that you set for the variable.

This section discusses the four main components of working with the OneWorld Scripting Tool script pane:

❑ Understanding the script pane structure

❑ Modifying scripts

❑ Script retention and reuse

❑ Reusing scripts

Understanding the Script Pane Structure

As you write OneWorld Scripting Tool scripts, you build a tree structure in the script pane that is based on parent-child relationships. Once you understand the structure of scripts, you can more easily modify your scripts and customize them to your specifications.

Since your scripting requires a OneWorld context, context commands form the core of each script you write. Each context command that you write and insert to the script creates a node, which you can identify in the script pane by a plus/minus sign that you can use to expand or collapse the node.

Each command that you write, whether it is a context command or an action command, forms a command line in the script. Action command lines are attached to context command nodes, are indented beneath the node, and are affected by any changes that you make to the node, such as changing the repeat count.

In some cases, a context command line that forms a node is also a child of another context command line that forms a node. In this case, OneWorld Scripting Tool indents the child node, which is, again, affected by any changes you make to the parent command.

Finally, the script pane contains an insertion cursor, which indicates the position of the next command that you write. You can change the position of the insertion cursor either by clicking a command line or by dragging the insertion cursor. You can drop the insertion cursor into the script tree as a child of a node, meaning that the next command you write is a child, or you can make the next command a parent that is independent of changes to other nodes in the script.

This chapter discusses the following components of the script pane:

- ❑ Command lines
- ❑ Insertion cursor
- ❑ Nodes

Command Lines

Command lines give visual expression to the choices that you make in the command pane to create context or action commands. The command lines in the script pane express either the OneWorld context in which you create your script or the actions that you take within the context. For example, a header in a form is a context; the action you take within that context might be typing data to a specified control.

Context commands direct OneWorld Scripting Tool to applications, UBEs, processing options, interconnected applications, forms, header controls, grid columns and QBE lines in OneWorld. They therefore express the environment in which actions, such as typing data and pressing buttons, are carried out. Context commands form the trunk of the script tree.

The context command that initializes a series of action commands and other context commands forms a node. The expand/collapse button in the script pane identifies the node. The expand/collapse button that represents the node appears in the script pane as a plus or minus sign inside a box next to the context command line.

In sum, context commands:

- Form nodes that can be identified in the script with the node symbol or button, which displays as a plus or minus sign.
- Form nodes that can be expanded or collapsed by clicking the expand/collapse button.
- Form nodes that function as the parents of action and sometimes other context commands. These children are indented beneath the parent context command in the script pane.
- Can form discrete command line units. If two nodes are parallel to one another, commands that you add to one node do not affect the node that is parallel to it.
- Can initiate a sequence of other commands. The sequence can consist of the action commands and other context commands that are children of the parent command.
- May be played back multiple times if you change the repeat count of a node. Any commands that are attached to a context command in the script pane are played back as many times as you specify in the repeat count.

You script action commands to specify actions to be taken after you have scripted context commands.

In contrast to context commands, action commands:

- Must be attached to, or be the children of, a context command
- Cannot have children attached to them
- Cannot be assigned repeat counts

- Are always indented beneath context commands in the script pane, showing that they are subservient to context commands in the command hierarchy.

See Also

- *Panes in the OneWorld Scripting Tool Form*

Insertion Cursor

The insertion cursor, identified in the script pane as a red arrow, points to the position in the script at which you can insert a new command. If you insert commands sequentially without making any adjustments to the script, the insertion cursor appears at the end of the script each time that you insert a command.

However, you can move the insertion cursor from one point in the script to another by clicking a command line. This moves the insertion cursor to the position directly below an action command line. If you then create and insert a new command, it appears at the point of the insertion cursor.

If you click a context command line, the insertion cursor appears at the end of the script, attached by a direct line to the node formed by the context command. If you leave the insertion cursor in this position, a new context command that you write creates a node, indicated by a minus or plus sign, that is parallel to the node on you clicked. New commands that you write are attached to this node.

Nodes

A node consists of a parent context command and any related context commands and/or action commands that you attach to it. Once you have established a node by scripting a context command, you can write action commands and sometimes other context commands to develop the node and the script.

This topic covers the following components of nodes and concepts about nodes that are integral to understanding the script pane structure:

❑ Expand/collapse button

❑ Parallel nodes

❑ Indented nodes

❑ Drag and drop

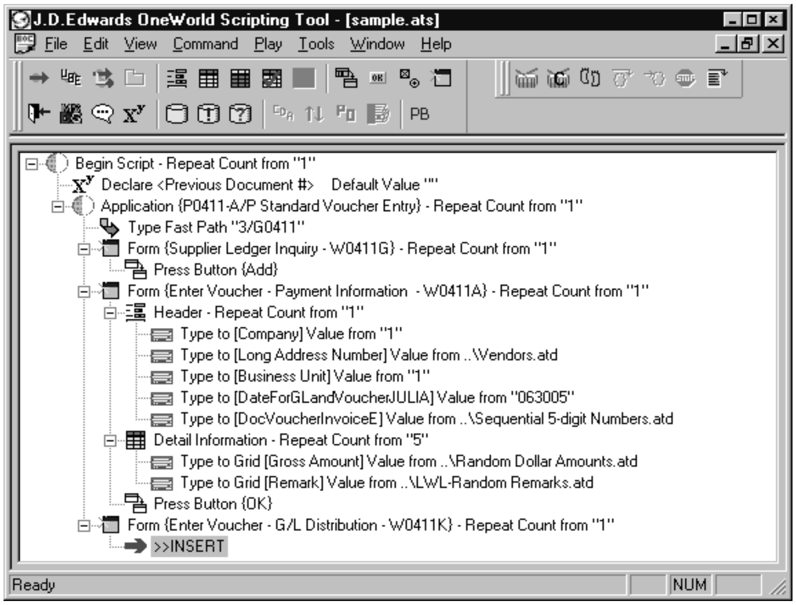
- ❑ Repeat count

Expand/Collapse Button

The button that identifies the node also allows you to expand or collapse it. The expanded node reveals all command lines that are attached to the node. The collapsed node reveals only the context command that you scripted to initiate the node. When you expand a node, the button displays a minus sign. When you collapse a node, the button displays a plus sign.

Parallel Nodes

A change that you make to a node that is parallel to another affects only the node you change because it is parallel to the other node in the script hierarchy. They are siblings to one another. For example, if you write an Application command and write two form commands in the same application, the Form commands are represented by nodes that are parallel to one another in the script pane. Any change that you make to one does not affect the other.

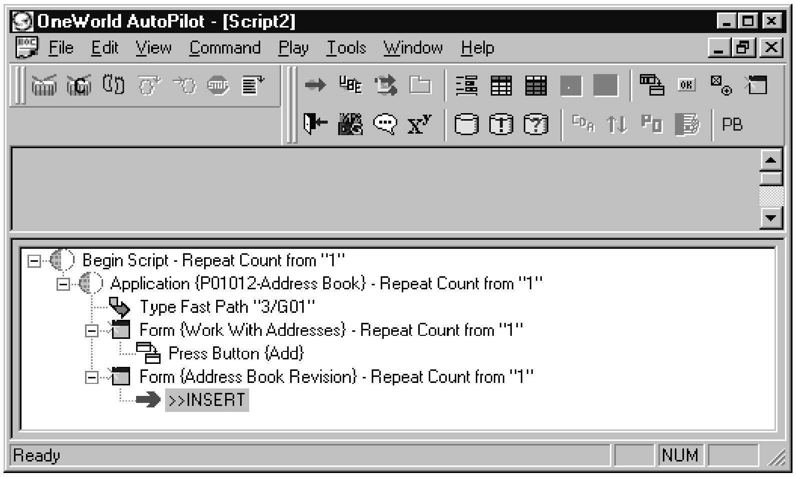


```
Begin Script - Repeat Count from "1"
  Declare <Previous Document #>    Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        Type to [Long Address Number] Value from ..\Vendors.atd
        Type to [Business Unit] Value from "1"
        Type to [DateForGLandVoucherJULIA] Value from "063005"
        Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
      Detail Information - Repeat Count from "5"
        Type to Grid [Gross Amount] Value from ..\Random Dollar Amounts.atd
        Type to Grid [Remark] Value from ..\LWL-Random Remarks.atd
      Press Button {OK}
    Form {Enter Voucher - G/L Distribution - W0411K} - Repeat Count from "1"
      >>INSERT
```

Indented Nodes

A node that is indented beneath another node in the script pane is affected by any change that you make to the parent. The indention of nodes reveals the hierarchy of context commands. For example, if you want to work with a particular form in OneWorld, you must first choose a OneWorld application. Therefore, OneWorld Scripting Tool inserts and indents the Form command line below the Application command line.

Similarly, if you decide to add a form by pressing a button, OneWorld Scripting Tool inserts and indents the action command Press Toolbar Button below the Form command line.

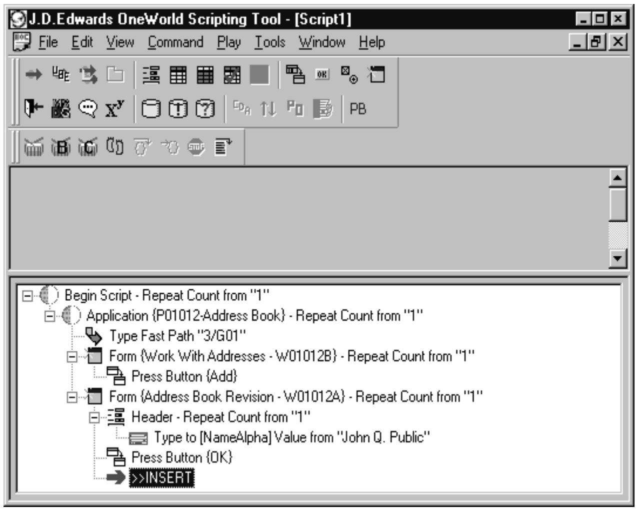

OneWorld AutoPilot - [Script2]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
  Application {P01012-Address Book} - Repeat Count from "1"
    Type Fast Path "3/G01"
    Form {Work With Addresses} - Repeat Count from "1"
      Press Button {Add}
    Form {Address Book Revision} - Repeat Count from "1"
      >>INSERT
```

The hierarchy of nodes expresses the logic you follow to build OneWorld Scripting Tool scripts. For example, you might write the following sequence of commands to enter data to the header of a OneWorld form and update the database by clicking OK:

- Application
- Form
- Header
- Type to
- Press Toolbar Button {OK}

Because you must launch an application before you can write any of the subsequent commands, the Application command forms a parent node in the script pane. Likewise, you must launch a form before you can enter data in a header control, so the Form command is a parent to the Header command, which is indented. Finally the header is the context for performing the actions of entering data to a control and clicking OK, so the action commands Type to and Press Toolbar button are indented beneath the Header command line.


J.D.Edwards OneWorld Scripting Tool - [Script1]
File Edit View Command Play Tools Window Help
PB

```
Begin Script - Repeat Count from "1"
  Application {P01012-Address Book} - Repeat Count from "1"
    Type Fast Path "3/G01"
    Form {Work With Addresses - W01012B} - Repeat Count from "1"
      Press Button {Add}
    Form {Address Book Revision - W01012A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [NameAlpha] Value from "John Q. Public"
      Press Button {OK}
      >>INSERT
```

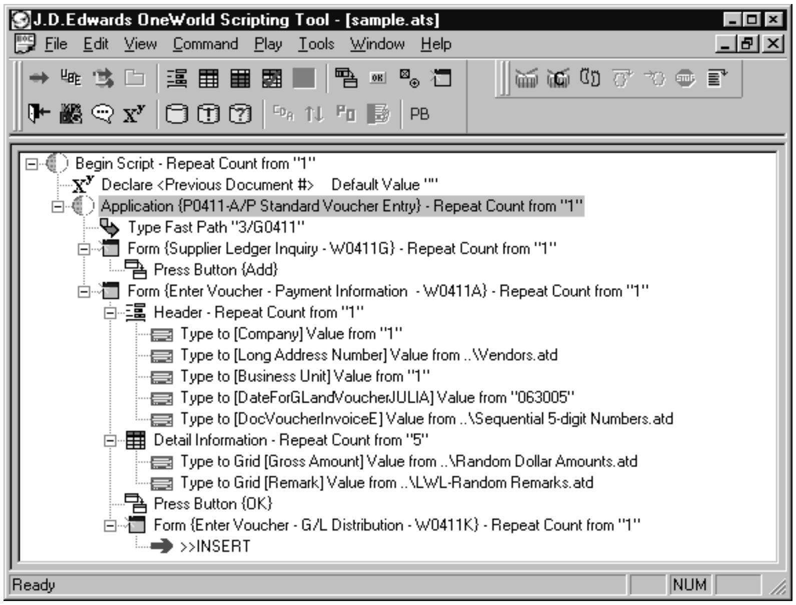
Any context or action command that you insert to the script as a child of another command is affected by changes you make to the parent. For example, if you change the repeat count in an Application command line to 3, during playback the application launches three times, as does the form and any action commands that you write, such as clicking a toolbar button in the form, are performed three times.

Drag and Drop

You change the sequence of commands and the structure of the script by using the mouse to drag and drop commands. Remember that several rules govern how you can use the drag-and-drop capability in the script pane:

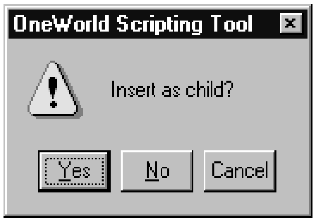
- An action command within one context command node cannot be dragged into another context command node. For example, you cannot take a Type to command attached to a Form command node and drag it into another Form command node.
- A context command node attached to one Application node cannot be dragged into another Application node.
- A context command node that you drag onto another context command node and insert as a child includes all commands that are attached to it.
- A context command node that you insert as a child is included in the playback of the parent context command node.
- The repeat count of the parent context command node applies to that node and to any other nodes that are attached to it as children.

You can make one parallel node a child of the other if you drag and drop it into the node that was formerly its sibling. OneWorld Scripting Tool indicates the parent-child relationship by indenting one node beneath the other.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
    Declare <Previous Document #>    Default Value ""
    Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
        Type Fast Path "3/G0411"
        Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
            Press Button {Add}
        Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
            Header - Repeat Count from "1"
                Type to [Company] Value from "1"
                Type to [Long Address Number] Value from ..\Vendors.atd
                Type to [Business Unit] Value from "1"
                Type to [DateForGLandVoucherJULIA] Value from "063005"
                Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
            Detail Information - Repeat Count from "5"
                Type to Grid [Gross Amount] Value from ..\Random Dollar Amounts.atd
                Type to Grid [Remark] Value from ..\LWL-Random Remarks.atd
            Press Button {OK}
            Form {Enter Voucher - G/L Distribution - W0411K} - Repeat Count from "1"
                >>INSERT
```

Before OneWorld Scripting Tool creates the parent-child relationship, it displays the dialog box asking you to confirm that you want to insert one node as a child of another.



Insert as child?

Repeat Count

Every context command that creates a node in the script pane contains a repeat count. The repeat count specifies the number of times that OneWorld Scripting Tool plays the node and all commands attached to it. You can change the repeat count by selecting the node, entering a new repeat count in the command pane, and clicking the Update button.

See Also

- *Updating the Repeat Count in a Node*

Modifying Scripts

OneWorld Scripting Tool allows you to work in the script during and after script creation. This permits you to customize scripts to your precise specifications in order to test most effectively OneWorld applications. You can also alter the structure of the script tree either stylistically, by expanding or collapsing nodes or substantively, by adding, deleting, editing, or dragging commands.

You can change the order of the commands and therefore the structure of the scripts you create, either as you are scripting or after you have completed scripting a series of commands. You can use your mouse, your keyboard and the OneWorld Scripting Tool command pane to add, delete, or edit commands. You can also modify the structure of the script tree by using the mouse to move the insertion cursor and command lines.

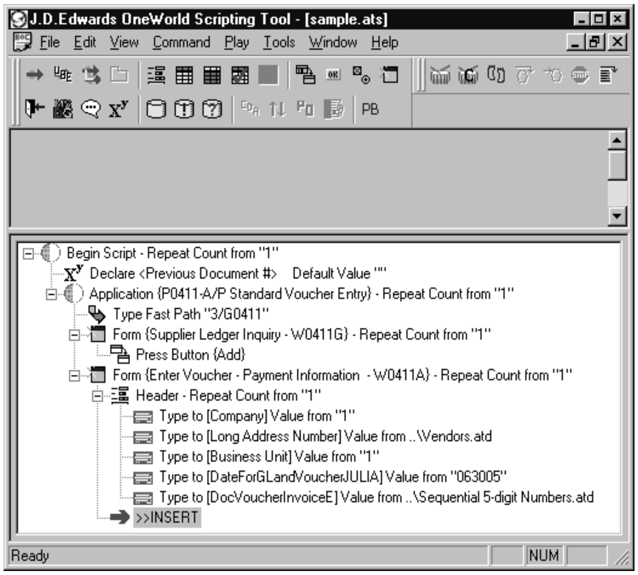
This chapter covers the following tasks:

- ❑ Expanding and collapsing a node
- ❑ Adding command lines
- ❑ Deleting command lines
- ❑ Moving command lines
- ❑ Editing command lines

Expanding and Collapsing a Node

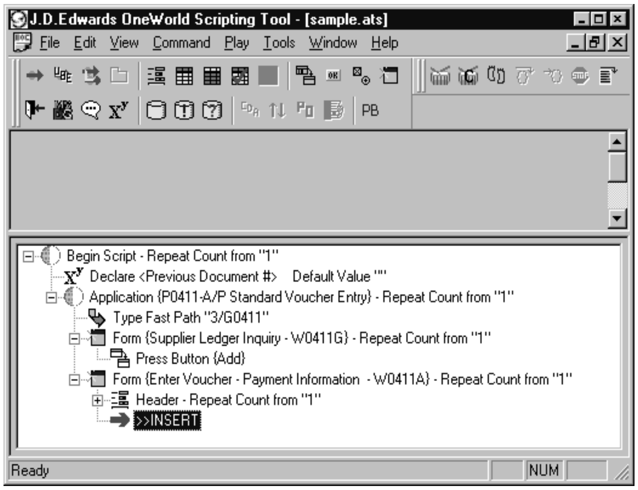
You might want to see the script that you create in its entirety, with all command lines exposed. When you expand all the nodes in a script, you are able to do just that. However, as your scripts get longer, you may want to see only a portion of the scripted commands. In that case, you can collapse nodes so that only the context commands that originated them are visible in the script pane. You can collapse or expand the entire script or certain portions of the script by clicking the node buttons, which are identified in the script pane by plus/minus signs.

Clicking a node button with a plus sign expands the node.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

Begin Script - Repeat Count from "1"
Declare <Previous Document #> Default Value ""
Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
Type Fast Path "3/G0411"
Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
Press Button {Add}
Form {Enter Voucher - Payment Information - W0411A} - Repeat Count from "1"
Header - Repeat Count from "1"
Type to [Company] Value from "1"
Type to [Long Address Number] Value from ..\Vendors.atd
Type to [Business Unit] Value from "1"
Type to [DateForGLandVoucherJULIA] Value from "063005"
Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
>>INSERT

Clicking a node button with a minus sign collapses the node.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

Begin Script - Repeat Count from "1"
Declare <Previous Document #> Default Value ""
Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
Type Fast Path "3/G0411"
Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
Press Button {Add}
Form {Enter Voucher - Payment Information - W0411A} - Repeat Count from "1"
Header - Repeat Count from "1"
>>INSERT

You can choose any point at which to collapse the tree. Choosing parent and child nodes to collapse provides a further illustration of the script tree structure. For example, when you click the Expand/Collapse button on a parent node, any nodes that you have inserted to it as children also collapse. When you click a child node, only that node collapses.

This topic covers the following tasks:

- Collapsing the script tree
- Collapsing the script tree one node at a time
- Expanding the script tree
- Expanding the script tree one node at a time

▶ To collapse the script tree

1. In the script pane of the OneWorld Scripting Tool form, expand any or all the nodes by clicking any node button that shows a plus sign.

   Expanded nodes show a button with a minus sign.

2. click any node that shows a minus sign.

   The node you clicked collapses and shows only the context command line.

▶ To collapse the script tree one node at a time

1. In the script pane of the OneWorld Scripting Tool form, expand all nodes by clicking any buttons that show a plus sign.
2. Collapse the node at the bottom of the script tree by clicking the node button.
3. Proceeding up the script tree, collapse each expanded node by clicking the node button.

   When you reach the top of the script tree, only the Begin Script command remains.

▶ To expand the script tree

1. In the script pane of the OneWorld Scripting Tool form, collapse any or all the nodes by clicking any node button that shows a minus sign.

   Collapsed nodes show a button with a plus sign.

2. click any node that shows a plus sign.

   The node you clicked expands and shows the context command line and any commands attached to it.

▶ To expand the script tree one node at at time

1. In the script pane of the OneWorld Scripting Tool form, expand all nodes by clicking any buttons that show a plus sign.
2. Collapse the node at the bottom of the script tree by clicking the node button.
3. Proceeding up the script tree, collapse each expanded node by clicking the node button.

When you reach the top of the script tree, only the Begin Script command remains.

4. click the node button of the Begin Script command line.

   The next node appears.

5. click the next node button with a plus sign that appears.
6. Continue to click each node button with a plus sign until you reach the bottom of the script tree and all command lines are visible.

Adding Command Lines

You might decide that you want to insert a command to your script after you have passed the point you want to insert it. For example, you might decide that you want to script an input to a header control after you've scripted the move to another form, or you might decide that you want to insert a Grid command so that you can script grid column inputs in a form after you've moved on to another form. You can accomplish this by placing the insertion cursor at the point in the script to which you want to insert a new command.

You can only use the insertion cursor to insert a new command to the script if the buttons in the cool bar are lit and active. If they are not, left-click and hold down the mouse button, drag the insertion cursor on top of a command line and release the mouse button. If you are asked to insert the cursor as a child, click Yes.

▶ To add a command line to an existing script

1. In the script pane of the OneWorld Scripting Tool form, highlight the insertion cursor by clicking it.
2. Left-click and hold down the mouse button.
3. Drag the insertion cursor to the point in the script at which you want to insert a new command.
4. As you drag the cursor, watch the indicator arrow that appears.

   The arrow, which points up or down, indicates whether you place the cursor above or below a highlighted command line.

5. When you reach the point at which you want to insert the new command, release the mouse button.
6. Follow the steps required to insert the desired command to the script.

Note: If you cannot insert a command at a given point in the script, OneWorld Scripting Tool displays a disallow signal as you are dragging the insertion cursor.

Deleting Command Lines

You might want to remove lines from a script you are working on. To do so, you can use the mouse. Remember that if you delete a command line that forms a node, you also delete the context command that initiated the node and all commands attached to it. By contrast, if you delete an action command, you delete only the command you choose.

▶ To delete a command line

1. In the script pane of the OneWorld Scripting Tool form, highlight a command by clicking it in the script.
2. Right-click the mouse.
3. click Delete.

Moving Command Lines

You can further modify an existing script by moving command lines. You can move action commands that are attached to a context command to change, for example, the order that the action commands play back.

You can change the order of action commands within a context command node by using OneWorld Scripting Tool's drag-and-drop capability. During playback, OneWorld Scripting Tool replays these commands within the node in the new order. However, you can also change the structure of a script by moving a context command and inserting it as a child of another context command. When you do so, a change that you make to the repeat count of the parent context command also applies to the child context command.

This topic covers the following tasks:

- Changing the sequence of action commands
- Changing the sequence of context commands

▶ To change the sequence of action commands

1. In the script pane in the OneWorld Scripting Tool form, highlight an action command line by clicking it.
2. Left-click the mouse.

3. Holding down the mouse button, drag the highlighted command line until it is on top of another command that you choose.

   The indicator arrow that appears as you are dragging the command line tells you if the command appears above or below the targeted command line in the script pane.

4. Release the mouse.

   Repeat these steps as often as you desire to rearrange the order of the action commands.

▶ To change the sequence of context commands

1. In the script pane of the OneWorld Scripting Tool form, highlight a node by clicking it.
2. Left-click the mouse, then hold the button down and drag the mouse.

   A disallow symbol prevents you from dropping the node in an improper spot in the script. If the node cannot be dropped, the disallow symbol disappears.

3. When the target node is highlighted and the indicator arrow is pointing downward, release the mouse.
4. In the dialogue box, click Yes or No in answer to the question "Insert as Child?"

   OneWorld Scripting Tool inserts the dragged node in an indented position, or as a child to the target node if you answer Yes. If you answer No, the nodes are parallel.

Editing Command Lines

You can also make substantive changes to the content of the script by editing the command lines in the script pane. You make these changes by highlighting a command line in the script pane, then working in the command pane to make new choices from lists to update the content of the command.

Note: Press Button command lines cannot be edited. These must be deleted or added to the script as desired.

This topic covers the following tasks:

❑ Editing an action command line

❑ Editing a context command line

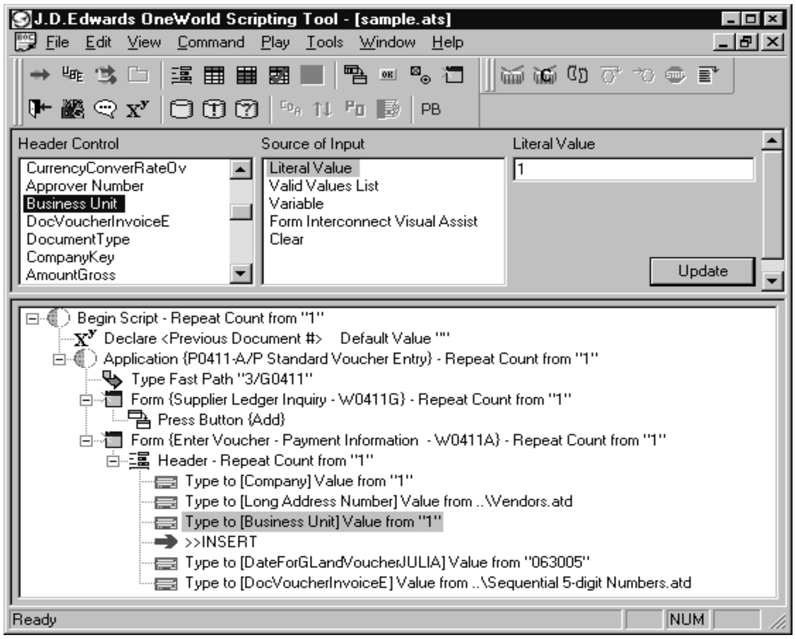
Editing an Action Command Line

While you can change only the repeat counts of context command lines, you can change the content of many action commands by clicking the command line, and then making new choices in the command pane.

▶ To edit an action command line

1. In the script pane of the OneWorld Scripting Tool form, click an action command line.

   The command pane displays lists from which you can make choices to update the content of the command line.



Header Control
- CurrencyConverRateOv
- Approver Number
- Business Unit
- DocVoucherInvoiceE
- DocumentType
- CompanyKey
- AmountGross

Source of Input
- Literal Value
- Valid Values List
- Variable
- Form Interconnect Visual Assist
- Clear

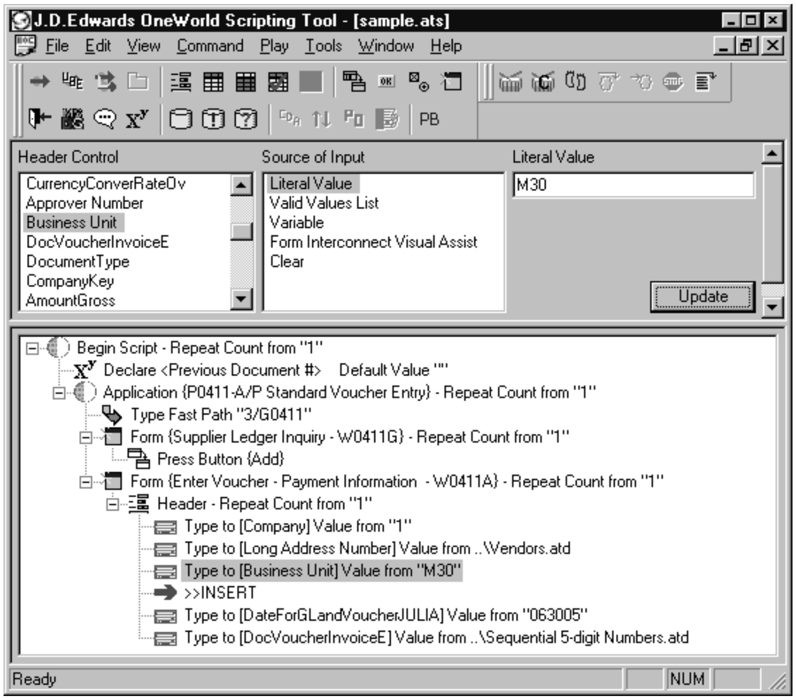
Literal Value: 1

```
Begin Script - Repeat Count from "1"
    Declare <Previous Document #>    Default Value ""
    Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
        Type Fast Path "3/G0411"
        Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
            Press Button {Add}
        Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
            Header - Repeat Count from "1"
                Type to [Company] Value from "1"
                Type to [Long Address Number] Value from ..\Vendors.atd
                Type to [Business Unit] Value from "1"
                >>INSERT
                Type to [DateForGLandVoucherJULIA] Value from "063005"
                Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
```

   Note: OneWorld Scripting Tool highlights the original choices that you made in the command pane.

2. In the command pane lists, click any new choices that you want to make.
3. Click the Update button.

   OneWorld Scripting Tool changes the command line in the script pane to reflect the changes that you made.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Update
Ready
NUM

Header Control: CurrencyConverRateOv, Approver Number, Business Unit, DocVoucherInvoiceE, DocumentType, CompanyKey, AmountGross

Source of Input: Literal Value, Valid Values List, Variable, Form Interconnect Visual Assist, Clear

Literal Value: M30

```
Begin Script - Repeat Count from "1"
    Declare <Previous Document #>   Default Value ""
    Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
        Type Fast Path "3/G0411"
        Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
            Press Button {Add}
        Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
            Header - Repeat Count from "1"
                Type to [Company] Value from "1"
                Type to [Long Address Number] Value from ..\Vendors.atd
                Type to [Business Unit] Value from "M30"
                >>INSERT
                Type to [DateForGLandVoucherJULIA] Value from "063005"
                Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
```

Editing a Context Command Line

The substance of the context command itself cannot be changed unless you delete it and insert a new one. However, you can change the number of times that OneWorld Scripting Tool loops through the node during script playback.

▶ To edit a context command line

1. In the script pane of the OneWorld Scripting Tool form, click the context command line in the script pane.
2. In the command pane, choose a value from the Define Repeat Count list, such as literal or variable.
3. In the Repeat Count list, type the number of times you want OneWorld Scripting Tool to loop through the node during playback.
4. Click the Update button.

Script Retention and Reuse

OneWorld Scripting Tool allows you to do more than simply write scripts on a one-time basis. OneWorld Scripting Tool permits you to save, modify, reuse, combine, and send scripts. These capabilities are important because they broaden the scope of and audience for your tests. OneWorld Scripting Tool helps you to accomplish goals that are integral to building a system of scripts:

- Saving scripts, which you can reuse or modify
- Including scripts with other scripts to broaden the scope of testing
- Passing variables between scripts in a master script composed of a parent script and one or more children
- Sharing scripts

You can save scripts either on your local drive or in the OneWorld Scripting Tool script repository. When you build scripts by including one or more scripts with another, you can retrieve the scripts either from the local drive or from the repository. Scripts that you create by including scripts can pass variable values; you accomplish this by declaring variables as external and forging links between variables in separate scripts. Finally, you can send any script you create to colleagues using your e-mail tool.

This chapter covers the following tasks:

❑ Script saving

❑ Script includes

❑ Variable linking between scripts

❑ Script sharing

Script Saving

You can save scripts as you work by clicking the File option in the menu bar. You assign a name to the script, which is saved in the .ats directory. Save the file as a name that relates to the application you are testing. If you continue to work on the script, you can save as you work.

If your computer freezes or if OneWorld or OneWorld Scripting Tool fail while you are in a OneWorld Scripting Tool session, any scripts you have open automatically save. You can set the conditions under which OneWorld Scripting Tool auto-saves as you work using the Configure tab of the Options form.

See Also

- *Options for Configuring OneWorld Scripting Tool*

Script Includes

You can expand your testing scope by including one or more scripts on your local drive or in the script repository with a parent script. OneWorld Scripting Tool's Include command allows you to do this. OneWorld Scripting Tool creates a copy of the script you want to include and inserts it at the point in the open script that you have placed the insertion cursor. The included script becomes a child in a master script.

Whether you include a script you created on your local drive, or a script from the repository, OneWorld Scripting Tool automatically sends you to a form that includes all of the scripts that you have stored locally or to a form from which you can select one or more scripts from the repository. You choose the scripts to include, and OneWorld Scripting Tool inserts them as children of the master script. An Include command line contains the path to the included script, for example, [C:\atg\ats\UBE blind app.ats].

You sometimes must edit a script before you include it with another. For example, if you scripted data input in one script, and those data are also included in another, you must delete those data from the script before you write an Include command. If you do not, when OneWorld Scripting Tool plays back the included script, it loads data twice into a OneWorld form, which results in an error. You can open a script that has one or more scripts included in it and edit any of the included scripts. OneWorld Scripting Tool reloads the original script with the changes that you made to the included script.

Each time you write a command to include a script, OneWorld Scripting Tool displays a dialog box that asks if you want to continue script playback on include branch error. If you click yes on this box, you ensure that if an error occurs during the playback of the included script, OneWorld Scripting Tool reports the error, but continues on with playback of other included scripts, if there are any. This might be particularly useful if you are running very long scripts or batches of scripts.

See Also

- *Understanding the Script Repository*

Variable Linking between Scripts

When you include scripts with a parent script to create a longer script, you might also want to share variable values between the scripts. OneWorld Scripting Tool provides a mechanism to accomplish this called variable linking. In linking variables, you declare a variable in a parent script. In writing a script that you intend to include in the parent script, you also declare a variable, but if you want to link the variable, you declare it as external. OneWorld Scripting Tool allows you to link the externally declared variable to any variable you declared in the parent script. The link means that you can pass between scripts the value that you set for the variable.

To increase the versatility of your scripting, OneWorld Scripting Tool allows you to designate a default value for any variable that you designate as external. Doing so allows you to run an included script in stand-alone mode. OneWorld Scripting Tool uses the default value wherever the value is needed in the script.

You can link variables between locally generated scripts, reposited scripts, or you can link variables between a local script and a reposited script. In any case, you use the Tools option in the menu bar to run a script include. If the included script contains a variable that you have declared as external, OneWorld Scripting Tool prompts you to identify the variable to which you want to establish a link in the parent script.

This topic covers the following components of linking variables between scripts:

❑ External variables for script linking

❑ Default values for external variables

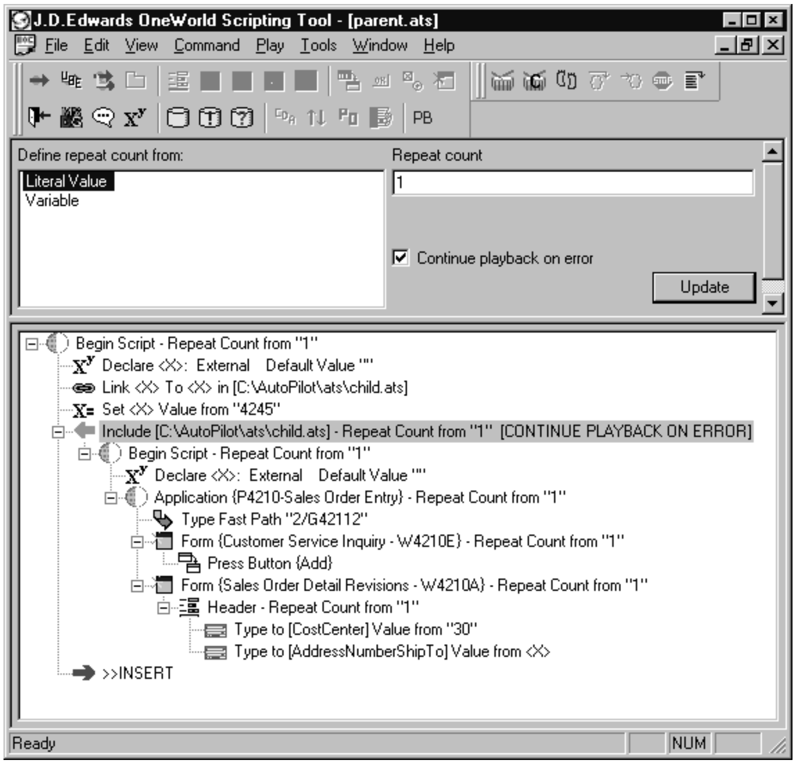
❑ Variable links

External Variables for Script Linking

You use external variables to help pass values between scripts. An external variable can receive a value from a variable in another script to which it is linked, or it can pass a value to a variable in another script. An option in the command pane allows you to designate a variable as external.

For example, you create Script A and declare a variable as X. You then create Script B, declare a variable as X, and designate it as external. Next, you include B with A. Script A is the parent script, Script B the child. OneWorld Scripting Tool asks that you link the externally declared variable to a variable in Script A. You link variable X to variable X.

So far you have simply provided the mechanism for passing a variable value from one script to another. However, suppose in Script A you set the value of variable X to an address number, such as 4245. With the link established, OneWorld Scripting Tool can now pass this value to Script B when you run the two scripts together.


J.D.Edwards OneWorld Scripting Tool - [parent.ats]
File Edit View Command Play Tools Window Help
PB
Update
Ready
NUM

Define repeat count from:
Literal Value
Variable

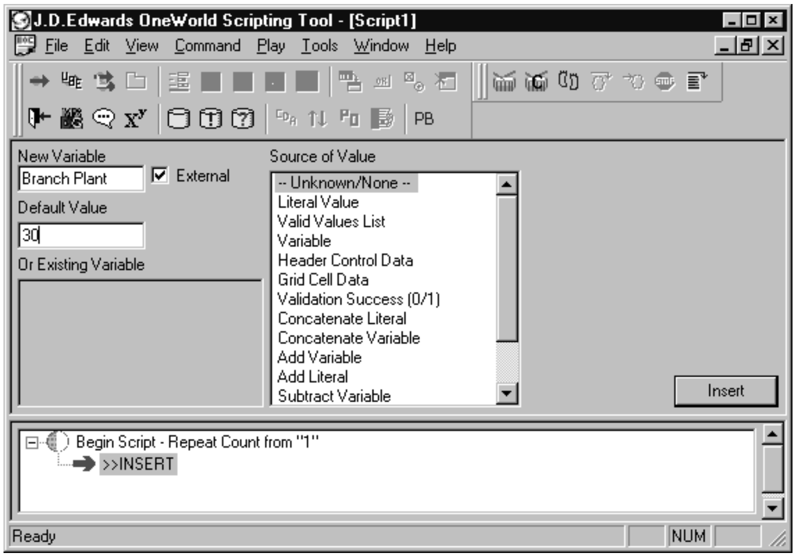
Repeat count: 1

☑ Continue playback on error

```
Begin Script - Repeat Count from "1"
  Declare <X>:  External   Default Value ""
  Link <X> To <X> in [C:\AutoPilot\ats\child.ats]
  Set <X> Value from "4245"
  Include [C:\AutoPilot\ats\child.ats] - Repeat Count from "1"  [CONTINUE PLAYBACK ON ERROR]
    Begin Script - Repeat Count from "1"
      Declare <X>:  External   Default Value ""
      Application {P4210-Sales Order Entry} - Repeat Count from "1"
        Type Fast Path "2/G42112"
        Form {Customer Service Inquiry - W4210E} - Repeat Count from "1"
          Press Button {Add}
        Form {Sales Order Detail Revisions - W4210A} - Repeat Count from "1"
          Header - Repeat Count from "1"
            Type to [CostCenter] Value from "30"
            Type to [AddressNumberShipTo] Value from <X>
  >>INSERT
```

Default Values for External Variables

You can declare a variable as external and assign it a default value. You give a variable a default value so that you can run a script in stand-alone mode. For example, you might write a script that tests one set of functions by itself. You might then include this script with one or more others. You can pass values between variables in the scripts by declaring a variable as external and linking it to a declared variable in the parent script. However, you might also want to have the ability to play back the original script in stand-alone mode. If you assign a default value to the variable and run the script in stand-alone mode, OneWorld Scripting Tool uses the default value to run the script. If you leave the default value of the external variable blank, OneWorld Scripting Tool reads the value as a null string.



New Variable: Branch Plant

☑ External

Default Value: 30

Or Existing Variable

Begin Script - Repeat Count from "1"
>>INSERT

When you create a script with an included script and linked variables, OneWorld Scripting Tool can pass a default value in the parent script to linked variables in the child script. However, if you assign a default value to an external variable in the child script, OneWorld Scripting Tool does not pass this value to the linked variable in the parent script. In this case, OneWorld Scripting Tool either overrides the default value with a value you set for the variable in the parent script, or it passes the value as a null string if you do not set a variable value in the parent script.

The following table summarizes three scenarios and the results that occur when you write scripts with default variable values:

External Variable Default Value Set In:	**Variable Link To**:	**Result**
Parent script	External variable in child (included) script	OneWorld Scripting Tool passes default value to linked variables in any child (included) scripts
Child script	Variable in parent script	OneWorld Scripting Tool overrides default value during playback, either with value set for variable in parent script or with null string if no value is set for variable in parent script
Stand-alone script	Not applicable	Variable with default value behaves as a local variable. OneWorld Scripting Tool uses default value wherever the script indicates

Variable Links

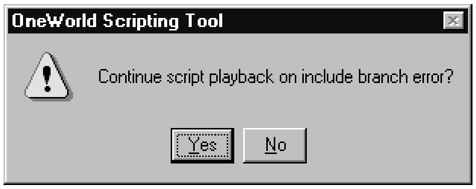
When you include with your parent script other scripts that contain external variables, you must define the links to each external variable so that OneWorld Scripting Tool can pass values between scripts. Defining the links allows OneWorld Scripting Tool to store data in a declared variable in the parent script or retrieve data from the parent script and reuse it in included scripts that contain external variables.

This topic discusses the following components of the linking variable process:

- ❑ Continue Script Playback on Include Branch Error dialogue box
- ❑ Link Variable form
- ❑ Recursive value searching
- ❑ Broken links

Continue Script Playback on Include Branch Error Dialogue Box

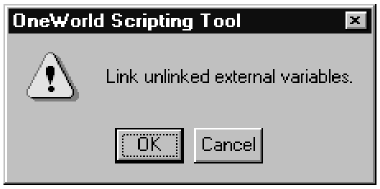
Before you run an Include command OneWorld Scripting Tool asks you if you want to continue the script playback on include branch error.



Continue script playback on include branch error?

Continuing the script on branch playback error means that if OneWorld Scripting Tool encounters an error in an included script during playback, playback moves on to the next included script rather than failing the entire playback session. Choosing Yes in this dialog box is recommended if you intend to test a long series of included scripts.

After you have indicated that you do or do not want to continue the script on an error, OneWorld Scripting Tool prompts you to link any unlinked external variables.

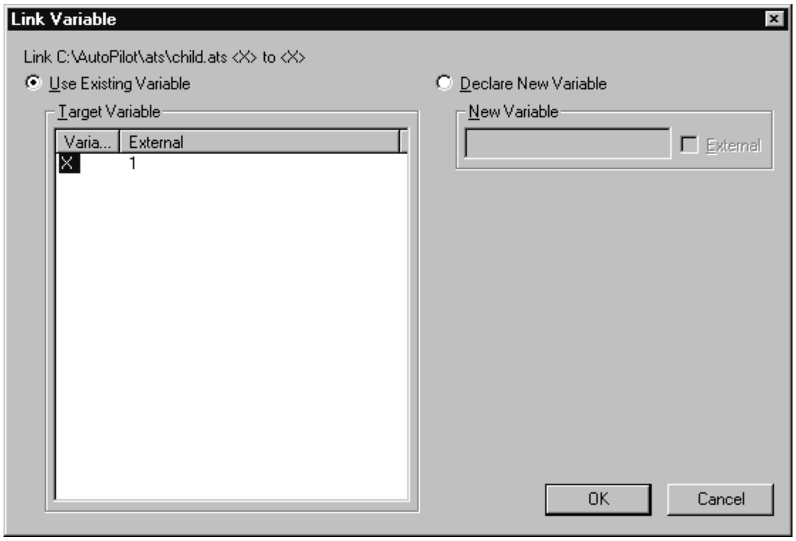


OneWorld Scripting Tool

Link unlinked external variables.

Link Variable Form

You use the Link Variable form to establish the link between the variable that you declared as external in your included script and a variable that you declared in the parent script.



Link Variable

Link C:\AutoPilot\ats\child.ats <X> to <X>

● Use Existing Variable ○ Declare New Variable

Target Variable

Varia...	External
X	1

New Variable

□ External

Note that the Link Variable form contains a link path that identifies the name of the included script and its externally declared variable, along with the name of the variable in the parent script that is highlighted in the field of the form. Hence, the link path: Link C:\atg\ats\child.ats <x> to <x> indicates that the included script is Script4.ats and that its externally declared variable <x> will be linked to the variable <x> in the parent script. If you want to change the link path, you click the name of a different variable from the parent script.

When you select a variable OneWorld Scripting Tool establishes the link and inserts a Link script object and command line in the script pane. OneWorld Scripting Tool inserts and maintains the link relationship in the parent script.

It is important to remember that if you click Cancel in the Link Variable form, and then attempt to save the script, OneWorld Scripting Tool continues to prompt you to supply the variable link. You can save the script without establishing the link; however, when you open it, the Variable Link form appears again.

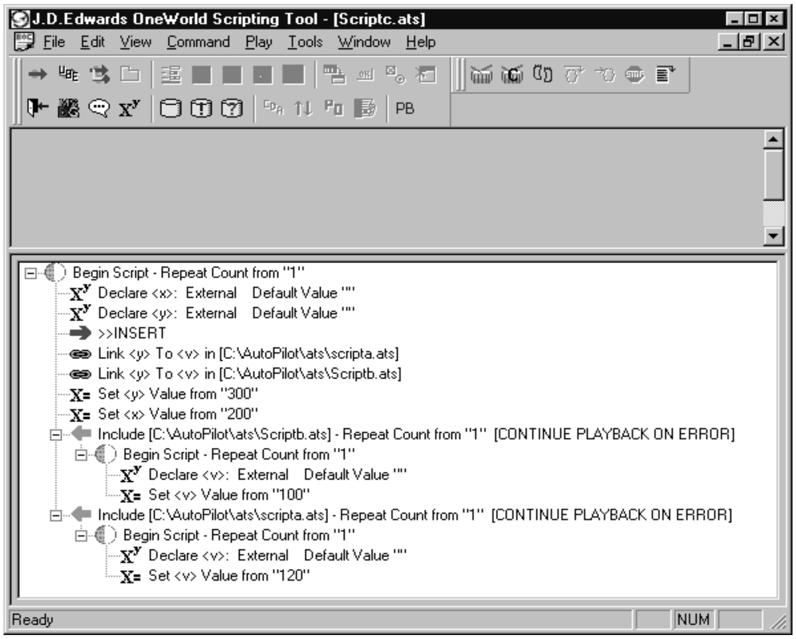
Recursive Value Searching

OneWorld Scripting Tool searches recursively for links, meaning that the search continues until it meets one of two specified conditions:

- The linked variable in the parent script is not external.
- The linked variable is in the master parent script, meaning that OneWorld Scripting Tool's search has reached the "top" of the script tree.

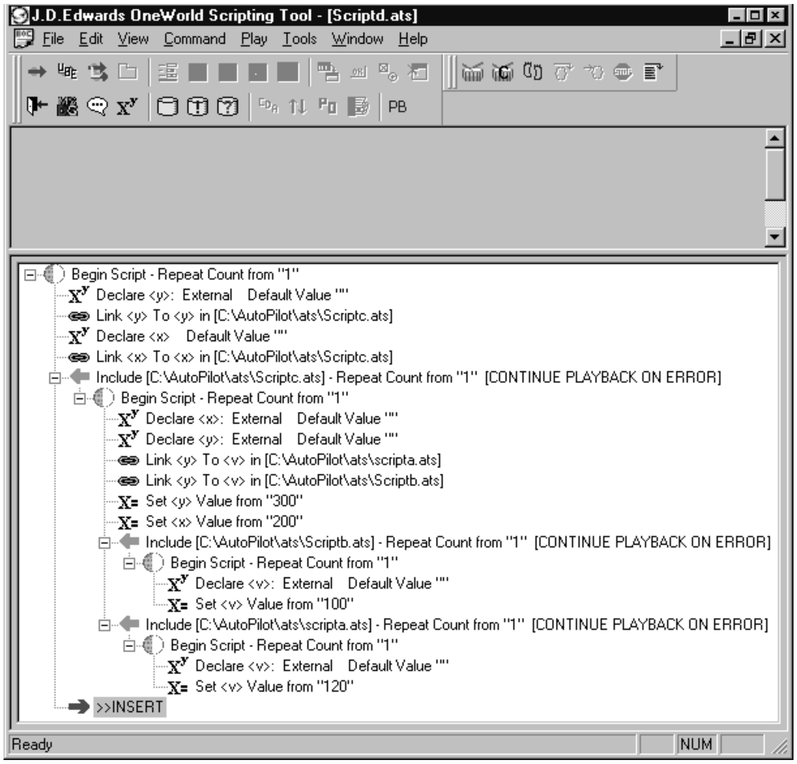
OneWorld Scripting Tool's ability to repeatedly search for a value permits you to establish links between variables in multiple script parent-child relationships. When you play back a script that has external variables, OneWorld Scripting Tool searches for the external variable's link in the parent script. When OneWorld Scripting Tool finds the link, the Link script object looks at the variable's Declare command. If the declared variable is not external, OneWorld Scripting Tool stops its search. However, if the declared variable is external, OneWorld Scripting Tool continues to look for links.

Suppose you create four separate scripts: A, B, C, and D. Each contains variables that you declare as external. You set the value of the variable in Script D, and then include Scripts C and D with B. When you run the Include command, OneWorld Scripting Tool asks you to create the links between the parent script (B) and the included scripts (C and D). When you run the script, OneWorld Scripting Tool searches for the link in the parent script. If you have not declared the linked variable in the parent script as external, the recursive process ceases.


J.D.Edwards OneWorld Scripting Tool - [Scriptc.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
    Declare <x>:  External   Default Value ""
    Declare <y>:  External   Default Value ""
    >>INSERT
    Link <y> To <v> in [C:\AutoPilot\ats\scripta.ats]
    Link <y> To <v> in [C:\AutoPilot\ats\Scriptb.ats]
    Set <y> Value from "300"
    Set <x> Value from "200"
    Include [C:\AutoPilot\ats\Scriptb.ats] - Repeat Count from "1"  [CONTINUE PLAYBACK ON ERROR]
        Begin Script - Repeat Count from "1"
            Declare <v>:  External   Default Value ""
            Set <v> Value from "100"
    Include [C:\AutoPilot\ats\scripta.ats] - Repeat Count from "1"  [CONTINUE PLAYBACK ON ERROR]
        Begin Script - Repeat Count from "1"
            Declare <v>:  External   Default Value ""
            Set <v> Value from "120"
```

Now, however, you decide to include Script B with Script A. Script A now becomes the master parent script. You now declare the variable in Script B as external and include it with Script A. When you run the script, OneWorld Scripting Tool searches for the link in the parent script. When OneWorld Scripting Tool reaches Script B, it reads the declared variable. Since that variable is external, it continues to search the tree until it reaches the master parent script, Script A.


J.D.Edwards OneWorld Scripting Tool - [Scriptd.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
    Declare <y>:  External    Default Value ""
    Link <y> To <y> in [C:\AutoPilot\ats\Scriptc.ats]
    Declare <x>    Default Value ""
    Link <x> To <x> in [C:\AutoPilot\ats\Scriptc.ats]
    Include [C:\AutoPilot\ats\Scriptc.ats] - Repeat Count from "1"  [CONTINUE PLAYBACK ON ERROR]
        Begin Script - Repeat Count from "1"
            Declare <x>:  External    Default Value ""
            Declare <y>:  External    Default Value ""
            Link <y> To <v> in [C:\AutoPilot\ats\scripta.ats]
            Link <y> To <v> in [C:\AutoPilot\ats\Scriptb.ats]
            Set <y> Value from "300"
            Set <x> Value from "200"
            Include [C:\AutoPilot\ats\Scriptb.ats] - Repeat Count from "1"  [CONTINUE PLAYBACK ON ERROR]
                Begin Script - Repeat Count from "1"
                    Declare <v>:  External    Default Value ""
                    Set <v> Value from "100"
            Include [C:\AutoPilot\ats\scripta.ats] - Repeat Count from "1"  [CONTINUE PLAYBACK ON ERROR]
                Begin Script - Repeat Count from "1"
                    Declare <v>:  External    Default Value ""
                    Set <v> Value from "120"
    >>INSERT
```

You can create as many parent-child relationships between scripts as you desire. Because it continues to search for values until it finds none, OneWorld Scripting Tool can maintain as many links as you desire.

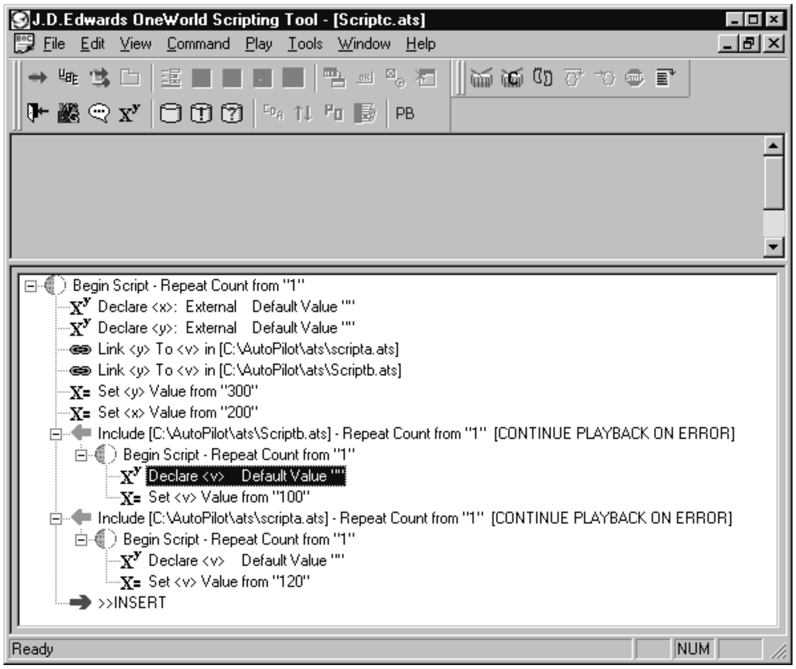
It is important to remember that the master parent script, meaning the script at the top of the tree, should not contain variables that you declare as external. These are unresolved variables, meaning that they have been declared as external, yet they have not been linked. Declaring these variables as external causes your script playback to fail. However, good scripting technique dictates that any external variable should always link to another variable in a parent script.

See Also

- *Indented Nodes*

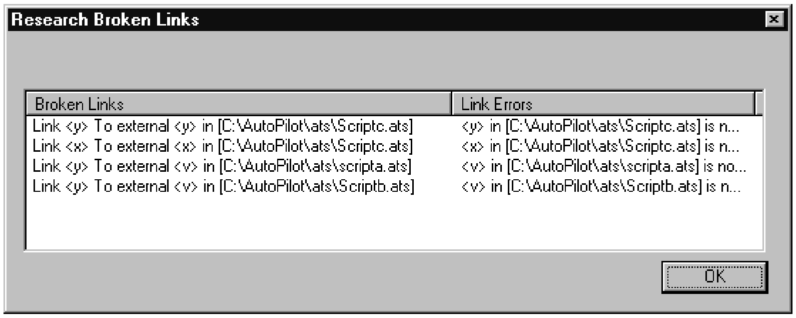
Broken Links

If you remove a variable's external attribute, you break its link to the link object, which can no longer find its reference. When you see a link object in a parent script, you should make sure that a variable in the child script is designated as external in the script pane. If that designation is not there, you have a broken link.


J.D.Edwards OneWorld Scripting Tool - [Scriptc.ats]
File Edit View Command Play Tools Window Help
PB

```
Begin Script - Repeat Count from "1"
    Declare <x>:  External   Default Value ""
    Declare <y>:  External   Default Value ""
    Link <y> To <v> in [C:\AutoPilot\ats\scripta.ats]
    Link <y> To <v> in [C:\AutoPilot\ats\Scriptb.ats]
    Set <y> Value from "300"
    Set <x> Value from "200"
    Include [C:\AutoPilot\ats\Scriptb.ats] - Repeat Count from "1"  [CONTINUE PLAYBACK ON ERROR]
        Begin Script - Repeat Count from "1"
            Declare <v>    Default Value ""
            Set <v> Value from "100"
    Include [C:\AutoPilot\ats\scripta.ats] - Repeat Count from "1"  [CONTINUE PLAYBACK ON ERROR]
        Begin Script - Repeat Count from "1"
            Declare <v>    Default Value ""
            Set <v> Value from "120"
    >>INSERT
```

OneWorld Scripting Tool notifies you that you have broken the link when you attempt to load the script containing the parent and the included scripts. The Research Broken Links form displays the broken links.



Broken Links	Link Errors
Link <y> To external <y> in [C:\AutoPilot\ats\Scriptc.ats]	<y> in [C:\AutoPilot\ats\Scriptc.ats] is n...
Link <x> To external <x> in [C:\AutoPilot\ats\Scriptc.ats]	<x> in [C:\AutoPilot\ats\Scriptc.ats] is n...
Link <y> To external <v> in [C:\AutoPilot\ats\scripta.ats]	<v> in [C:\AutoPilot\ats\scripta.ats] is no...
Link <y> To external <v> in [C:\AutoPilot\ats\Scriptb.ats]	<v> in [C:\AutoPilot\ats\Scriptb.ats] is n...

Note that the Research Broken Links form contains the name of the broken link path and a description of the link error. If you click the broken link in the form field, OneWorld Scripting Tool highlights the corresponding Link command line in the script pane.

You might break the links by inadvertently deleting the linked variable in the parent script or by deleting one of the included scripts before you have removed the links in the parent script. In either of these cases, you must repair the broken links.

Script Sharing

When you have created a script and modified it to your specifications, you can send it through e-mail to another person who might be interested in using it or including it with another script. Only people who have installed OneWorld Scripting Tool on their machines can run the script, however.

Reusing Scripts

After you have created and modified scripts in OneWorld Scripting Tool, you can save them and then reuse them as you need them. You can include scripts that you have created in other scripts that you are working on, and you can use e-mail to send saved scripts to other scripters and testers provided that the person with whom you are corresponding has OneWorld Scripting Tool installed.

With the ability to save and reuse scripts, you can cut the workload of colleagues who may be testing the same applications that you are. You accomplish these tasks using the OneWorld Scripting Tool menu bar and your computer's file directory.

OneWorld Scripting Tool also allows you to declare variables as external. This means that you can link the variable that you declare as external to a variable in another script and pass the value of the variable between scripts. Moreover, you can change the value of the variable to which you link the external. The ability to pass values between scripts and to change those values increases the versatility of your scripts, making them reusable and dynamic entities.

This chapter covers the following tasks involved in script reuse:

- ❏ Including scripts
- ❏ Creating variable links

Including Scripts

Including scripts allows you to broaden your testing scope. You can include one or more scripts with another, either from your local driver or from the script repository. Each time you include scripts with another script, you create a master script. The script that contains the included scripts is the parent, and the included scripts are its children. You can edit included scripts within the master, but any changes you make to an included script affects the master.

This topic covers the following tasks:

- Including one local script with another
- Including two or more local scripts in a new script
- Including a reposited script with another script

- Editing an included script

▶ To include one local script with another

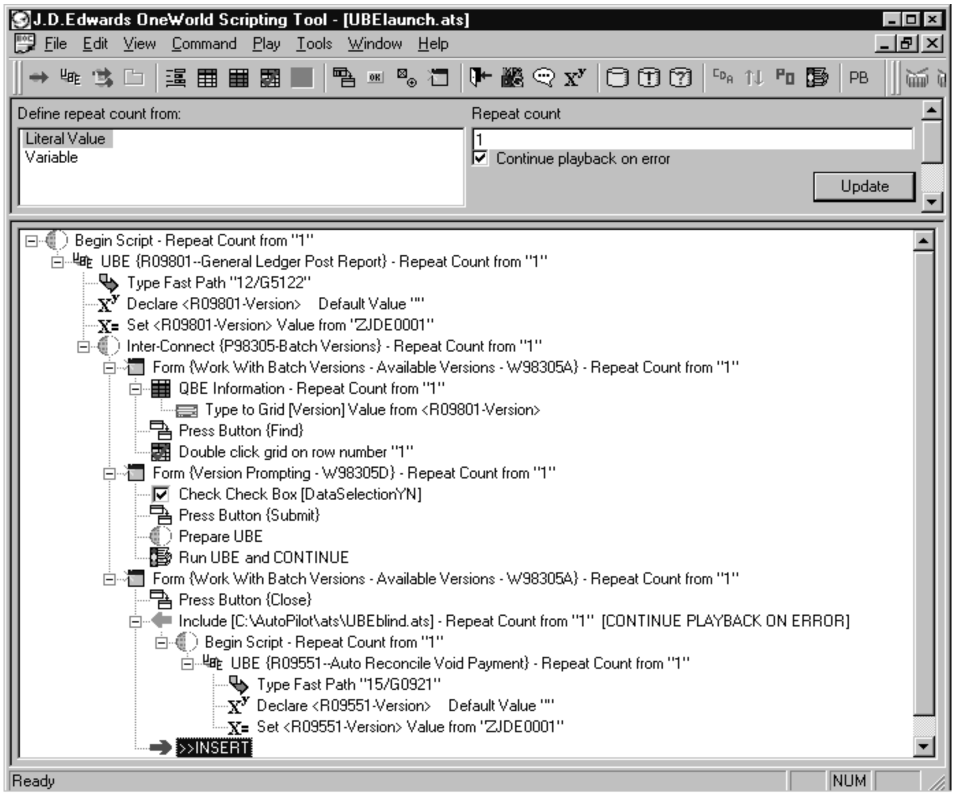
1. In the OneWorld Scripting Tool form, open a script.
2. In the script pane, place the insertion cursor at the point that you want OneWorld Scripting Tool to insert the included script.
3. In the menu bar, click Tools.
4. Click Include Local Script.
5. In the Select files to include form, click the name of the script you want to include.

   If you want to include more than one script, hold down either the Control key or the Shift Key and click another script. When you click the name of a script, its name appears in the File Name list.

6. Click Open.
7. In the Continue script playback on include branch error form, click Yes or No.

   OneWorld Scripting Tool inserts the script you chose to the script pane of the open script at the point of the insertion cursor. OneWorld Scripting Tool inserts a “Continue Playback” or “Fail Playback” message on the Include command line.

   You can change the message from “Continue Playback” to “Fail Playback” by clicking the command line and unchecking the Continue playback on error option in the command pane.


J.D.Edwards OneWorld Scripting Tool - [UBElaunch.ats]
File Edit View Command Play Tools Window Help
PB
Update
Ready
NUM

Define repeat count from:
Literal Value
Variable

Repeat count
1
☑ Continue playback on error

```
Begin Script - Repeat Count from "1"
  UBE {R09801--General Ledger Post Report} - Repeat Count from "1"
    Type Fast Path "12/G5122"
    Declare <R09801-Version>    Default Value ""
    Set <R09801-Version> Value from "ZJDE0001"
    Inter-Connect {P98305-Batch Versions} - Repeat Count from "1"
      Form {Work With Batch Versions - Available Versions - W98305A} - Repeat Count from "1"
        QBE Information - Repeat Count from "1"
          Type to Grid [Version] Value from <R09801-Version>
        Press Button {Find}
        Double click grid on row number "1"
      Form {Version Prompting - W98305D} - Repeat Count from "1"
        Check Check Box [DataSelectionYN]
        Press Button {Submit}
        Prepare UBE
        Run UBE and CONTINUE
      Form {Work With Batch Versions - Available Versions - W98305A} - Repeat Count from "1"
        Press Button {Close}
        Include [C:\AutoPilot\ats\UBEblind.ats] - Repeat Count from "1"  [CONTINUE PLAYBACK ON ERROR]
          Begin Script - Repeat Count from "1"
            UBE {R09551--Auto Reconcile Void Payment} - Repeat Count from "1"
              Type Fast Path "15/G0921"
              Declare <R09551-Version>    Default Value ""
              Set <R09551-Version> Value from "ZJDE0001"
        >>INSERT
```

8. Click File in the menu bar.
9. Click Save As.
10. In the Save As form, type in a new file name for the script.
11. Click Save.

▶ To include two or more local scripts in a new script

1. In the menu bar of the OneWorld Scripting Tool form, click File.
2. Click New.

   A new OneWorld Scripting Tool form appears.

3. Click Tools.
4. Click Include Local Script.
5. In the Select files to include form, click the name of the first script to include.
6. Hold down the Control key.
7. Click the name of the second script to include.

   OneWorld Scripting Tool highlights the names of both scripts.

8. Click Open.
9. In the Continue script playback on include branch error form, click Yes or No.

   OneWorld Scripting Tool inserts the two scripts into the script pane of the new script.

10. In the menu bar, click File.
11. Click Save As.
12. In the Save As form, type in a new file name for the script.
13. Click Save.

    You can insert additional commands to the new script if you need to.

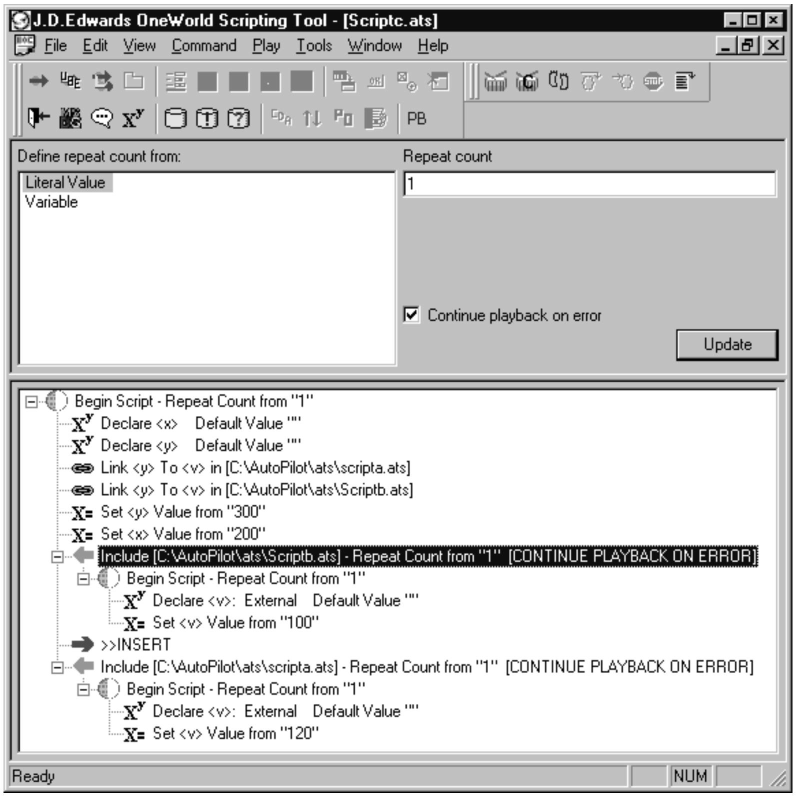
▶ To include a reposited script with another script

1. In the OneWorld Scripting Tool form, open a script.
2. In the script pane, position the insertion cursor at the point you want to include a script.
3. Click Tools.
4. Choose Include Reposited Script.
5. In the Include Repository Script form, click the name of one or more script titles.
6. Click the Include button.
7. In the Continue script playback on include branch error form, click Yes or No.

   OneWorld Scripting Tool places the included script in the script pane of the open script at the point of the insertion cursor.

▶ To edit an included script

1. In the script pane of the OneWorld Scripting Tool form, select the Include command line of the script that you want to edit.



J.D.Edwards OneWorld Scripting Tool - [Scriptc.ats]
File Edit View Command Play Tools Window Help
PB
Update
Ready
NUM

Define repeat count from:

Literal Value
Variable

Repeat count: 1

☑ Continue playback on error

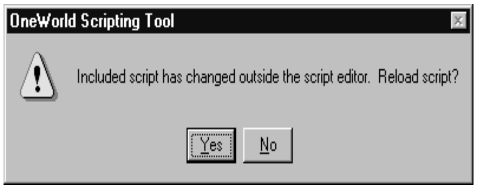
```
Begin Script - Repeat Count from "1"
    Declare <x>   Default Value ""
    Declare <y>   Default Value ""
    Link <y> To <v> in [C:\AutoPilot\ats\scripta.ats]
    Link <y> To <v> in [C:\AutoPilot\ats\Scriptb.ats]
    Set <y> Value from "300"
    Set <x> Value from "200"
    Include [C:\AutoPilot\ats\Scriptb.ats] - Repeat Count from "1"  [CONTINUE PLAYBACK ON ERROR]
        Begin Script - Repeat Count from "1"
            Declare <v>:  External   Default Value ""
            Set <v> Value from "100"
    >>INSERT
    Include [C:\AutoPilot\ats\scripta.ats] - Repeat Count from "1"  [CONTINUE PLAYBACK ON ERROR]
        Begin Script - Repeat Count from "1"
            Declare <v>:  External   Default Value ""
            Set <v> Value from "120"
```

2. Right click and choose Edit.

   Caution: OneWorld Scripting Tool opens the included script. Note that the included script might be a parent of other scripts. To edit a child of the script you are editing, you must select it and choose Edit again.

3. Perform the necessary edits to the included script and choose Close.
4. Save the changes to the included script.

   OneWorld Scripting Tool displays a form asking if you want to reload the script.



Included script has changed outside the script editor. Reload script?

5. To load the changes to the included script, click Yes.

   OneWorld Scripting Tool reloads the parent script with the changes to the included script.

6. Save the changes to the master script.

Creating Variable Links

When you create variable links, you write the commands that make it possible for OneWorld Scripting Tool to pass variable values between two or more scripts. You declare a variable and set its value just as you do in a stand-alone script. However, by declaring the variable as external, you indicate that it can be linked to a variable in another script. OneWorld Scripting Tool uses links to pass a value you set in one variable in one script to another script.

Taken as a whole, the process of forging variable links requires that you write two or more scripts, declare certain variables as external, set a variable value, and then create the links between the variables. You can link variables between scripts that you maintain locally or between reposited scripts. Finally, if you break links between variables, you must reforge the links for your script to run.

- Declaring a variable as external
- Assigning a default value to an external variable
- Linking variables between local scripts
- Linking variables between reposited scripts
- Researching and repairing broken links

Declaring a Variable as External

You declare a variable as external only if you want to link it to a variable in another script for the purpose of passing a value between the scripts. You declare an external variable just as you declare a variable in a stand-alone script. OneWorld Scripting Tool offers an option that you check to declare the variable as external. Note that you can also write a script with a local variable, but later update the variable and make it external. You might do this when you want to include the script with another script.

▶ To declare a variable as external

1. In the command menu of the OneWorld Scripting Tool form, click Variables.
2. Type the name of a new variable in the unpopulated New Variable list or choose a previously declared variable from the Existing Variable list.
3. Choose the External option.
4. If you are declaring the variable but not setting a value, choose Unknown/None from the Source of Value list and Click the Insert button.
5. If you are setting a value for the variable, choose from the Source of Value list and the value selection list.
6. Click the Insert button.

Note that OneWorld Scripting Tool includes the word “External” in the Declare command line in the script pane.

See Also

- *The Source of Input List*
- *Using a Variable as a Source of Input*

Assigning a Default Value to an External Variable

You can set a default value for a variable, whether or not you choose to declare it as external. By assigning a default value, you ensure that you can run your script in stand-alone mode, even if you make it external and link it to a variable in another script. Remember that if you do link the variable with the default value to a variable in a parent script, OneWorld Scripting Tool overrides the default value during playback, either with the value of the variable in the parent script, or with a null string if the variable in the parent script has no value.

▶ To assign a default value to an external variable

1. In the command menu of the OneWorld Scripting Tool form, click Variables.
2. In the OneWorld Scripting Tool command pane, type the name of a variable to the New Variable list or choose a variable from the Existing Variable list.
3. Click the External option.
4. In the Default list, type a value for the variable.
5. Click the Insert button.

Linking Variables between Local Scripts

You can establish links between scripts that you generate locally. To do so you write a parent script with one or more declared variables, then one or more scripts that you want to include with the parent. The scripts to be included must contain externally declared variables if you want to establish links. You then open the parent script and choose the scripts to be included from your .ats directory. The link path in the parent script points to the scripts from your .ats directory that you have included as children in your script.

▶ To link variables between local scripts

1. In the menu bar of the OneWorld Scripting Tool form, with a script open, click Tools.
2. Click Include Local Script.

3. In the Select files to include form, click the name of the script you want to include.

   If you want to include more than one script, hold down the Control key and click another script. When you click the name of a script, its name appears in the File Name list.

4. Click Open.
5. In the Continue script playback on include branch error dialogue box, click Yes or No.
6. In the dialogue box asking you to link unlinked external variables, click OK.

   Note: You are asked to link unlinked variables only if you have declared variables as external in one or more of the scripts that you are including with a parent.

7. In the Link Variables form, choose the name of the declared variable in the parent script that you want to link to the external variable in the included script.
8. Click the OK button.
9. Repeat steps 6 and 7 as many times as necessary to establish links for all external variables.

Linking Variables between Reposited Scripts

OneWorld Scripting Tool also allows you to link variables between reposited scripts or between local and reposited scripts. The reposited script that you want to include with a parent script must again contain a variable you declared as external. You open the parent script and choose the scripts to included from the Include Repository Script form. The link path in the parent script points to scripts from the repository that you have included as children and contains the script's name rather than the path that you see when you include a local script.

▶ To link variables between reposited scripts

1. In the menu bar of the OneWorld Scripting Tool form, click Tools.
2. Choose Include Reposited Script.
3. In the Include Repository Script form, click the name of one or more script titles.
4. Click the Include button.
5. In the Continue script playback on include branch error dialogue box, click Yes or No.
6. In the dialogue box asking you to link unlinked external variables, click OK.

7. In the Link Variables form, choose the name of the declared variable in the parent script that you want to link to the external variable in the included script.
8. Click the OK button.
9. Repeat steps 7 and 8 as many times as necessary to establish links for all external variables.

See Also

- *Including Scripts*
- *Understanding the Script Repository*

Researching and Repairing Broken Links

You forge a variable link between an external variable and a declared variable in a parent script. Once you have forged that link, you must maintain it if you want OneWorld Scripting Tool to pass variable values between scripts. If you break the link between an external variable and a declared variable in the parent script, OneWorld Scripting Tool asks you to research the broken links and to restore them, using the Research Broken Links form.

▶ To research and repair broken links

1. In the Research Broken Links form, note in the Broken Links and Link Error headings the path to the script that contains the broken link(s).
2. Under the same headings, note the variables whose links you have broken.
3. Click OK.
4. Open the script that contains the broken link(s).
5. In the script pane, find each variable whose link you have broken.
6. Click the Declare command line.
7. In the command pane, choose the External option.
8. Click the Insert button.
9. Once you have declared as external each variable whose link was broken, save and close the script.

Play Back

Playing Back the Script

You play back your script to see if it can run the specified commands without error. You can play back the script at any point, whether or not you are finished writing commands.

OneWorld Scripting Tool offers different options for playing back the script:

- From the beginning to the end without interruption
- From a chosen cursor position to the end without interruption
- From the beginning of a chosen script branch (node) to the end of the branch
- From a chosen cursor position line by line
- From a chosen position to a designated breakpoint

You can stop playback at any time by clicking Stop in the play menu of the menu bar or the Stop button on the cool bar.

You can script waiting times during playback. When you script a wait period, the playback process halts for a length of time that you determine, then resumes. You can also pause playback indefinitely, resuming it only when you press the Pause key on the keyboard.

You can configure playback by choosing Options from the Tools menu and clicking the Playback tab. For example, you can set OneWorld Scripting Tool to capture and display the results of a playback session. The results are presented as an event stream: a time-stamped, chronological record of each OneWorld Scripting Tool and OneWorld event that occurred during the session.

When you play the script, OneWorld Scripting Tool stops the playback if a OneWorld error occurs. If you have configured playback to save and display test results, the message Script playback was not successful displays in the Test Results form. If OneWorld Scripting Tool encounters no errors during playback, it displays the message Script playback completed successfully.

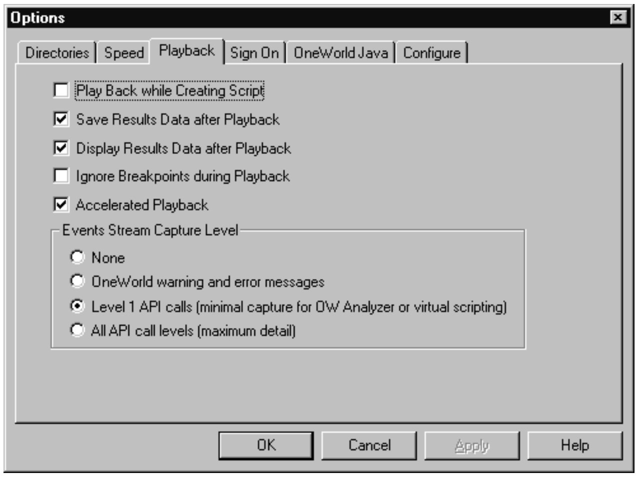
This section covers the following topics:

❑ Understanding script playback functions

❑ Running script playback

Understanding Script Playback Functions

OneWorld Scripting Tool allows you to handle certain playback tasks automatically and to run playback manually. You can set up automatic script playback features. For example, you can specify that OneWorld Scripting Tool is to automatically capture and save the results of each script playback session.



□ Play Back while Creating Script
☑ Save Results Data after Playback
☑ Display Results Data after Playback
□ Ignore Breakpoints during Playback
☑ Accelerated Playback

Events Stream Capture Level
○ None
○ OneWorld warning and error messages
● Level 1 API calls (minimal capture for OW Analyzer or virtual scripting)
○ All API call levels (maximum detail)

You manually control the playback using eight buttons on the cool bar and the Pause key on the keyboard. Each cool bar button represents a specific playback function. You can also find each option in the Play menu. The table below describes the manual playback functions:

Playback Function	**Play/Command menu text**
Play back the entire script from the top	Play from Top (Play menu)
Play back the script from a chosen cursor position to the end of the script	Play from Cursor (Play menu)
Play back a selected branch of the script	Play Branch (Play menu)

Play from a chosen script command	Stepping on (Play menu)
Play to the next command	Step Next (Play menu)
Continue playback from a chosen spot in the script to a predetermined breakpoint after you have chosen to step on the script	Continue to Breakpoint (Play menu)
Stop playback	Stop Playback (Play menu)
Ignore breakpoints during playback of the OneWorld Scripting Tool script	Ignore Breakpoints during Playback (Play menu)
Insert a wait period to occur during script playback	Comment/Wait (Command menu)
Insert a comment to the script	Comment/Wait (Command menu)
Pause playback	N/A; press Pause button on keyboard

This chapter explains the two components of OneWorld Scripting Tool's script playback capability:

❑ Automatic script playback configuration

❑ Manual script playback options

Automatic Script Playback Configuration

You can set or change your script playback configuration at any time to let OneWorld Scripting Tool handle certain playback features without your intervention. To configure script playback, you use the Options form, which you access from the Tools menu in the AutoPilot form. The playback configuration features are:

❑ Playback during script creation

❑ Storage and display of playback data

❑ Breakpoint handling

❑ Playback speed

❑ Event stream

Playback during Script Creation

When you choose the Playback during Script Creation option, OneWorld Scripting Tool runs each command in OneWorld after you insert it to the script. If you want to write your script without any delay caused by script playback, you should not choose this option. On the other hand, if you want to observe each command that you script as it runs in OneWorld, you choose this command.

Storage and Display of Playback Data

OneWorld Scripting Tool allows you to store and to display the results of each OneWorld Scripting Tool script playback. If you choose to save results data, OneWorld Scripting Tool saves script playback results as a binary large object in the OneWorld F98214 table. If you choose to display test results after playback, OneWorld Scripting Tool automatically displays the Test Results form after playback. This form contains tabs that you use to get further information about script playback.

See Also

- *Understanding Script Reporting*
- *Options for Configuring OneWorld Scripting Tool*

Breakpoint Handling

A breakpoint is a point that you toggle in the script that halts playback until you manually continue it or cancel it. If you want to play the script uninterrupted, but keep any breakpoints you have toggled, you click the Ignore Breakpoints during Playback option. Doing so means that you do not have to toggle off breakpoints, and then toggle them back on. When you want to play the script with the breakpoints in effect, you simply turn off the option.

Playback Speed

If you choose the Accelerated Playback option, code in OneWorld delivers notification to OneWorld Scripting Tool as soon as processing is complete so that playback can immediately continue. In general, you should choose this option only when you are running relatively simple scripts that do not require OneWorld to do a large amount of processing.

Event Stream

The term "event stream" refers to the flow of information from OneWorld to OneWorld Scripting Tool that occurs during playback. You click one of four options to configure script playback to capture this information:

- None
- OneWorld warning and error messages
- Level 1 API calls
- All API call levels

See Also

- *OneWorld Virtual User Tool Guide*
- *OneWorld Analyzer Tool Guide*

Manual Script Playback Options

After you have configured script playback, you run playback using the command menu or the playback buttons on the cool bar. Script playback options allow you to manually control the way your script plays back. You can play the script from the top without interruption, play the script from any chosen any spot to the end, play only a chosen branch of the script, play back the script one command at a time, play it to a breakpoint, or stop playback. You can insert wait periods in the script to delay playback a set period of time before it resumes, or you can manually pause and resume script playback.

You can also insert comments that give information about the script. These comments help to explain the purpose of the script.

This topic describes the following script playback execution functions:

- Play from top
- Play from cursor to end of script
- Play a chosen branch of the script
- Play from a chosen script line command
- Play to the next script line command
- Continue to breakpoint
- Toggle a breakpoint
- Wait before proceeding

- Script comment
- Stop playback
- Pause Playback
- Ignore breakpoints during playback

Play from Top

When you click Play from Top in the play menu, script playback begins with the first command line and continues until the end of the script unless OneWorld Scripting Tool encounters an error.

When you use this or any of the other playback functions, you can stop the playback yourself by clicking the Stop button on the cool bar or by clicking the Pause button on your keyboard and then clicking the Stop button. Remember also to remove any breakpoints you have inserted to the script before you try to play back the entire script.

Play from Cursor to End of Script

The Play from Cursor to End of Script command allows you to choose any spot in the script, and then play the script to completion, provided that OneWorld Scripting Tool does not encounter an error or a breakpoint you have inserted to the script.

Play a Chosen Branch of the Script

The Play Selected Branch command lets you play only a single script node consisting of one or more context commands and a series of action commands. This command allows you to isolate a single section of the script to observe playback.

Play from a Chosen Script Line Command

To manually control playback from a chosen point in the script, you choose the Stepping On command. You can choose to play back either from the top, from a chosen cursor position, or from a chosen branch of the script. You can then choose how you want to play back the script, either one command line at a time or to a chosen breakpoint.

Play to the Next Script Line Command

After you have clicked Stepping on and chosen to play back either from the top, from a chosen cursor position, or from a chosen branch, you choose the Step Next command to play the script one command line at a time. Script playback does not proceed until you click the Step Next button on the cool bar.

Toggle a Breakpoint

A breakpoint is a command line you choose for playback to halt until you resume or cancel playback. You can insert as many breakpoints in the script as you desire. Doing so gives you another way to isolate areas of the script and observe the playback.

You toggle the breakpoint by placing the insertion cursor after the line in the script where you want playback to break, right-clicking the mouse, and choosing Toggle Breakpoint. You can script multiple playback breakpoints. The breakpoint itself does not stop playback. You can only do so by clicking the Stop button.

Continue to Breakpoint

After you click Stepping On and choose a position from which to play the script, you can play back the script to a breakpoint. OneWorld Scripting Tool plays all commands until it reaches the breakpoint.

Wait Before Proceeding

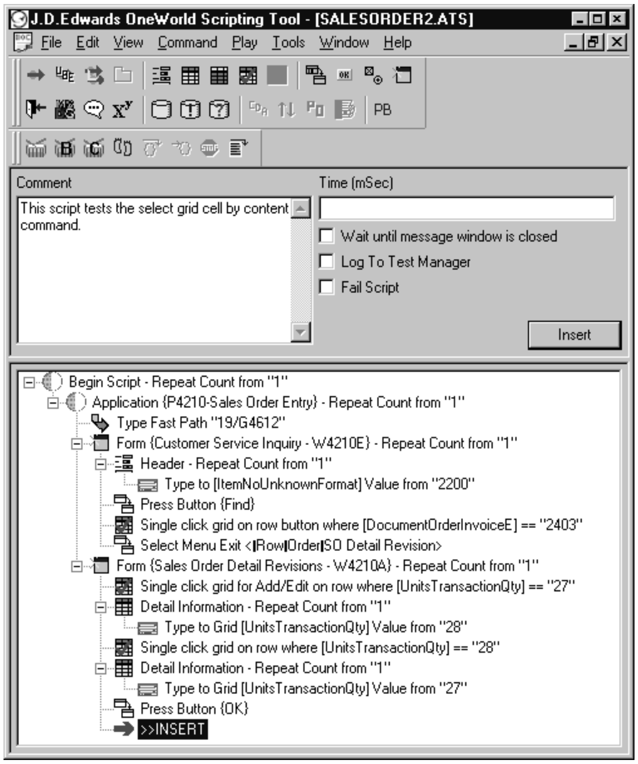
You use the Wait/Comment command to script waiting periods, or pauses, in the playback. You can insert one or more wait commands in the script pane at any point you want. After the prescribed wait period has elapsed, playback resumes without your intervention. You can specify the duration of the wait. You can insert to the script waits of sufficient duration to simulate the real time that it might take to actually perform data entry to a header or detail area.

A Wait command produces a pause of determinate length. By contrast, when you press the Pause button on the keyboard, OneWorld Scripting Tool pauses playback until you click the Pause button again.

Script Comment

Using the Wait/Comment command, you can write brief comments that might, for example, explain the reason that you inserted a wait. If you exchange scripts with a colleague, you can use the comments to explain what occurred at a particular point in the script or to explain what the script is designed to test. You type your comment into an unpopulated Comment list in the command pane and insert it to the script.

> **Note:** OneWorld Scripting Tool truncates the comment in the script pane at 54 characters, counting spaces.


J.D.Edwards OneWorld Scripting Tool - [SALESORDER2.ATS]
File Edit View Command Play Tools Window Help
PB
Insert

Comment

This script tests the select grid cell by content command.

Time (mSec)

□ Wait until message window is closed

□ Log To Test Manager

□ Fail Script

```
⊟ Begin Script - Repeat Count from "1"
  ⊟ Application {P4210-Sales Order Entry} - Repeat Count from "1"
      Type Fast Path "19/G4612"
    ⊟ Form {Customer Service Inquiry - W4210E} - Repeat Count from "1"
      ⊟ Header - Repeat Count from "1"
          Type to [ItemNoUnknownFormat] Value from "2200"
        Press Button {Find}
        Single click grid on row button where [DocumentOrderInvoiceE] == "2403"
        Select Menu Exit <|Row|Order|SO Detail Revision>
    ⊟ Form {Sales Order Detail Revisions - W4210A} - Repeat Count from "1"
        Single click grid for Add/Edit on row where [UnitsTransactionQty] == "27"
      ⊟ Detail Information - Repeat Count from "1"
          Type to Grid [UnitsTransactionQty] Value from "28"
        Single click grid on row where [UnitsTransactionQty] == "28"
      ⊟ Detail Information - Repeat Count from "1"
          Type to Grid [UnitsTransactionQty] Value from "27"
        Press Button {OK}
        >>INSERT
```

OneWorld Scripting Tool also allows you to cut or copy comments from other scripts or from other documents and paste them into the Comment list of the command pane. This might prove useful if, for example, a comment you insert in one script is applicable to several other scripts.

When you click the Comment/Wait button, the command pane also displays three options:

- Wait until message window is closed
- Log to Test Manager
- Fail script

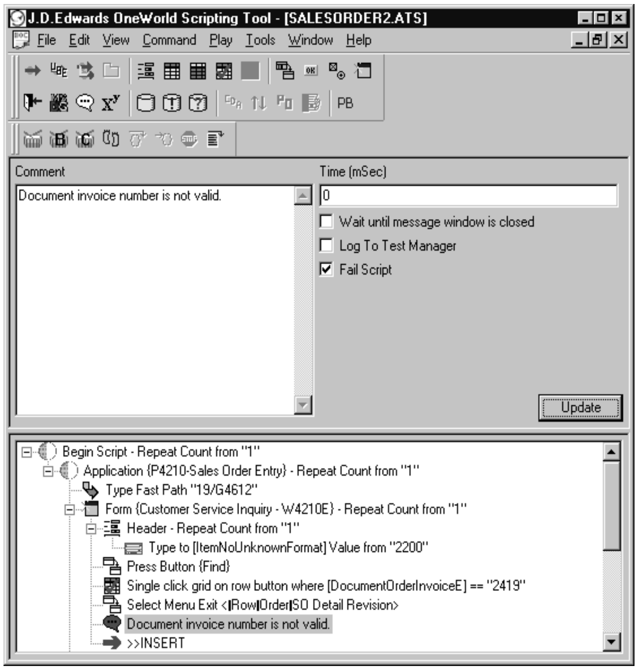
Choose the Wait until message window is closed option if your script presses the Delete button in a OneWorld form. If you choose the option, OneWorld Scripting Tool does not proceed with script playback until it has clicked OK on the OneWorld Confirm Delete form.

Choose the Log to Test Manager option if you plan to include a script as part of batch testing. You use the Test Manager tool to assemble script playlists for batch playback. If you choose the Log to Test Manager option, the comments you insert in a script are sent to Test Manager and included in a report after playback.

For more information on Test Manager, see *OneWorld Scripting Tool Test Manager* and *Managing Script Testing* in this guide.

Choose the Fail script option if a critical event in your script caused the script to fail. If you choose this option, OneWorld Scripting Tool automatically fails the script at the point you insert the command

If you choose to fail the script, OneWorld Scripting Tool inserts a comment symbol in red.


J.D.Edwards OneWorld Scripting Tool - [SALESORDER2.ATS]
File Edit View Command Play Tools Window Help
PB
Update

Comment: Document invoice number is not valid.

Time (mSec): 0

☐ Wait until message window is closed
☐ Log To Test Manager
☑ Fail Script

```
Begin Script - Repeat Count from "1"
  Application {P4210-Sales Order Entry} - Repeat Count from "1"
    Type Fast Path "19/G4612"
    Form {Customer Service Inquiry - W4210E} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [ItemNoUnknownFormat] Value from "2200"
      Press Button {Find}
      Single click grid on row button where [DocumentOrderInvoiceE] == "2419"
      Select Menu Exit <|Row|Order|SO Detail Revision>
      Document invoice number is not valid.
      >>INSERT
```

You can use the Fail script option in conjunction with logging comments to Test Manager by choosing both options. OneWorld Scripting Tool fails the script and writes any comments you include on the failure to Test Manager’s summary report when you run a batch test.

Stop Playback

At any point during playback, you can stop the process by clicking Stop in the play menu or the Stop button on the cool bar. If you turn on a playback function, the Stop button turns red. When you click the Stop button, OneWorld Scripting Tool displays a Script Playback Cancelled message, and the Stop button returns to gray.

Pause Playback

You can use the Pause button on your keyboard to gain an added measure of control over script playback. You can use the Pause button to control script playback from the top or from any chosen cursor position.

Pressing the Pause button during script playback pauses playback, but only until you choose to continue or to stop playback altogether. When you press the Pause button a second time, OneWorld Scripting Tool runs the next command only. To proceed to the next command, you press the pause button again. You can continue in this way until you reach the end of the script or you decide that you want to stop playback. Using the Pause button in this way allows you to run your playback step by step without having to use the mouse.

OneWorld Scripting Tool must have control of the screen when you press the Pause button. To make sure that OneWorld Scripting Tool has control of the screen, simply position the cursor anywhere in the OneWorld Scripting Tool form and click the mouse.

Ignore Breakpoints during Playback

You might want to preserve breakpoints that you have toggled in the script, but run the script one or more times without breakpoints. Rather than toggle the breakpoints on and off, you can click the Ignore Breakpoints during Playback button. When you do so, you can play back the script from the top, and OneWorld Scripting Tool ignores the breakpoints. When you want to run the script to the breakpoint you have designated, simply turn off the Ignore Breakpoints during Playback button and play back the script.

Running Script Playback

You run the various script playback functions in OneWorld Scripting Tool using the six choices in the play menu or the six playback buttons on the cool bar. You can also use the Wait/Comment command to script one or more pauses in the playback and to insert comments into selected command lines in the script or as stand-alone lines. Finally, you can use the mouse to right-click a command line in the script pane and to toggle a breakpoint on or off.

This chapter describes how to run the following playback functions:

- ❑ Playing the script from the top
- ❑ Playing the script from a chosen cursor position
- ❑ Playing the script from a chosen command line
- ❑ Playing the script to the next command
- ❑ Pausing playback
- ❑ Toggling a breakpoint
- ❑ Playing the script to a breakpoint
- ❑ Continuing playback to a breakpoint
- ❑ Ignoring breakpoints in the script
- ❑ Inserting a Wait command in the script
- ❑ Inserting a comment in the script
- ❑ Failing a script
- ❑ Setting transaction times in the script

Playing the Script from the Top

You use the Play from Top function if you want to see if a script can play continuously from beginning to end. Before you play the script from the top, be sure to close all open OneWorld applications. If you do not do so, playback might not complete successfully. OneWorld Scripting Tool expects to control the mouse during the playback process. Do not move the mouse or attempt to open any applications or programs while playback is proceeding. Also be sure to turn off the Stepping On function.

▶ To play the entire script from the top

1. In the script pane of the OneWorld Scripting Tool form, make sure you have removed any breakpoints, that you have turned off the Stepping On button, and that you have closed all OneWorld windows.
2. Click Play in the menu bar and choose Play from Top from the drop-down menu.

   Note: You can also start the playback process by pressing the F5 key on the keyboard. This key functions as a hot key.

Playing the Script from a Chosen Cursor Position

You use the Play from Cursor function if you want to choose a position below the top of the script from which you want to play back. Make sure you have turned off the Stepping On button and that you have closed all OneWorld windows before you run this playback function.

▶ To play back the script from a chosen cursor position

1. In the script pane in the OneWorld Scripting Tool form, choose a command line in the script from which you want to play back.
2. click the command line to highlight it.
3. Click Play from Cursor in the play menu.

Playing the Script from a Chosen Command Line

You can use the Stepping On button to set up playing back the script one line at a time or from breakpoint to breakpoint. Clicking the Stepping On button is a preliminary step that you must take to play back the script one line at a time or to play back from breakpoint to breakpoint.

▶ To play from a chosen script command

1. In the play menu of the OneWorld Scripting Tool form, click Stepping On.
2. Click Play from Cursor.

Playing the Script to the Next Command

After you have clicked the Stepping On button, you choose a command line from which to play the script back, either from the top, from a branch of the script, or from another chosen cursor position. You can then play the script one line at a time or from breakpoint to breakpoint.

▶ To play the script back one line at a time

1. In the script pane in the OneWorld Scripting Tool form, choose a command line in the script from which you want to play back.
2. In the script pane, click the command line.
3. In the play menu, click Stepping On.
4. On the cool bar, click of the following buttons:
   - Play from Top
   - Play Branch
   - Play from Cursor

   OneWorld Scripting Tool enables the Step Next, Continue to Breakpoint and Stop buttons.
5. Click Step Next.
6. To proceed to the next line, click Step Next.
7. To play to the next breakpoint in the script, click Continue to Breakpoint.
8. To discontinue the playback, click Stop Playback.

Pausing Playback

You can also use the Pause button on the keyboard to play the script back one line at a time. However, using the Pause button allows you to control playback without using the mouse. You use the Pause button two ways to control playback:

- To pause script playback after beginning playback from the top
- To pause script playback after beginning playback from a chosen command line

▶ To pause script playback after beginning playback from the top

1. In the script pane of the OneWorld Scripting Tool form, remove any breakpoints
2. In the cool bar, click the Stepping On button to turn it off.
3. Click Play in the menu bar and choose Play from Top from the drop-down menu.
4. At the point you want to halt playback, press the Pause button on the keyboard.
5. To continue playback to the next command line, press the Pause button again.

   Note: OneWorld Scripting Tool does not proceed to the next command line until you press the Pause button again.

6. To stop playback, click the Stop button in the cool bar.

▶ To pause script playback after beginning playback from a chosen command line

1. In the script pane in the OneWorld Scripting Tool form, choose a command line in the script from which you want to play back.
2. Click the command line.
3. In the Play menu, choose Stepping On.
4. Click the Play from Cursor button.
5. Click the Pause button.

   OneWorld Scripting Tool runs the selected command line and advances to the next command line.

6. Continue through the script one command line at a time, pressing the Pause button each time you want to run a command.

Toggling a Breakpoint

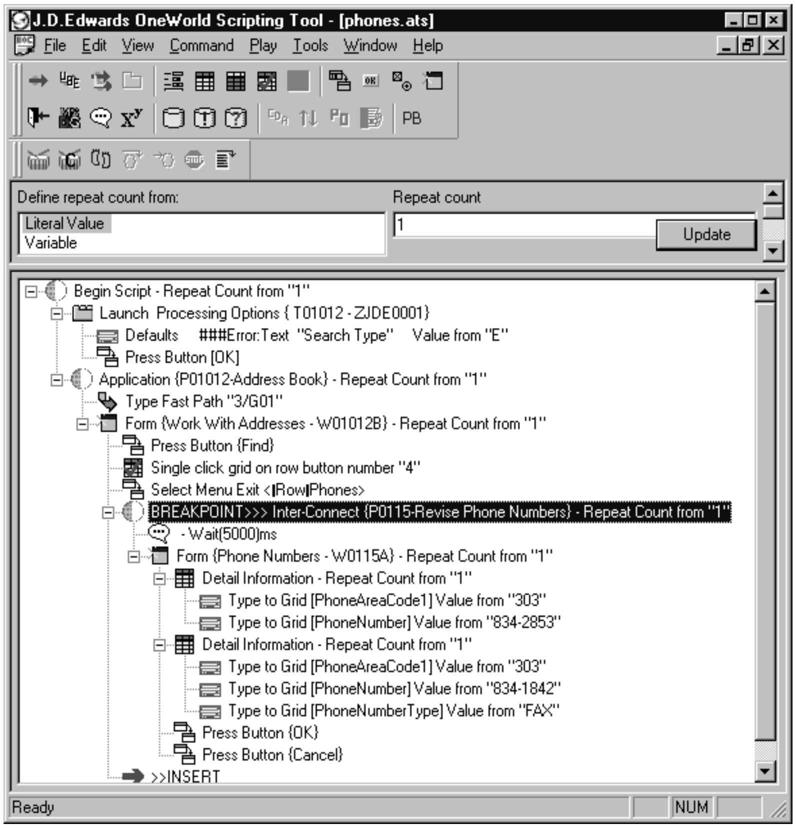
If you want the script to play only to a predetermined command line in the script, you can toggle a breakpoint by highlighting then right clicking the command line at which you want playback to break. You can toggle as many breakpoints as you like. Toggling a breakpoint may be especially useful if you have created a lengthy script and want to play back only a portion of it rather than the whole script.

When you toggle a breakpoint, then play back the script, the playback proceeds to the line at which you set the breakpoint, then halts until you either stop playback or continue it to another breakpoint. You can also highlight and right click a command line to remove a breakpoint.

▶ To toggle a breakpoint

1. In the script pane of the OneWorld Scripting Tool form, choose a playback breakpoint by highlighting a line in the script.
2. Right-click the mouse.
3. Click Toggle Breakpoint.

   OneWorld Scripting Tool inserts the breakpoint to the script.



Define repeat count from:
Literal Value
Variable

Repeat count: 1

```
⊟ Begin Script - Repeat Count from "1"
  ⊟ Launch  Processing Options { T01012 - ZJDE0001}
      Defaults    ###Error:Text  "Search Type"    Value from "E"
      Press Button [OK]
  ⊟ Application {P01012-Address Book} - Repeat Count from "1"
      Type Fast Path "3/G01"
    ⊟ Form {Work With Addresses - W01012B} - Repeat Count from "1"
        Press Button {Find}
        Single click grid on row button number "4"
        Select Menu Exit <|Row|Phones>
      ⊟ BREAKPOINT>>> Inter-Connect {P0115-Revise Phone Numbers} - Repeat Count from "1"
          - Wait(5000)ms
        ⊟ Form {Phone Numbers - W0115A} - Repeat Count from "1"
          ⊟ Detail Information - Repeat Count from "1"
              Type to Grid [PhoneAreaCode1] Value from "303"
              Type to Grid [PhoneNumber] Value from "834-2853"
          ⊟ Detail Information - Repeat Count from "1"
              Type to Grid [PhoneAreaCode1] Value from "303"
              Type to Grid [PhoneNumber] Value from "834-1842"
              Type to Grid [PhoneNumberType] Value from "FAX"
            Press Button {OK}
            Press Button {Cancel}
      >>INSERT
```

4. To remove the breakpoint, click a command line to which you entered a breakpoint.
5. Right click the mouse.
6. Click Toggle Breakpoint.

   OneWorld Scripting Tool removes the breakpoint you inserted.

Playing the Script to a Breakpoint

After you have toggled a breakpoint, you can play back the script, either from the top or from a cursor position of your choice. When OneWorld Scripting Tool reaches the command line containing the breakpoint, playback halts. However, OneWorld Scripting Tool does not cancel playback. If you want to continue scripting, or if you want to play back the script differently, you must click the Stop button to cancel playback.

▶ To play the script to a breakpoint

1. From the script pane in the OneWorld Scripting Tool form, toggle a breakpoint in the script.
2. In the play menu, click Play from Top.
3. If the script plays to the breakpoint and you want to continue scripting, toggle off the breakpoint.
4. In the play menu, click Stop.

Continuing Playback to a Breakpoint

Once you have toggled one or more breakpoints and stepped on the script, you can play the script back from breakpoint to breakpoint by clicking the Continue to Breakpoint button on the cool bar.

▶ To continue playback to a breakpoint

1. From the script pane in OneWorld Scripting Tool, choose a playback breakpoint by highlighting a line in the script.
2. Right-click the mouse.
3. Click Toggle Breakpoint to insert the breakpoint in the script.
4. Choose a point in the script from which you want to play back.
5. Highlight the line by clicking it.
6. In the play menu, click Stepping On.
7. Click Play from Cursor.
8. Click Continue to Breakpoint.

   Note: You can set as many breakpoints in the script as you desire and click Continue to Breakpoint each time playback reaches one.

Ignoring Breakpoints in the Script

You can ignore breakpoints you have toggled in the script if you want to play back the script without interruptions but do not want to toggle off the breakpoints. To do so, you click the Ignore Breakpoints in the Script button. If you then want to stop playback at the breakpoints, you simply click the button again to toggle off.

▶ To ignore breakpoints during script playback

1. In the OneWorld Scripting Tool script pane, toggle on one or more breakpoints in the script.
2. Click the Ignore Breakpoints during Script Playback button in the OneWorld Scripting Tool cool bar.

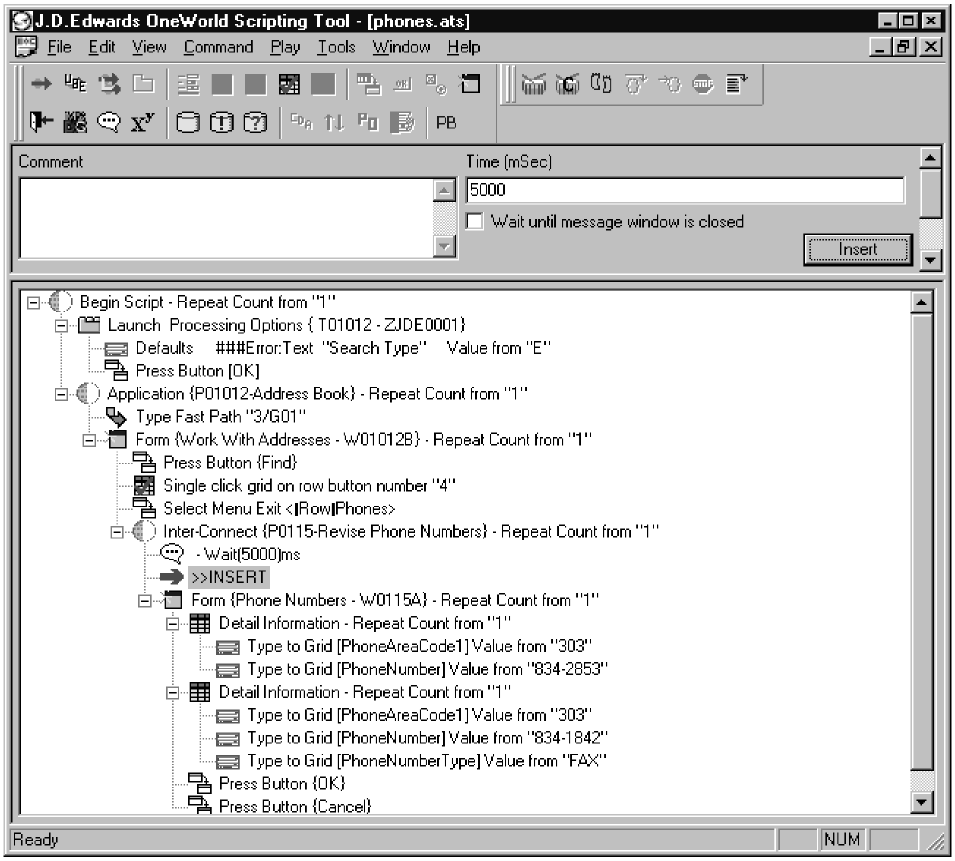
Inserting a Wait Command in the Script

If you insert a breakpoint in the script, playback halts when OneWorld Scripting Tool reaches the breakpoint. Playback does not resume or stop without your intervention. On the other hand, if you click Comment/Wait in the command menu, you can script a specified wait period, or pause, at a predetermined script command line. When the playback reaches this command line, the wait occurs, then playback proceeds.

▶ To insert a Wait command in the script

1. In the script pane in the OneWorld Scripting Tool form, highlight a command line in the script by clicking it.
2. In the command menu, click Comment/Wait.
3. Press the Tab key or place the cursor in the unpopulated Time (msec) list.
4. Type a time, in milliseconds, for the wait.

   Caution: Do not use commas in typing the time of the wait.



Comment

Time (mSec) 5000

□ Wait until message window is closed

```
⊟ Begin Script - Repeat Count from "1"
   ⊟ Launch  Processing Options { T01012 - ZJDE0001}
      Defaults     ###Error:Text  "Search Type"    Value from "E"
      Press Button [OK]
   ⊟ Application {P01012-Address Book} - Repeat Count from "1"
      Type Fast Path "3/G01"
      ⊟ Form {Work With Addresses - W01012B} - Repeat Count from "1"
         Press Button {Find}
         Single click grid on row button number "4"
         Select Menu Exit <|Row|Phones>
         ⊟ Inter-Connect {P0115-Revise Phone Numbers} - Repeat Count from "1"
            - Wait(5000)ms
            >>INSERT
            ⊟ Form {Phone Numbers - W0115A} - Repeat Count from "1"
               ⊟ Detail Information - Repeat Count from "1"
                  Type to Grid [PhoneAreaCode1] Value from "303"
                  Type to Grid [PhoneNumber] Value from "834-2853"
               ⊟ Detail Information - Repeat Count from "1"
                  Type to Grid [PhoneAreaCode1] Value from "303"
                  Type to Grid [PhoneNumber] Value from "834-1842"
                  Type to Grid [PhoneNumberType] Value from "FAX"
               Press Button {OK}
               Press Button {Cancel}
```

5. Click the Insert button.
6. Run a playback command.

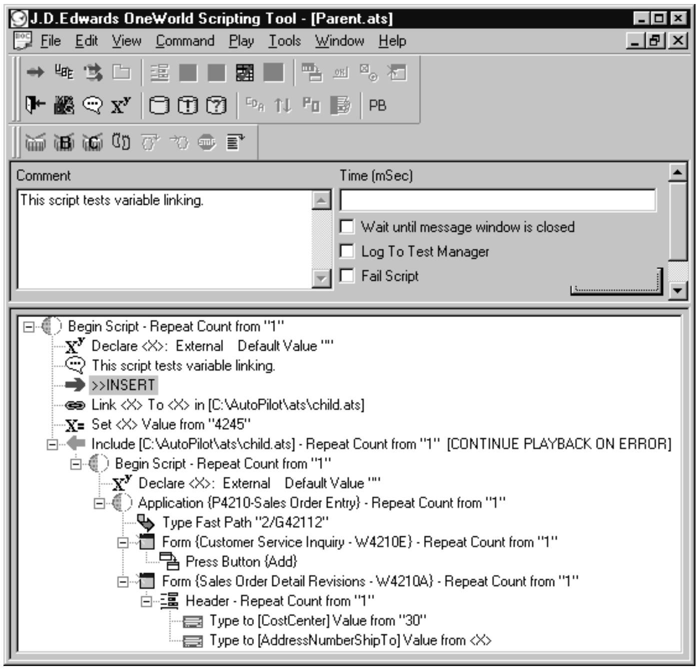
Inserting a Comment in the Script

You can also use the Comment/Wait command to insert into the script pane comments about the command line you have chosen or general comments about the script, including its purpose. If you want to include the script in batch testing run by Test Manager and want the comments to appear in a summary report after batch testing, choose the Log to Test Manager option.

▶ To insert a comment in the script

1. In the script pane of the OneWorld Scripting Tool form, place the insertion cursor at the point in the script that you want the comment to appear.
2. In the command menu, click Comment/Wait.
3. In the unpopulated Comment list of the OneWorld Scripting Tool command pane, type a comment.

4. Choose the Log to Test Manager option if you want OneWorld Scripting Tool to include the comment in a summary report after testing.
5. Click the Insert button.


J.D.Edwards OneWorld Scripting Tool - [Parent.ats]
File Edit View Command Play Tools Window Help
PB

Comment

This script tests variable linking.

Time (mSec)

□ Wait until message window is closed
□ Log To Test Manager
□ Fail Script

```
Begin Script - Repeat Count from "1"
    Declare <X>:  External    Default Value ""
    This script tests variable linking.
    >>INSERT
    Link <X> To <X> in [C:\AutoPilot\ats\child.ats]
    Set <X> Value from "4245"
    Include [C:\AutoPilot\ats\child.ats] - Repeat Count from "1"  [CONTINUE PLAYBACK ON ERROR]
        Begin Script - Repeat Count from "1"
            Declare <X>:  External    Default Value ""
            Application {P4210-Sales Order Entry} - Repeat Count from "1"
                Type Fast Path "2/G42112"
                Form {Customer Service Inquiry - W4210E} - Repeat Count from "1"
                    Press Button {Add}
                Form {Sales Order Detail Revisions - W4210A} - Repeat Count from "1"
                    Header - Repeat Count from "1"
                        Type to [CostCenter] Value from "30"
                        Type to [AddressNumberShipTo] Value from <X>
```

Failing a Script

You can automatically fail a script by choosing the Fail script option. This option appears in the command pane when you click the Comment/Wait button. The Fail script option is useful if you are going to include the script in a batch to be run by Test Manager.

▶ To fail a script

1. In the script pane of the OneWorld Scripting Tool form, place the insertion cursor at the point that you want fail the script.
2. In the command menu, click Comment/Wait.
3. In the command pane, choose the Fail script option and click the Insert button.

   OneWorld Scripting Tool inserts a red Comment symbol in the script pane to indicate that the script fails at that point.

Setting Transaction Times in the Script

A transaction is a series of events, bounded by a start and end point. You can also insert comments to the script to measure playback transaction time. You use the Comment command to assign a name to the transaction, to insert a starting point, such as launching an application, and to insert a finishing point, such as closing OneWorld. Setting transaction times in the script provides important information about the time OneWorld requires to run a series of commands.

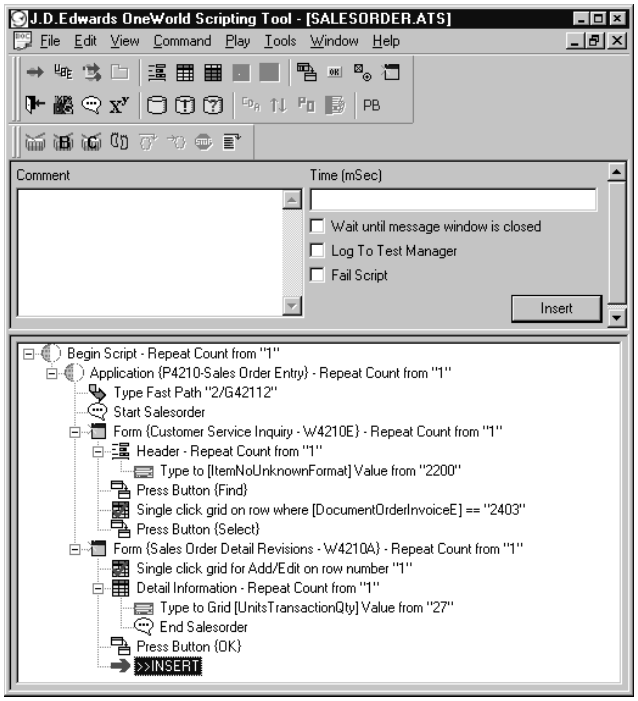
▶ To set transaction times in the script

1. In the script pane of the OneWorld Scripting Tool form, determine the command line that represents the start of the transaction, then place the insertion cursor directly above it.
2. In the OneWorld Scripting Tool menu bar, click Command.
3. Choose Wait/Comment.
4. In the unpopulated Comment list of the OneWorld Scripting Tool command pane, enter Start, a space, and a name for the transaction.
5. Click the Insert button.

   OneWorld Scripting Tool inserts a command line marking the start of the transaction.

6. Determine the command line that represents the end of the transaction, then place the insertion cursor after it.
7. In the OneWorld Scripting Tool menu bar, click Command.
8. Choose Comment/Wait.
9. In the unpopulated Comment list of the OneWorld Scripting Tool command pane, type End, a space, and a name for the transaction.

   OneWorld Scripting Tool inserts a command line marking the end of the transaction.


J.D.Edwards OneWorld Scripting Tool - [SALESORDER.ATS]
File Edit View Command Play Tools Window Help
PB
Insert

Comment

Time (mSec)

□ Wait until message window is closed

□ Log To Test Manager

□ Fail Script

```
Begin Script - Repeat Count from "1"
  Application {P4210-Sales Order Entry} - Repeat Count from "1"
    Type Fast Path "2/G42112"
    Start Salesorder
    Form {Customer Service Inquiry - W4210E} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [ItemNoUnknownFormat] Value from "2200"
      Press Button {Find}
      Single click grid on row where [DocumentOrderInvoiceE] == "2403"
      Press Button {Select}
    Form {Sales Order Detail Revisions - W4210A} - Repeat Count from "1"
      Single click grid for Add/Edit on row number "1"
      Detail Information - Repeat Count from "1"
        Type to Grid [UnitsTransactionQty] Value from "27"
        End Salesorder
      Press Button {OK}
      >>INSERT
```

Caution: The name that you assign to the end of the transaction must exactly match the name that you assign to the start of the transaction.

10. Click the Insert button.

See Also

- *OneWorld Virtual User Tool Guide*

Sample Script

Creating a Sample OneWorld Scripting Tool Script

You can create customized scripts in OneWorld Scripting Tool to test various applications and processes in OneWorld. When you create a script you use the command menu or the cool bar buttons to choose a context or action command that you want to write. Then you work in the command pane, making choices from lists and inserting them to the script.

The sequence of commands that you insert to the script is determined by the tasks that you want to complete in OneWorld. You tailor or customize your script to test running tasks that you frequently perform in OneWorld. In addition, you can use OneWorld Scripting Tool to test applications that you have not yet used or have not frequently used, and you can use it to test new builds of OneWorld that you might have recently added. When you create a OneWorld Scripting Tool script and play it back, you can expose user error that might cause a script to fail. You might also expose a problem that is preventing OneWorld from running correctly.

Although OneWorld Scripting Tool can be used to create a script to check the efficacy of any application, this sample script uses the application Standard Voucher Entry. This section develops a sample script built around this application, step by step. Note that the sample script presented here does not provide examples of every function or feature possessed by OneWorld Scripting Tool. For example, this script tests an interactive application and does not launch a UBE. Consult other sections of this guide if you need information on a function not covered by the sample script.

Remember that there is no absolute set of steps to follow in writing a script. The precise way that you do so is mainly determined by your knowledge of a specific OneWorld application.

Creating the Sample OneWorld Scripting Tool Script

It is important to remember that you use OneWorld Scripting Tool to create customized scripts that you tailor to the specific applications that you most often use. The sample script presented here presents in linear fashion how you use many of the commands discussed in the previous sections. However, the sample script does not include all of the context and action commands discussed earlier in this guide, nor does it represent in any way a definitive method for using OneWorld Scripting Tool. For example, this script tests functions performed in a OneWorld interactive application. However, you might want to test the launch and submission of a UBE. Doing so would require you to write a very different set of commands.

This chapter covers the following tasks undertaken to complete the sample script:

- ❑ Launching a OneWorld application and form
- ❑ Declaring a variable
- ❑ Adding a new form
- ❑ Typing data to a header control
- ❑ Typing data to a grid column
- ❑ Updating the repeat count
- ❑ Updating the database
- ❑ Setting the value of a variable
- ❑ Returning to a previous OneWorld form
- ❑ Entering data to a QBE line
- ❑ Finding records
- ❑ Selecting records and deleting them from the database
- ❑ Exiting the script

Launching a OneWorld Application and Form

Launching the OneWorld application from OneWorld Scripting Tool establishes one basic scripting context necessary for many other scripting commands that you might want to insert and run. Launching a OneWorld application does not have to be the first command that you script in a OneWorld Scripting Tool session. However, Application is often at least one of the first commands that you script.

Choosing a OneWorld application in OneWorld Scripting Tool also requires that you choose a OneWorld form that is part of that application. After you have launched a OneWorld application from OneWorld Scripting Tool, you can script a variety of additional context and action commands. For example, after you have established the context as a form, you can then script inputs to header controls, grid columns and QBE lines. After you have established one of these contexts, you can script pressing buttons to move to different forms or to perform functions within the active form.

▶ To launch a OneWorld application and form

1. From your desktop, click the OneWorld Scripting Tool icon on the desktop.

   The OneWorld Scripting Tool splash screen appears, followed by the OneWorld Scripting Tool form.

2. In the menu bar, click File and New.

   The command pane and script pane are unpopulated.

3. From the command menu, choose Application.

   The Application list is populated, while the Menu list is as yet unpopulated.

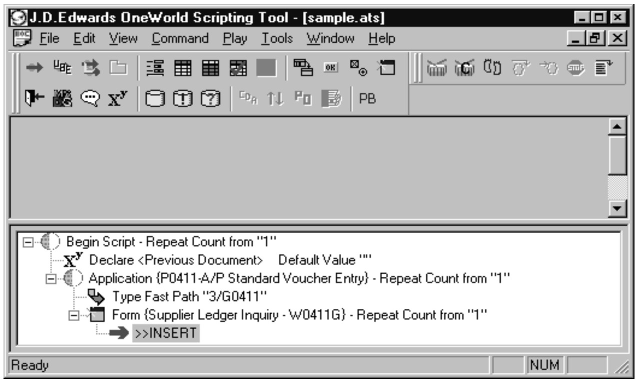
4. Click an application code, for example, P0411 for A/P Standard Voucher Entry.

   OneWorld Scripting Tool populates the Menu list with items that fall under three headings:

   - Menu Text, for example, Standard Voucher Entry
   - Fast Path, which correlates to a specific OneWorld form
   - Version of the application, which correlates to the processing options that exist for a particular application

5. Click a Menu Text item for the application you chose, for example, Standard Voucher Entry, Fast Path 3G0411, Version ZJDE0001.

6. Click the Insert button in the lower right-hand corner of the command pane.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
  Declare <Previous Document>    Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      >>INSERT
```

Note: When you turn on the playback button on the cool bar, OneWorld Scripting Tool launches OneWorld, and the OneWorld form specified by the Fast Path, in this case, Supplier Ledger Inquiry, appears.

Declaring a Variable

Before you can use a variable as a source of input, you must first declare it, or give it a name. Doing so establishes the place that you store the value. You can declare a variable at any point in the script, but once you do, you might want to place it at the top of the script to make it global, meaning that you can set its value at any point in the script. If you make the variable global, you can launch multiple applications within the script and use the stored value in any of the applications. If you decide to declare a variable after an application has been launched, you can use the mouse to drag the variable’s Declare command to the top of the script to make it global.

For this script you use a previous document number to retrieve voucher entry data. You decide early in the script to declare the variable so that you have a place already established to store the previous document number once you know what it is.

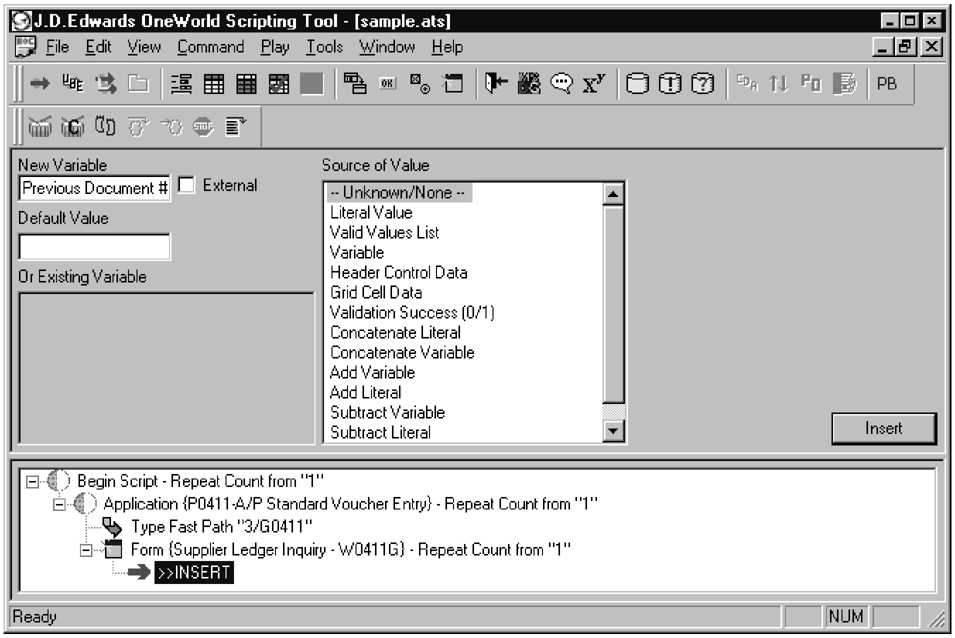
▶ To declare a variable

1. In the command menu of the OneWorld Scripting Tool form, click Variables.
2. Type a name for the variable in the Declare New Variable Name list.

   For this exercise, you call the variable Previous Document #.

3. In the Source of Value list, click None.

Note: Choosing None means that you simply want to give the variable a name at this point. You don't set its value yet.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

New Variable: Previous Document # ☐ External

Default Value:

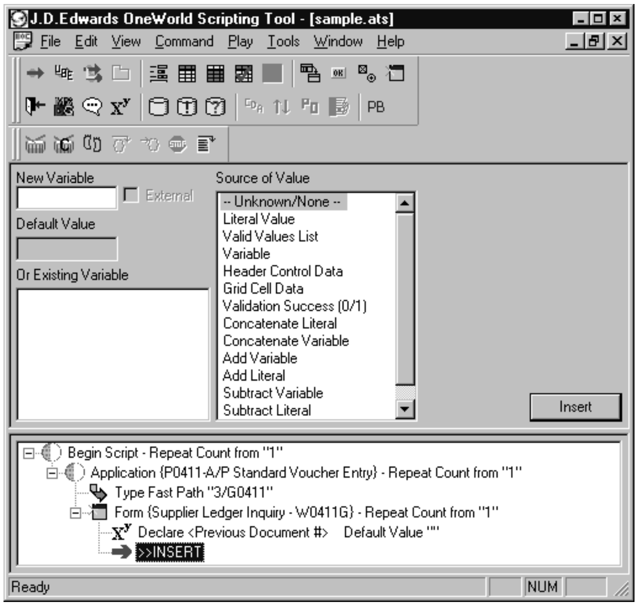
Or Existing Variable:

Source of Value:

- -- Unknown/None --
- Literal Value
- Valid Values List
- Variable
- Header Control Data
- Grid Cell Data
- Validation Success (0/1)
- Concatenate Literal
- Concatenate Variable
- Add Variable
- Add Literal
- Subtract Variable
- Subtract Literal

```
Begin Script - Repeat Count from "1"
    Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
        Type Fast Path "3/G0411"
        Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
            >>INSERT
```

4. Click the Insert button.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

New Variable: ☐ External

Default Value:

Or Existing Variable:

Source of Value:

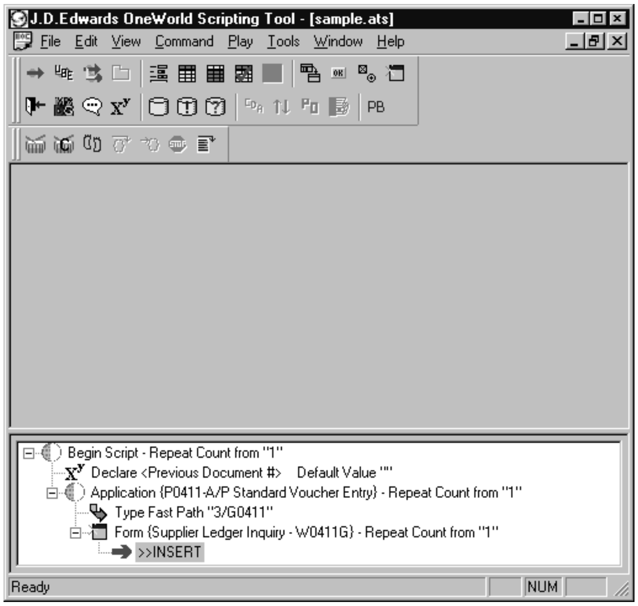
- -- Unknown/None --
- Literal Value
- Valid Values List
- Variable
- Header Control Data
- Grid Cell Data
- Validation Success (0/1)
- Concatenate Literal
- Concatenate Variable
- Add Variable
- Add Literal
- Subtract Variable
- Subtract Literal

```
Begin Script - Repeat Count from "1"
    Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
        Type Fast Path "3/G0411"
        Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
            Declare <Previous Document #>   Default Value ""
            >>INSERT
```

OneWorld Scripting Tool inserts the variable's Declare command after the Supplier Ledger Inquiry Form command line. At this point, you can use the variable only within that Form node because it is attached to the Form node.

5. Click the Declare command line in the script pane to highlight it.

6. Hold down the mouse button and drag the Declare command line until it is on top of the Application command line. When the indicator arrow is pointing up, release the mouse button.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB

```
Begin Script - Repeat Count from "1"
  Declare <Previous Document #>    Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      >>INSERT
```

You have now attached the Declare command line to the Begin Script node, and you can use a value that you set for the variable at any point in the script.

Adding a New Form

For this sample script, you want to move from the Supplier Ledger Inquiry form to the Enter Voucher Payment Information form. To do so, you script pressing the Add button to change forms. You also choose Enter Voucher Payment Information from the Next Form list so that the Form command line in OneWorld Scripting Tool mirrors the active form in OneWorld.

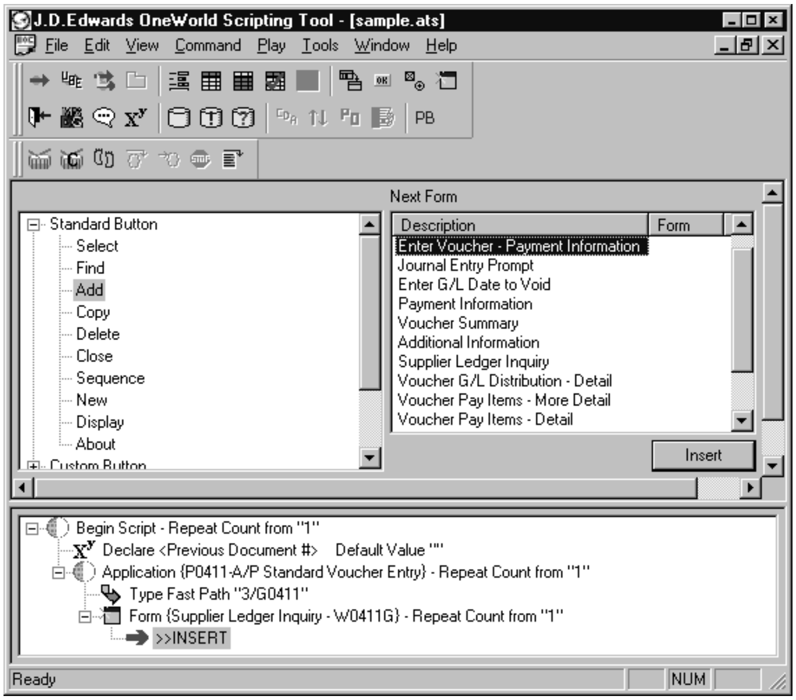
▶ To add a new form

1. In the command menu of the OneWorld Scripting Tool form, with the playback button turned on, click Press Toolbar Button.
2. From the Button list, click Press Standard Button.

   Note: The options listed under Press Standard Button in OneWorld Scripting Tool mirror the buttons on the menu bar in the OneWorld form.

3. Click Add.

4. From the Next Form list, click the name of the form that appears in OneWorld when you click Add, in this example, Enter Voucher - Payment Information.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Next Form
Insert
Ready
NUM

Standard Button
- Select
- Find
- Add
- Copy
- Delete
- Close
- Sequence
- New
- Display
- About

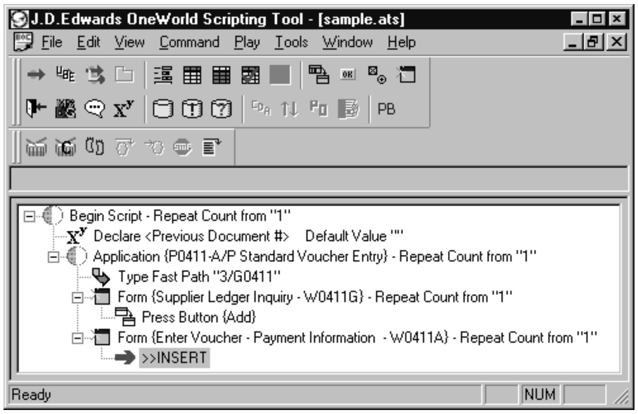
Custom Button

Description	Form
Enter Voucher - Payment Information	
Journal Entry Prompt	
Enter G/L Date to Void	
Payment Information	
Voucher Summary	
Additional Information	
Supplier Ledger Inquiry	
Voucher G/L Distribution - Detail	
Voucher Pay Items - More Detail	
Voucher Pay Items - Detail	

```
Begin Script - Repeat Count from "1"
    Declare <Previous Document #>    Default Value ""
    Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
        Type Fast Path "3/G0411"
        Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
            >>INSERT
```

5. Click the Insert button.

OneWorld Scripting Tool adds the commands you have inserted to the script pane. In the playback mode, the Enter Voucher Payment Information form appears in OneWorld at this point.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
    Declare <Previous Document #>    Default Value ""
    Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
        Type Fast Path "3/G0411"
        Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
            Press Button {Add}
        Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
            >>INSERT
```

Caution: If you chose Unknown/None from the Next Form list in step 4, you must complete an additional series of steps. You must script a Form command line that mirrors the form that is active in OneWorld. Remember that at all times the most current Form command line in the

OneWorld Scripting Tool script pane must mirror the form that you have active in OneWorld. If there is a difference, you cannot continue scripting.

6. With the OneWorld Scripting Tool playback button turned on so that OneWorld is active, note the name of the OneWorld form that is active, in this case the Enter Voucher - Payment Information form.
7. In OneWorld Scripting Tool, click Form in the command menu.
8. Choose the name of the active OneWorld form from the Form list. In this example, you choose Enter Voucher - Payment Information.

   In the script pane, a Form command line appears that includes the name of the form you chose.

9. Click the Insert button.

Typing Data to a Header Control

▶ To type data to a header control

Continue the sample script by adding commands to type inputs to header controls in OneWorld. Like Application and Form, Header is a context command because it establishes the context in which you take further actions.

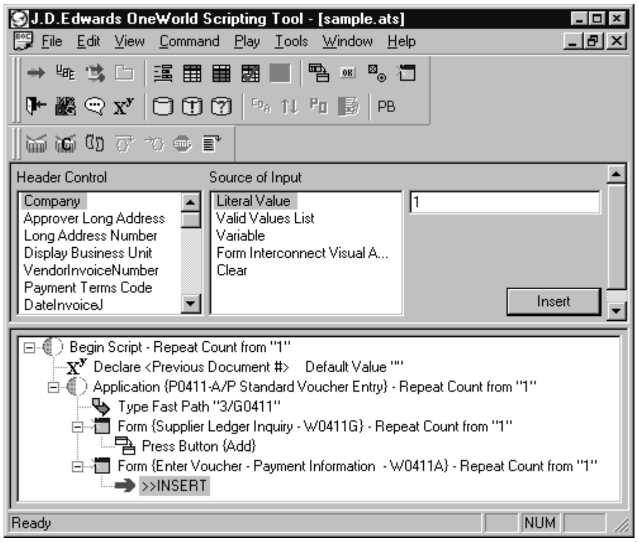
1. In the OneWorld Scripting Tool command pane, click Set Header Control Value.
2. In the Header Control list, choose the name of a header control into which you want to input data, for example, Company.

   Note: In the OneWorld form, the header control you choose in OneWorld Scripting Tool is highlighted with a BlueCue.

3. Choose the source of input to the control:
   - Literal value
   - Valid values list
   - Variable
   - UDC visual assist value
   - Form interconnect visual assist value

   For this example, choose Literal Value. When you choose a source of input, OneWorld Scripting Tool captions the value selection list. The wording of the caption depends on the source of input.

4. For this example, click inside the unpopulated Literal Value list and type 1.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

Header Control: Company, Approver Long Address, Long Address Number, Display Business Unit, VendorInvoiceNumber, Payment Terms Code, DateInvoiceJ

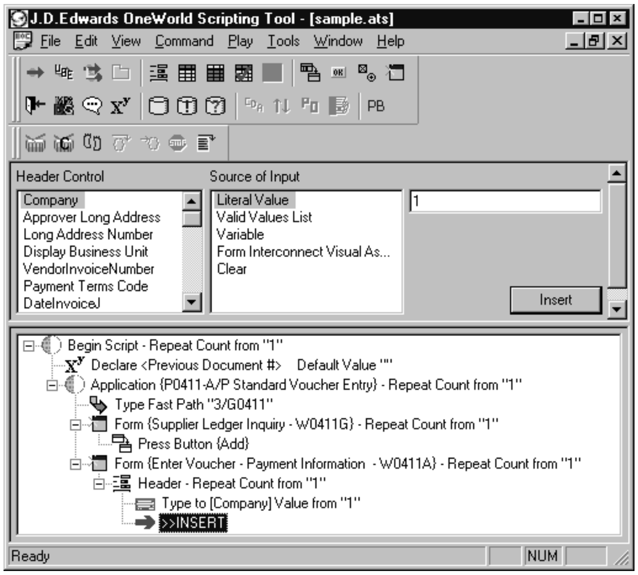
Source of Input: Literal Value, Valid Values List, Variable, Form Interconnect Visual A..., Clear

1

```
Begin Script - Repeat Count from "1"
  Declare <Previous Document #>    Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      >>INSERT
```

5. Click the Insert button.

   OneWorld Scripting Tool types the scripted input to the Company header control in the Enter Voucher - Payment Information form. OneWorld Scripting Tool encloses literal values in quotes in the script pane.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

Header Control: Company, Approver Long Address, Long Address Number, Display Business Unit, VendorInvoiceNumber, Payment Terms Code, DateInvoiceJ

Source of Input: Literal Value, Valid Values List, Variable, Form Interconnect Visual As..., Clear

1

```
Begin Script - Repeat Count from "1"
  Declare <Previous Document #>    Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        >>INSERT
```

6. Continue to script inputs to header controls by clicking Set Header Control Value in the command menu, choosing a control, choosing a source of input, and selecting a value.

Creating a Valid Values List

For this sample script, you want to script inputs to the header control Long Address Number. However, instead of entering a literal value to the control, you decide to use Valid Values List as the source of input.

Remember that a valid values list consists of values that you collect and store under a name of your choice. You use a valid values list if, for example, you want to input more than one value to a header or grid column. There are two types of valid values lists:

- List of literal values
- Simple database query

For this script, you create a list of literal values, which contains values that you assign. With a simple database query, OneWorld Scripting Tool draws the values from a database of your choosing. Note that you could continue to enter literal values to each header control. However, you can include as many values in the valid values list as you want. Each time you play back the script, OneWorld Scripting Tool automatically inserts one of the values to the appropriate header control.

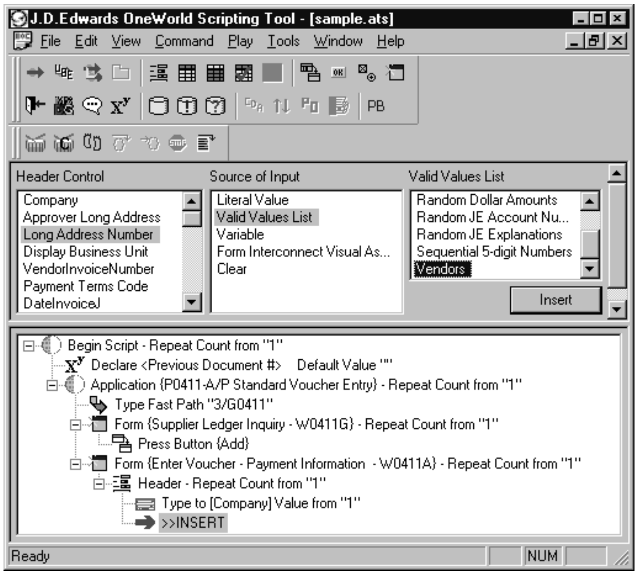
If the value you want to insert to a header or grid column is constant, choosing a literal value makes sense. However, if the value is likely to change, it makes sense to create a valid values list so that OneWorld Scripting Tool inserts a new value each time. In this case, the long address number is different for each vendor, so you decide to create a valid values list and give it the name Vendors.

▶ To create a valid values list

1. In the menu bar of the OneWorld Scripting Tool form, click Tools.
2. Choose Generate Valid Values List.
3. Choose the List of Literal Values option from the Select Data File Type window.
4. Click Next.
5. Type a file name in the File Properties list. For this example, name the list Vendors.
6. Type one or more values in the Values list. After each entry, press Enter. The values should be stacked vertically in the box.
7. Click Finish.
8. Click Set Header Control Value in the command menu.
9. In the Header Control list, choose the name of a header control into which you want to input data, for example, Long Address Number.
10. Click Valid Values List as the source of input.

11. Click the name of the valid values list you created, in this case, Vendors.
12. Click the Insert button.

    OneWorld Scripting Tool types the first value in the list to the Long Address Number control in the Enter Voucher - Payment Information form. OneWorld Scripting Tool identifies the valid values list in the script pane with a backslash, the name you assigned the list, and the extension .atd.

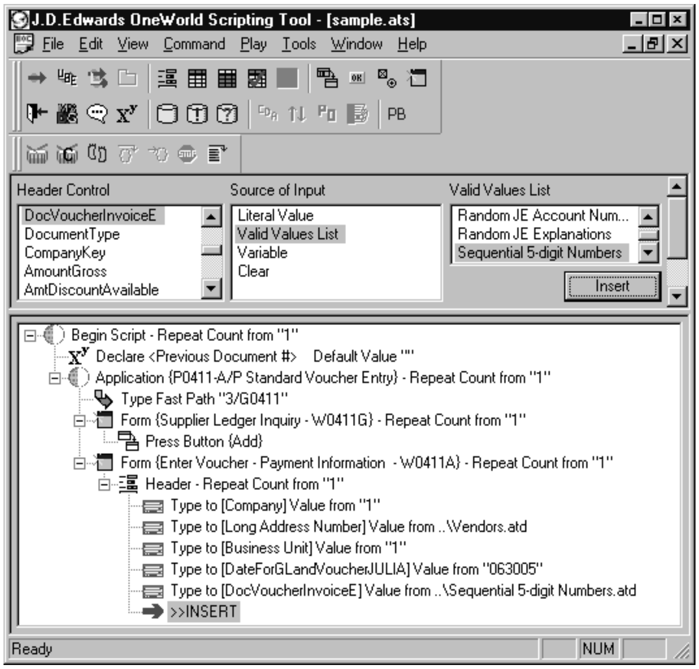


```
Begin Script - Repeat Count from "1"
  Declare <Previous Document #>    Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        >>INSERT
```

13. Complete the following tasks:

    - Script an input to the header control Business Unit. Use a literal value of 1.
    - Script an input to the header control DateForGLandVoucherJULIA. Use a literal value of 063005.
    - Script an input to header control DocVoucherInvoiceE, using a list of literal values you have created and given the name Sequential 5-Digit Numbers.

    When you have inserted inputs to each of the header controls, the script pane should contain five Type to action commands within the Header node.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

Header Control: DocVoucherInvoiceE, DocumentType, CompanyKey, AmountGross, AmtDiscountAvailable

Source of Input: Literal Value, Valid Values List, Variable, Clear

Valid Values List: Random JE Account Num..., Random JE Explanations, Sequential 5-digit Numbers

```
Begin Script - Repeat Count from "1"
  Declare <Previous Document #>   Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        Type to [Long Address Number] Value from ..\Vendors.atd
        Type to [Business Unit] Value from "1"
        Type to [DateForGLandVoucherJULIA] Value from "063005"
        Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
        >>INSERT
```

Typing Data to a Grid Column

You now decide to script inputs to the grid columns in the OneWorld form. For this sample script, you need to make voucher payment inputs to the grid columns. The Grid command, like the Header command, is a context command because it establishes the grid column in the OneWorld form as the environment in which you take additional actions.

The names of the grid columns in the Grid Column list of the OneWorld Scripting Tool command pane mirror the names of the grid columns on the OneWorld form. When you choose the name of a grid column from the Grid Column list, a BlueCue appears as an arrow over the corresponding grid column in the OneWorld form.

For the sample script, you decide to make voucher entries to multiple rows of the grid columns of the form. You could do this by entering literal values to each grid row, but it is easier to create a valid values list, and then update the repeat count of the command line. For this script, you want to type inputs to the Gross Amount and Remark grid columns.

▶ To type data to a grid column

1. In the command menu of the OneWorld Scripting Tool form, click Set Grid Cell Value.

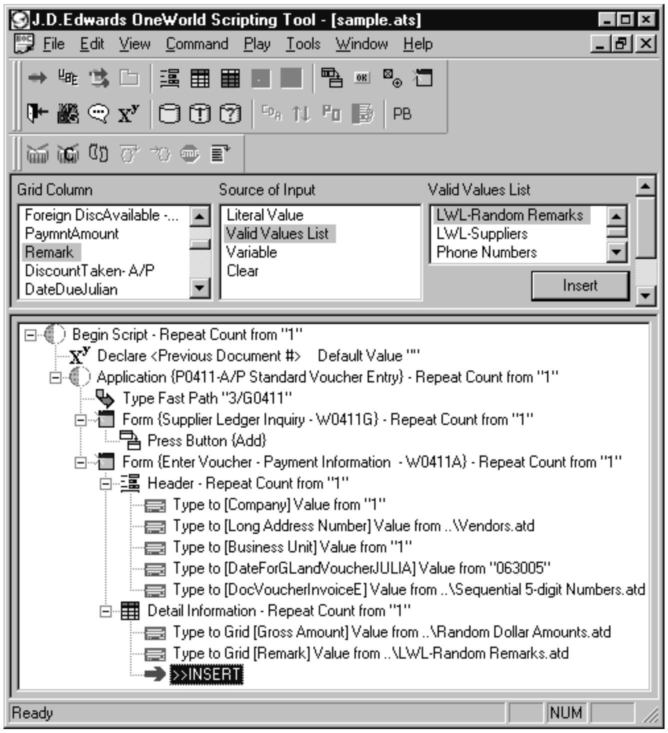
   The command pane has two lists:

- Grid Column
- Source of Input

2. Choose the name of a grid column from the Grid Column list. For example, choose Gross Amount.
3. Create a list of literal values that contains five values and name it Random Dollar Amounts. This list contains amounts gross amounts paid to vendors.
4. In the command menu, click Set Grid Cell Value.
5. Click Valid Values List in the Source of Input list.
6. Click Random Dollar Amounts.
7. Click the Insert button.
8. Repeat steps 1-6 to script an input to the Remarks column.

   Name the list of literal values Random JE Explanations. This list contains explanations for each entry in the Gross Amount column, for instance, “Rent.”

   When you have inserted inputs to each of the grid columns, the script pane should contain two Type to action commands within the Detail Information (grid) node.



Grid Column
- Foreign DiscAvailable -...
- PaymntAmount
- Remark
- DiscountTaken- A/P
- DateDueJulian

Source of Input
- Literal Value
- Valid Values List
- Variable
- Clear

Valid Values List
- LWL-Random Remarks
- LWL-Suppliers
- Phone Numbers

```
Begin Script - Repeat Count from "1"
  Declare <Previous Document #>   Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        Type to [Long Address Number] Value from ..\Vendors.atd
        Type to [Business Unit] Value from "1"
        Type to [DateForGLandVoucherJULIA] Value from "063005"
        Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
      Detail Information - Repeat Count from "1"
        Type to Grid [Gross Amount] Value from ..\Random Dollar Amounts.atd
        Type to Grid [Remark] Value from ..\LWL-Random Remarks.atd
        >>INSERT
```

To access English documentation updates, see https://knowledge.jdedwards.com/JDEContent/documentationcbt/overview/about_documentation_updates.pdf

Updating the Repeat Count

Because you have scripted as the source of input two lists of literal values that each contain five values, you might want to change the repeat count for this node so that each of the five values are input to the grid during playback.

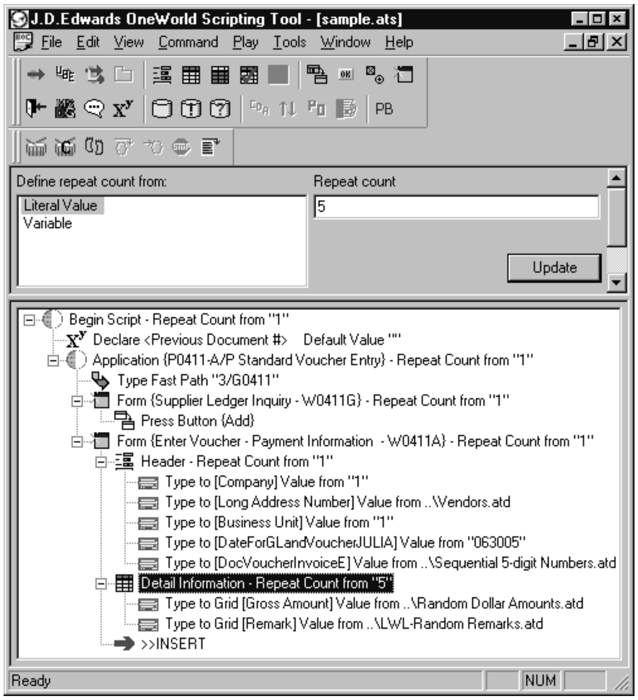
▶ To update the repeat count

1. In the command menu of the OneWorld Scripting Tool form, if the playback mode is turned on, turn it off by clicking the Playback button in the Cool bar.
2. In the script pane, click the node you want to update, in this case, Detail Information.

   In the command pane, two lists appear:

   - Define repeat count from
   - Repeat Count
3. Choose Literal Value from the Define repeat count list.
4. In the Repeat Count list, type a number.

   In this case, you type the number 5, because you want to script the entry to rows in the grid of five separate values that you included in your valid values lists.
5. Click the Update button.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Update
Ready
NUM

Define repeat count from: Literal Value / Variable

Repeat count: 5

```
Begin Script - Repeat Count from "1"
  Declare <Previous Document #>    Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        Type to [Long Address Number] Value from ..\Vendors.atd
        Type to [Business Unit] Value from "1"
        Type to [DateForGLandVoucherJULIA] Value from "063005"
        Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
      Detail Information - Repeat Count from "5"
        Type to Grid [Gross Amount] Value from ..\Random Dollar Amounts.atd
        Type to Grid [Remark] Value from ..\LWL-Random Remarks.atd
      >>INSERT
```

Note: The repeat Count in the Detail Information node is now 5. If you play back the script, OneWorld Scripting Tool loops through this node five times. Each time it loops, OneWorld Scripting Tool inserts a different value from you valid values lists to the Gross Amount and Remark grid columns.

6. Click the Playback button in the cool bar to turn the playback function back on.

Updating the Database

Having scripted the desired entries to the Enter Voucher - Payment Information form, you now want to update the OneWorld database with the new entries. Doing so requires that you script clicking the OK button, just as you would do when you are working in OneWorld. For this script, you also write a new Form command because clicking the OK button causes OneWorld to launch a new form.

In the new form, you enter data, click the OK button to add it to the database, and then return to the previous form by writing a new Form command.

This topic discusses:

- Updating the database and confirming a new form
- Entering data, updating the database, and returning to a previous form

Before You Begin

- ☐ Create a list of literal values to use in Account Number column of the Enter Voucher - G/L Distribution form. You can call this list Random JE Account Numbers. See *Creating a List of Literal Values*.

▶ To update the database and confirm a new form

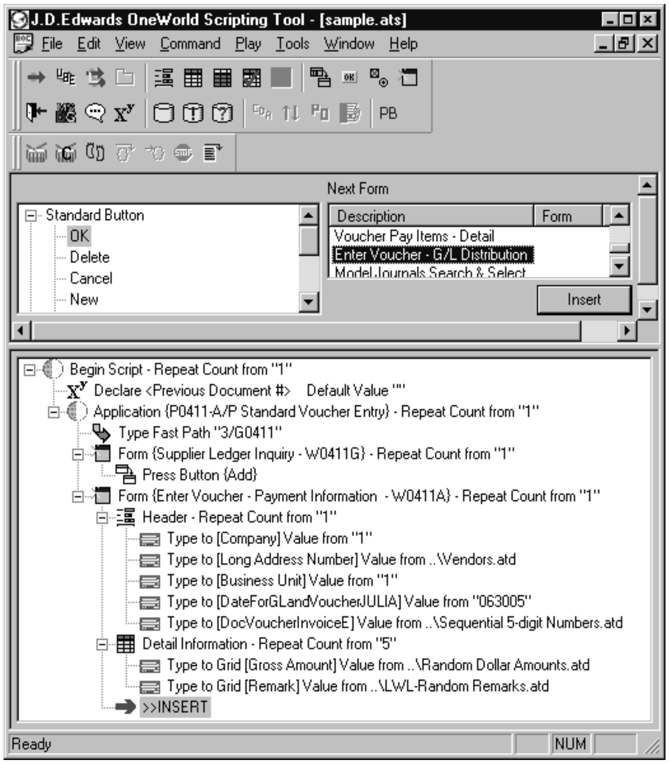
1. In the command menu of the OneWorld Scripting Tool form, click Press Toolbar Button.

   Two lists appear in the command pane:

   - Button
   - Next Form

   The Button list in the command pane defaults to Press Standard Button. For this script, under Press Standard Button, the options mirror the buttons on the OneWorld form:

   - OK
   - Delete
   - Cancel
2. Choose OK from the Button list.
3. From the Next Form list, choose the OneWorld form that follows when a user clicks OK for this application and version. For the sample script, you choose Enter Voucher - G/L Distribution.
4. Click the Insert button.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Next Form
Standard Button
OK
Delete
Cancel
New
Description
Form
Voucher Pay Items - Detail
Enter Voucher - G/L Distribution
Model Journals Search & Select
Insert
Ready
NUM

```
Begin Script - Repeat Count from "1"
  Declare <Previous Document #>    Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        Type to [Long Address Number] Value from ..\Vendors.atd
        Type to [Business Unit] Value from "1"
        Type to [DateForGLandVoucherJULIA] Value from "063005"
        Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
      Detail Information - Repeat Count from "5"
        Type to Grid [Gross Amount] Value from ..\Random Dollar Amounts.atd
        Type to Grid [Remark] Value from ..\LWL-Random Remarks.atd
      >>INSERT
```

The Enter Voucher - G/L Distribution form appears in OneWorld. You want to credit the full voucher payment amount to a particular account number. After you credit the account, you want to update the database, and then return to the Enter Voucher - Payment Information form. To do so, you complete the following tasks:

- Enter an account number to the appropriate grid column in the G/L Distribution form
- Click the OK button to update the database
- Confirm the Enter Voucher - Payment Information form

Note: You can create the list of literal values before you write the script. You can also simply enter a valid literal value to the Account Number column.

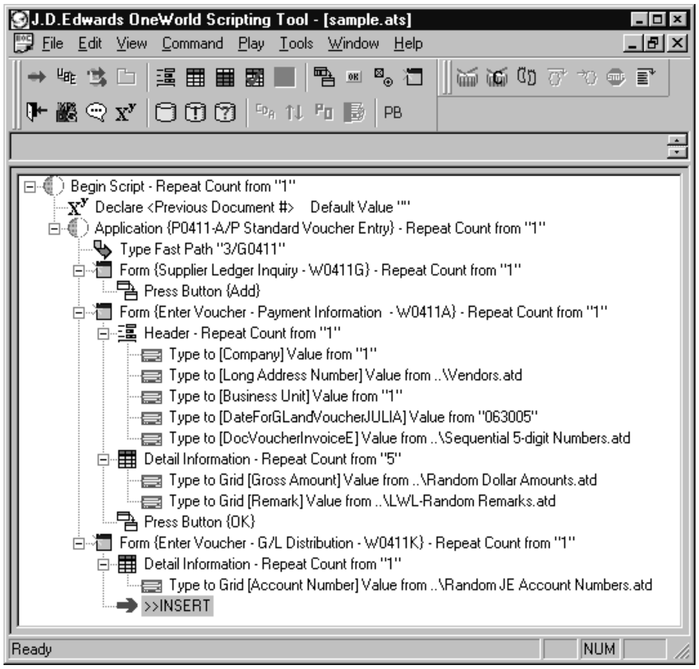
▶ To enter data, update the database, and return to a previous form

1. Create a list of literal values.

   Name the list Random JE Account Numbers. This list contains valid G/L bank account numbers.

2. In the command menu, click Set Grid Cell Value.
3. In the Grid Column list, choose Account Number.

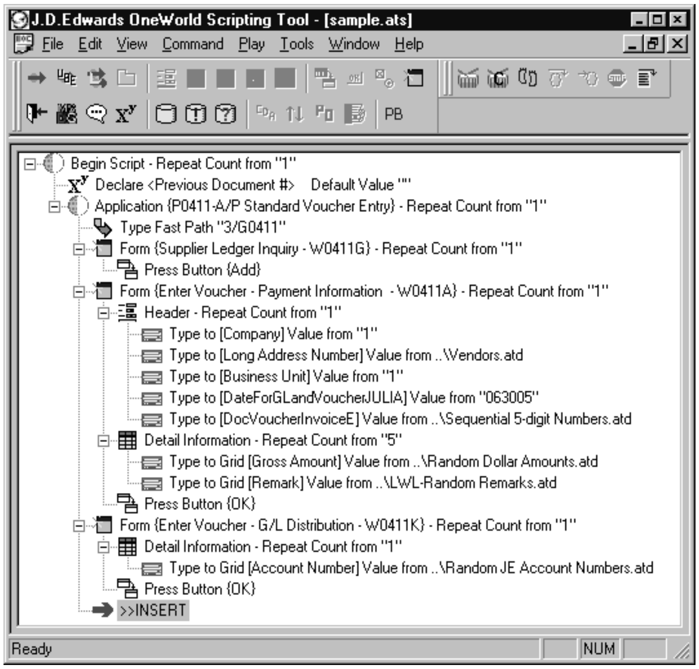
4. Click Valid Values List in the Source of Input list.
5. Click Random JE Account Numbers.
6. Click the Insert button.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
  Declare <Previous Document #>    Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        Type to [Long Address Number] Value from ..\Vendors.atd
        Type to [Business Unit] Value from "1"
        Type to [DateForGLandVoucherJULIA] Value from "063005"
        Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
      Detail Information - Repeat Count from "5"
        Type to Grid [Gross Amount] Value from ..\Random Dollar Amounts.atd
        Type to Grid [Remark] Value from ..\LWL-Random Remarks.atd
      Press Button {OK}
    Form {Enter Voucher - G/L Distribution - W0411K} - Repeat Count from "1"
      Detail Information - Repeat Count from "1"
        Type to Grid [Account Number] Value from ..\Random JE Account Numbers.atd
      >>INSERT
```

Having entered data to the Account Number grid column in the Enter Voucher - G/L Distribution form, and distributed the amount from the Enter Voucher - Payment Information to the indicated account number, you update the database and move back to the Enter Voucher - Payment Information form.

7. In the command menu, click Press Toolbar Button.
8. Choose Standard Button.
9. Choose OK.
10. In the Next Form list, choose Enter Voucher - Payment Information and click the Insert button.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
  Declare <Previous Document #>   Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        Type to [Long Address Number] Value from ..\Vendors.atd
        Type to [Business Unit] Value from "1"
        Type to [DateForGLandVoucherJULIA] Value from "063005"
        Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
      Detail Information - Repeat Count from "5"
        Type to Grid [Gross Amount] Value from ..\Random Dollar Amounts.atd
        Type to Grid [Remark] Value from ..\LWL-Random Remarks.atd
      Press Button {OK}
    Form {Enter Voucher - G/L Distribution - W0411K} - Repeat Count from "1"
      Detail Information - Repeat Count from "1"
        Type to Grid [Account Number] Value from ..\Random JE Account Numbers.atd
      Press Button {OK}
    >>INSERT
```

Setting the Value of a Variable

Having returned to an Enter Voucher - Payment Information form, you want to retrieve the previous document number and use it later to search in the Supplier Ledger Inquiry form for the data you input to the Gross Amount and Remarks grid columns in the Enter Voucher - Payment Information form. You want not only to retrieve, but also to store the document number, which you can do by assigning its value to the variable you declared earlier. Remember that the variable you declared earlier has only a name - Previous Document #. Its value must be derived from a source of your choosing.

▶ To set the value of a declared variable

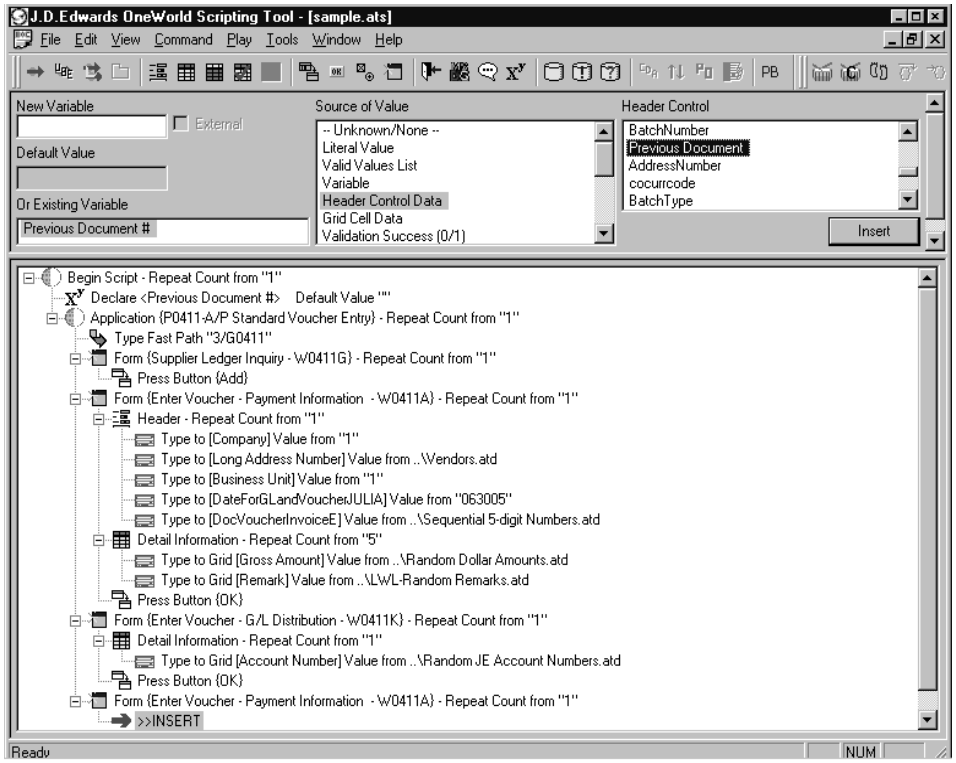
1. Click the Variables button in the cool bar.
2. In the Existing Variable list, choose the name of the variable you declared earlier in the script.
3. In the Source of Value list, choose from:
   - Literal Value
   - Valid Values List
   - Variable
   - Header Control Data
   - Grid column Data

In this sample script, you want to retrieve and store the previous Enter Voucher - Payment Information document number. Because the previous document number is found in a header control, you choose Header Control Data as the source of value.

4. Choose or type a value.

   In this case, OneWorld Scripting Tool populates the value selection list with the names of the header controls in the Enter Voucher - Payment Information form. The value selection list is therefore captioned Header Control. You choose Previous Document.

5. Click the Insert button.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Insert
Ready
NUM

New Variable

□ External

Default Value

Or Existing Variable: Previous Document #

Source of Value:
- -- Unknown/None --
- Literal Value
- Valid Values List
- Variable
- Header Control Data
- Grid Cell Data
- Validation Success (0/1)

Header Control:
- BatchNumber
- Previous Document
- AddressNumber
- cocurrcode
- BatchType

```
⊟ Begin Script - Repeat Count from "1"
    Declare <Previous Document #>    Default Value ""
  ⊟ Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
      Type Fast Path "3/G0411"
    ⊟ Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
        Press Button {Add}
    ⊟ Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      ⊟ Header - Repeat Count from "1"
          Type to [Company] Value from "1"
          Type to [Long Address Number] Value from ..\Vendors.atd
          Type to [Business Unit] Value from "1"
          Type to [DateForGLandVoucherJULIA] Value from "063005"
          Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
      ⊟ Detail Information - Repeat Count from "5"
          Type to Grid [Gross Amount] Value from ..\Random Dollar Amounts.atd
          Type to Grid [Remark] Value from ..\LWL-Random Remarks.atd
        Press Button {OK}
    ⊟ Form {Enter Voucher - G/L Distribution - W0411K} - Repeat Count from "1"
      ⊟ Detail Information - Repeat Count from "1"
          Type to Grid [Account Number] Value from ..\Random JE Account Numbers.atd
        Press Button {OK}
    ⊟ Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      >>INSERT
```

You have now assigned a value to the variable that you named Previous Document #. You have told OneWorld Scripting Tool to derive that value from the header control Previous Document in the OneWorld form Enter Voucher - Payment Information. OneWorld Scripting Tool stores that value in your declared variable. Since you earlier made the variable global, you can use this value at any point in the script you choose.

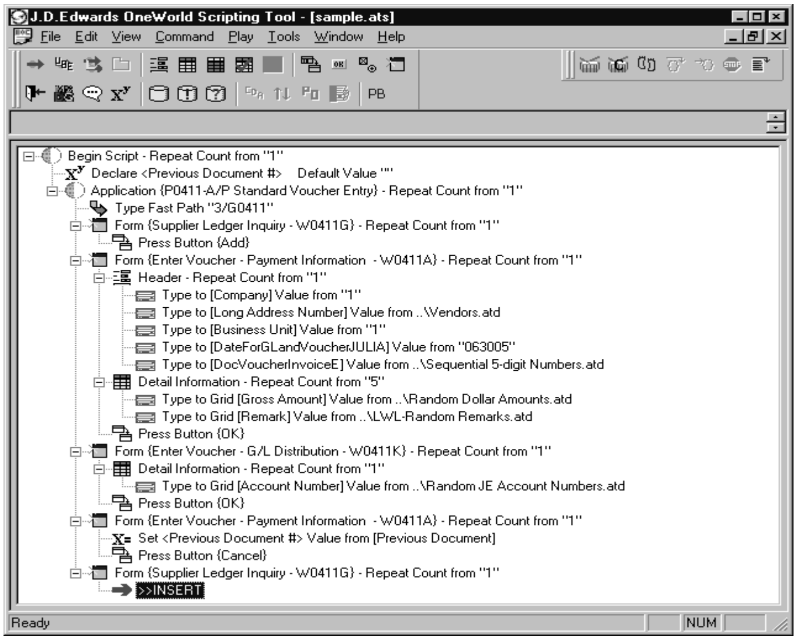
Returning to a Previous OneWorld Form

Now you want to take the value of the variable and enter it to the QBE line in a Supplier Ledger Inquiry form. In the sample script, clicking the Cancel button returns you to the Supplier Ledger Inquiry form. To return to a previous OneWorld form, you choose Press Toolbar Button from the command menu and choose Cancel from the Press Standard Button options in the Button list. You then choose the next OneWorld form that appears when you press the cancel button. Therefore, you choose Supplier Ledger Inquiry from the Next Form list.

▶ To return to a previous OneWorld form

1. In the command menu of the OneWorld Scripting Tool form, click Press Toolbar Button.
2. Choose the Cancel option from Press Standard Button in the Button list.
3. Select a form from the Next Form list, in this case Supplier Ledger Inquiry.
4. Click the Insert button.

   In OneWorld, the Supplier Ledger Inquiry form becomes active. In the OneWorld Scripting Tool script pane, the Form command line shows Supplier Ledger Inquiry.



```
Begin Script - Repeat Count from "1"
  Declare <Previous Document #>    Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        Type to [Long Address Number] Value from ..\Vendors.atd
        Type to [Business Unit] Value from "1"
        Type to [DateForGLandVoucherJULIA] Value from "063005"
        Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
      Detail Information - Repeat Count from "5"
        Type to Grid [Gross Amount] Value from ..\Random Dollar Amounts.atd
        Type to Grid [Remark] Value from ..\LWL-Random Remarks.atd
      Press Button {OK}
    Form {Enter Voucher - G/L Distribution - W0411K} - Repeat Count from "1"
      Detail Information - Repeat Count from "1"
        Type to Grid [Account Number] Value from ..\Random JE Account Numbers.atd
      Press Button {OK}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Set <Previous Document #> Value from [Previous Document]
      Press Button {Cancel}
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      >>INSERT
```

Entering Data to a QBE Line

Because you have assigned a value to the declared variable, Previous Document #, you can now use the value that you stored. Remember that the value is the document number of the Enter Voucher - Payment Information form to whose Gross Amount and Remarks grid columns you scripted inputs.

You decide to enter the stored value, the previous document number, to the QBE line of the Supplier Ledger Inquiry form, then script pressing the Find button to retrieve the values that you entered to the grid in the Enter Voucher - Payment Information form.

▶ To enter data to a QBE line

1. In the command menu of the OneWorld Scripting Tool form, click Set QBE Cell Value.
2. Choose a grid column from the Grid Column list, in this case Document Voucher Invoice Entry.
3. Choose Variable from the Source of Input list.
4. In the value selection list, choose the name of the variable you declared. Remember that it also now contains the value that you set.
5. Click the Insert button.

   OneWorld Scripting Tool inputs the previous Enter Voucher - Payment Information document number, which you stored in the variable you named Previous Document #, to the QBE line of the grid, in the Document Voucher Invoice Entry column.

Finding Records

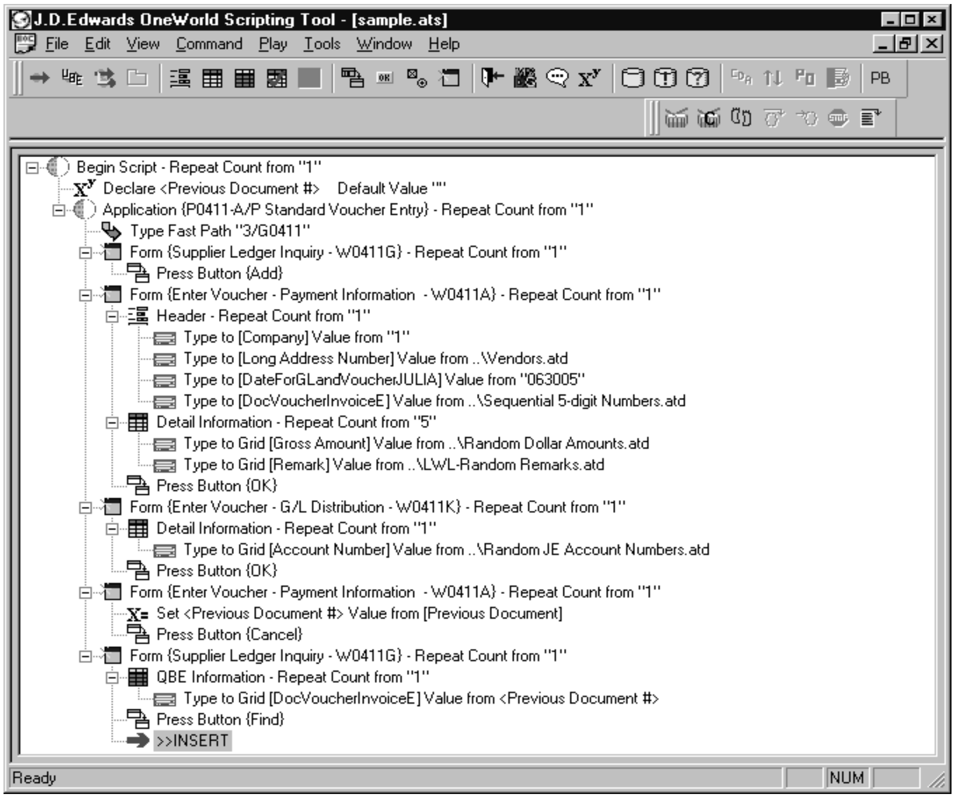
To find the values that you entered in the Enter Voucher - Payment Information form, using its document number, which you have typed to the Document Voucher Invoice Entry column, you must run a database query by pressing the Find button. Because your objective here is solely to retrieve the records, you do not choose another form from the Next Form list. Rather, you simply click the Insert button.

▶ To find records

1. In the command menu of the OneWorld Scripting Tool form, click Press Toolbar Button.
2. Choose Press Standard Button in the Button list.
3. Choose Find.

   It’s not necessary to make a selection from the Next Form list because you are remaining at the Supplier Ledger Inquiry form.

4. Click the Insert button.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
  Declare <Previous Document #>    Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        Type to [Long Address Number] Value from ..\Vendors.atd
        Type to [DateForGLandVoucherJULIA] Value from "063005"
        Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
      Detail Information - Repeat Count from "5"
        Type to Grid [Gross Amount] Value from ..\Random Dollar Amounts.atd
        Type to Grid [Remark] Value from ..\LWL-Random Remarks.atd
      Press Button {OK}
    Form {Enter Voucher - G/L Distribution - W0411K} - Repeat Count from "1"
      Detail Information - Repeat Count from "1"
        Type to Grid [Account Number] Value from ..\Random JE Account Numbers.atd
      Press Button {OK}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Set <Previous Document #> Value from [Previous Document]
      Press Button {Cancel}
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      QBE Information - Repeat Count from "1"
        Type to Grid [DocVoucherInvoiceE] Value from <Previous Document #>
      Press Button {Find}
      >>INSERT
```

The grid now fills with the voucher entries that relate to the document number you input to the Document Voucher Invoice Entry grid column of the QBE line.

Selecting Records and Deleting Them from the Database

Now you want to script the deletion of records from the database. These records are represented in the detail area of the Supplier Ledger Inquiry form because you scripted pressing the Find button in the previous task, Finding Records. To delete these records, you must select them, then script clicking the Delete button.

The records you select to delete in this task are actually deleted when you click the Insert button and save the script, just as they would be in a live OneWorld session. Before you perform this task, make sure you are in a test environment and do not click the Insert button until you are sure that you have selected the correct records to delete. Note that OneWorld Scripting Tool automatically handles the Confirm Delete dialog box that appears in OneWorld when you have selected a grid line for deletion by clicking the OK button; you are not given the opportunity to click OK or Cancel. If you think you have made an error, delete the command line or exit the script without saving the command.

▶ To select records and delete them from the database

1. In the command menu of the OneWorld Scripting Tool form, click Select Grid Line.
2. In the command pane, choose from the Source of Row Number list.
3. Type a literal value or choose a valid values list or variable in the value selection list. In this case, type a literal value of 1.

   By doing so, you choose the row that contains the record that you want to delete.

4. Click the single click grid row option in the Operation list.

   This command highlights the row in the detail area of the grid.

5. Click the Insert button.
6. In the command menu, click Press Toolbar Button.
7. Choose Press Standard Button in the Button list.
8. Choose Delete.
9. Click the Insert button.

   OneWorld Scripting Tool automatically clicks OK on the Confirm Delete form that appears in OneWorld and deletes the record that you selected. You can repeat this command as many times as necessary to delete any and all records that you want to delete.

Exiting the Script

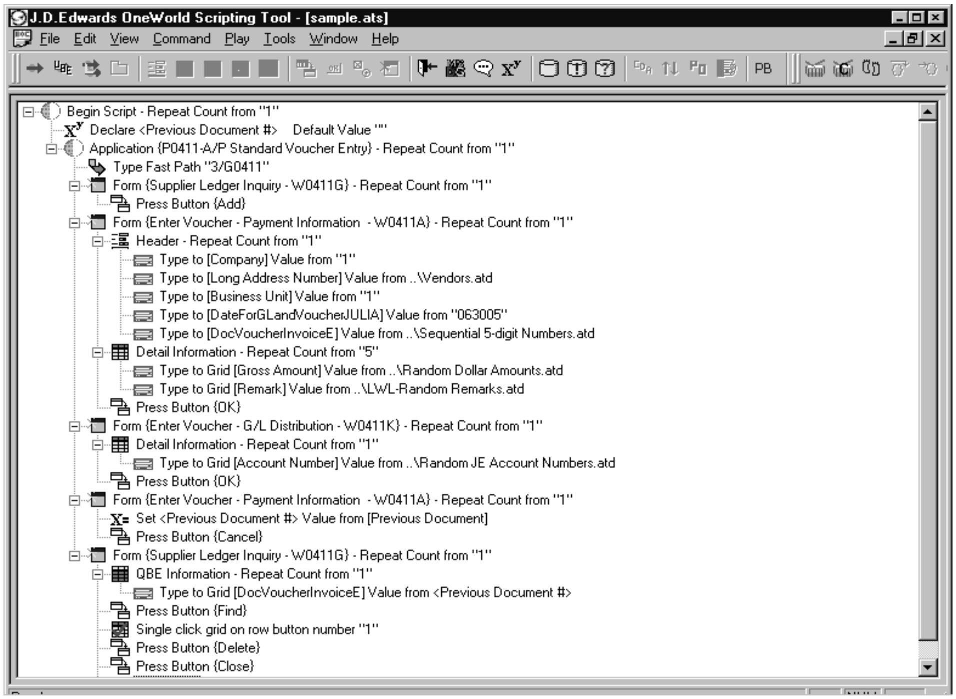
At the conclusion of your scripting, you exit all open applications and return to OneWorld Explorer. For the sample script, you press the Cancel button. Because you started in Supplier Ledger Inquiry, this command allows you to exit A/P Standard Voucher Entry and complete the script.

▶ To exit the script

1. In the command menu of the OneWorld Scripting Tool form, click Press Toolbar Button.
2. Choose Standard Button.
3. Choose Cancel.

   Since you are exiting the application, do not make a choice from the Next Form list.

4. Click the Insert button.


J.D.Edwards OneWorld Scripting Tool - [sample.ats]
File Edit View Command Play Tools Window Help
PB

```
Begin Script - Repeat Count from "1"
  Declare <Previous Document #>    Default Value ""
  Application {P0411-A/P Standard Voucher Entry} - Repeat Count from "1"
    Type Fast Path "3/G0411"
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      Press Button {Add}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Header - Repeat Count from "1"
        Type to [Company] Value from "1"
        Type to [Long Address Number] Value from ..\Vendors.atd
        Type to [Business Unit] Value from "1"
        Type to [DateForGLandVoucherJULIA] Value from "063005"
        Type to [DocVoucherInvoiceE] Value from ..\Sequential 5-digit Numbers.atd
      Detail Information - Repeat Count from "5"
        Type to Grid [Gross Amount] Value from ..\Random Dollar Amounts.atd
        Type to Grid [Remark] Value from ..\LWL-Random Remarks.atd
      Press Button {OK}
    Form {Enter Voucher - G/L Distribution - W0411K} - Repeat Count from "1"
      Detail Information - Repeat Count from "1"
        Type to Grid [Account Number] Value from ..\Random JE Account Numbers.atd
      Press Button {OK}
    Form {Enter Voucher - Payment Information  - W0411A} - Repeat Count from "1"
      Set <Previous Document #> Value from [Previous Document]
      Press Button {Cancel}
    Form {Supplier Ledger Inquiry - W0411G} - Repeat Count from "1"
      QBE Information - Repeat Count from "1"
        Type to Grid [DocVoucherInvoiceE] Value from <Previous Document #>
      Press Button {Find}
      Single click grid on row button number "1"
      Press Button {Delete}
      Press Button {Close}
```

Note: By clicking Cancel, you return to OneWorld Explorer or to ActivEra Solution Explorer. You should end each scripting session by doing so, particularly if you intend to create additional scripts. Leaving OneWorld windows open can impede playback of your scripts and make it difficult to write additional scripts.

5. In the menu bar, click File.
6. Type a file name
7. Click Save.

You have completed the sample script. During the course of creating the script, you launched an application, chose a form and version associated with the application and moved between forms by pressing buttons. You typed data to header controls, grid columns, and QBE lines. You derived those data from literal values that you typed to the value selection list and from valid values lists that you created, then chose from the value selection list. You added the data to the database, retrieved it, and deleted it. You declared a variable, set its value, and used it as a source of input to a QBE line. Finally, you canceled out of the application and saved the script.

Storing Scripts and Test Results

Storing Scripts and Test Results

Scripts that you write are reusable, dynamic objects whose usefulness can extend far beyond the time that you complete them. OneWorld Scripting Tool not only allows you to write and run scripts; it allows you to make those scripts part of a larger base of knowledge about OneWorld and to manage batch testing of scripts.

The script repository is a key component of the OneWorld Scripting Tool knowledge base. The repository is a database of scripts that you can add to or draw from. The database is stable because reposited scripts are controlled copies that can only be changed by the owner or an administrator with permissions. The database is varied because many people with different areas of OneWorld expertise might contribute to it. Finally, the database is organized because you can assign defining properties to each script that you reposit. These properties help to categorize your script by application, for example.

Capturing and storing test results is another important way OneWorld Scripting Tool allows you to build a knowledge base about OneWorld. If you configure OneWorld Scripting Tool to capture playback results, it generates an event stream during playback. The event stream is a chronological, time-stamped record of OneWorld Scripting Tool and OneWorld events that occur during playback. OneWorld Scripting Tool stores these test results locally and in a repository, the OneWorld F98214 table. You can use these results to troubleshoot OneWorld processes. For example, you might identify a processing bottleneck that occurred or isolate a OneWorld error message.

The results repository is a pivotal object in the J.D. Edwards automated testing scheme. You can import an event stream from the repository to J.D. Edwards OneWorld Analyzer Tool to conduct a more detailed analysis of playback events. You might also import an event stream to the Virtual Script Editor, which is part of J.D. Edwards OneWorld Virtual User Tool. Using the Virtual Script Editor, you can generate from the event stream a virtual script, which you can play back to simulate multiple OneWorld users on a single workstation.

Finally, OneWorld Scripting Tool helps you to manage testing your scripts. Using the OneWorld Scripting Tool Test Manager, you can create playlists of locally saved and reposited scripts and conduct batch testing, which frees you from the time-consuming chore of running one test at a time. You might use Test Manager to conduct testing of an entire suite of applications in one session.

This section discusses the following script management topics:

❑ Understanding the script repository

❑ Working with the script repository

- ❑ Understanding script reporting
- ❑ Understanding OneWorld Scripting Tool Test Manager
- ❑ Managing script testing

See Also

- *OneWorld Analyzer Tool Guide*
- *OneWorld Virtual User Tool Guide*

Understanding the Script Repository

The script repository functions in some ways like a library. It is a centralized location where you can find and retrieve materials, in this case, scripts. You are able to search for scripts in the repository by browsing through all the scripts that are available, or by focusing your search on, for example, scripts that tested a particular suite of applications. You might want simply to view a script or to see how it runs. In this case, you can return it to the repository in the condition that you found it.

In addition, much like a library of books, the script repository acquires new materials. Each time you or someone else creates a script, OneWorld Scripting Tool gives you the opportunity to assign distingushing properties to it, and then add it to the repository, from which it can be retrieved and viewed by others. Scripts can circulate freely among those who have access to the system, or you can assign levels of security to scripts to restrict their circulation.

You can also use the repository differently than you would a library in that you can change the materials that you remove from it. For example, you check a script out of the repository with the intention not only of viewing it, but also with the intention of making changes to it. Thus, the OneWorld Scripting Tool script repository is more than a simple storage unit that holds old scripts and acquires new ones. It also permits a dynamic interchange between those that use it.

In sum, you can use the script repository to build a database of scripts. You use OneWorld Scripting Tool to:

- Categorize scripts according to a set of user-definable criteria
- Identify scripts with unique names
- Add scripts to the repository
- Browse for scripts
- Check scripts in or out of the repository
- Retrieve copies of scripts
- Modify scripts
- Assign security to scripts
- Track changes made to scripts
- Identify scripts that are included in other scripts

This chapter covers the script repository topics:

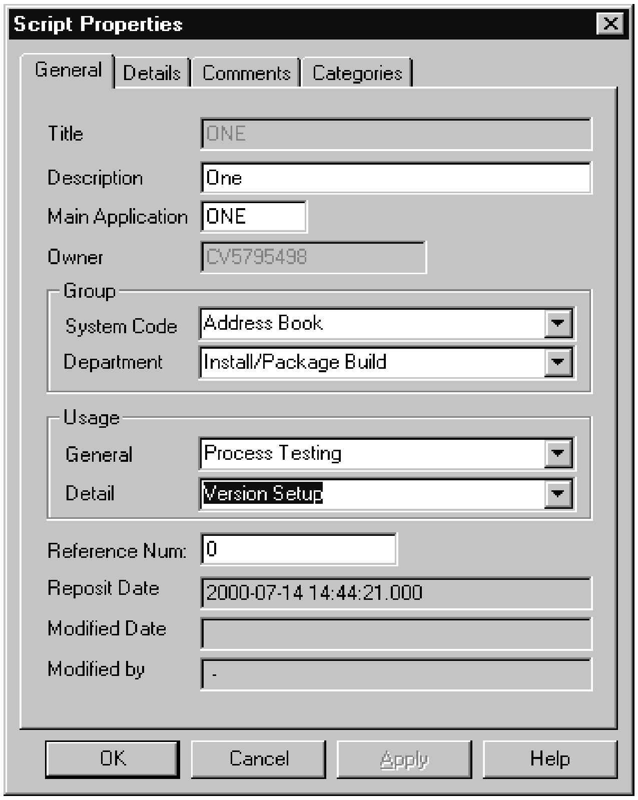
- ❑ Script categorization
- ❑ Property pages for scripts
- ❑ Naming conventions for saved scripts
- ❑ Add to Repository command
- ❑ Browse Repository Scripts command
- ❑ Script deletion
- ❑ Get Copy command
- ❑ Check Out command
- ❑ Undo Checkout command
- ❑ My Checkouts
- ❑ Check In command
- ❑ Query by include

Script Categorization

Much as a librarian would in adding a book to a library's collection, when you add a script to the repository, you assign it properties, such as title, description, the application the script tests, the purpose of the test, and so on. The properties pages attached to each repository script provide important summary information for those who check out a script, and they provide a way for you to categorize scripts, making them easier to find.

Categorizing scripts also assists you when you want to run batch-test scripts using OneWorld Scripting Tool Test Manager. Using Test Manager, you can browse the repository for scripts in a particular category, add them to a playlist, and automatically play them back. For more information on Test Manager, see *Understanding OneWorld Scripting Tool Test Manager* and *Managing Script Testing* in this section.

The Script Properties form contains controls with scroll buttons.



Title: ONE
Description: One
Main Application: ONE
Owner: CV5795498

Group
System Code: Address Book
Department: Install/Package Build

Usage
General: Process Testing
Detail: Version Setup

Reference Num: 0
Reposit Date: 2000-07-14 14:44:21.000
Modified Date:
Modified by: -

The combo boxes contain user-defined values. You choose from these values to categorize your script. The use of a consistent set of user-defined values, maintained in the OneWorld UDC tables (F0004 and F0005), rather than individual user text entries, means that the information in the database is more consistent and reliable, and is easier to access by browsing.

You can add to the values that display in the combo boxes by using the Work with UDC Codes (P0004) OneWorld application. For information on adding UDC codes, see *OneWorld Tools Foundation.* The following table lists the relevant UDC codes used by OneWorld Scripting Tool to populate the combo boxes in the Script Properties form:

Product Code (in OneWorld application P0004)	**User–Defined Code (in OneWorld application P004)**	**Combo Box in Script Properties Form in OneWorld Scripting Tool**
98 (Technical Tools)	SY (System Code)	System Code control under General tab
H97 (Benchmarking/Performance)	DN (Department)	Department control under General tab
H97	GU (General)	General control under General tab

H97	DU (Detail)	Detail control under General tab
H97	OT (Other)	Test Case control under Details tab

Note: The user-defined combo box values also appear in the Select Script form and the Add Script to Repository form. See the discussions of these forms in this chapter for details.

Property Pages for Scripts

Before you reposit a script, you should complete property pages that provide fundamental information about it, such as title, description, owner, application the script tests, and so on. Completing property pages can also help to classify the script as part of a large-scale testing effort. For example, you can designate the following script properties that make the script part of a larger suite of scripts:

- System code
- Department
- General use, such as benchmarking
- Detailed use, such as batch applications

You document the properties of your script by making entries to the Script Properties form. When you save the script, OneWorld Scripting Tool saves the data you entered to the property pages along with the script.

When you reposit the script, OneWorld Scripting Tool saves the property page data in the database. This data loads when you check out a script from the repository, overwriting any subsequent property page changes you might have made to the local script.

The Script Properties tabs include:

❑ General

❑ Details

❑ Comments

❑ Categories

General Tab

Under the General tab of the Script Properties form, you find a series of controls to which you enter data that defines your script.

Your entries to the controls under this tab provide baseline information about the script and its origins. There are nine controls under the General tab of the Script Properties form:

Control under the General tab	Purpose
Title	Script title that OneWorld Scripting Tool automatically enters when you save the script. Title can be changed after you check in the script.
Description	Brief description of the script, such as the OneWorld function that it tests.
Main Application	Identification of the primary application that your script tests.
Owner	Script owner, automatically identified as the person who reposits the script. Owner can be after the script is checked in.
System Code	A user-definable list of the reporting system code.
Department	A user-definable list of the department or group names.
General	The general testing purpose of the script, such as benchmarking. The values for this parameter are user-definable.
Detail	The specific testing purpose of the script, such as batch applications. The values for this parameter are user-definable.
Reference Number	A code that identifies the script. For example, you might use it to enter a SAR number that the script tests, or a regression test that the script runs to verify the correction of a bug.

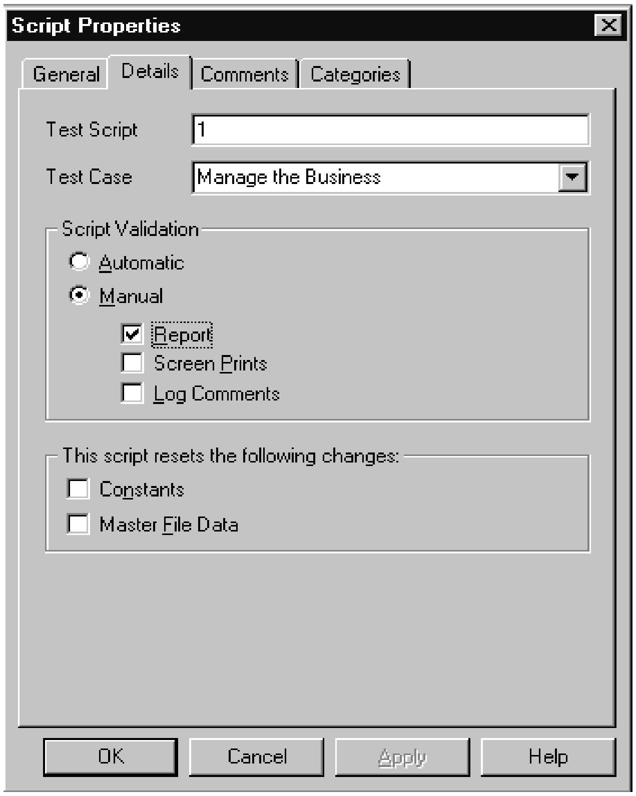
Note: After you reposit the script, get it from the repository, and view its properties, controls under the General tab display the reposit date and the last person to open the script, as well as the time and date that the script was opened.

Details Tab

Under the Details tab you can enter information that defines how the script fits into a test management scheme as well as validation information and data-resetting information.

Three controls allow you to enter quality assurance-related data that is useful large-scale testing efforts:

- Test Suite: You enter the collection of related scripts to which yours belongs, such as Tools Applications.
- Test Case: You enter the specific function you are testing, such as turning on the Address by Effective Date feature. The values for this parameter are user-definable.



Test Script: 1

Test Case: Manage the Business

Script Validation

○ Automatic

● Manual

☑ Report

☐ Screen Prints

☐ Log Comments

This script resets the following changes:

☐ Constants

☐ Master File Data

Two radio button options allow you to identify whether script validation occurs automatically or if you need to review the script output manually to determine if it ran successfully:

- Automatic
- Manual

You choose the automatic validation option if a successful script run automatically means that the functions you tested worked, and there is no further need for you to review the results. On the other hand, in some cases, such as when you test UBEs, you must manually review the output of the script run to determine if it was successful. For example, you might need to make sure that a UBE report generated successfully.

If you choose the Manual option under the Details tab, OneWorld Scripting Tool enables three additional options, any or all of which you can choose:

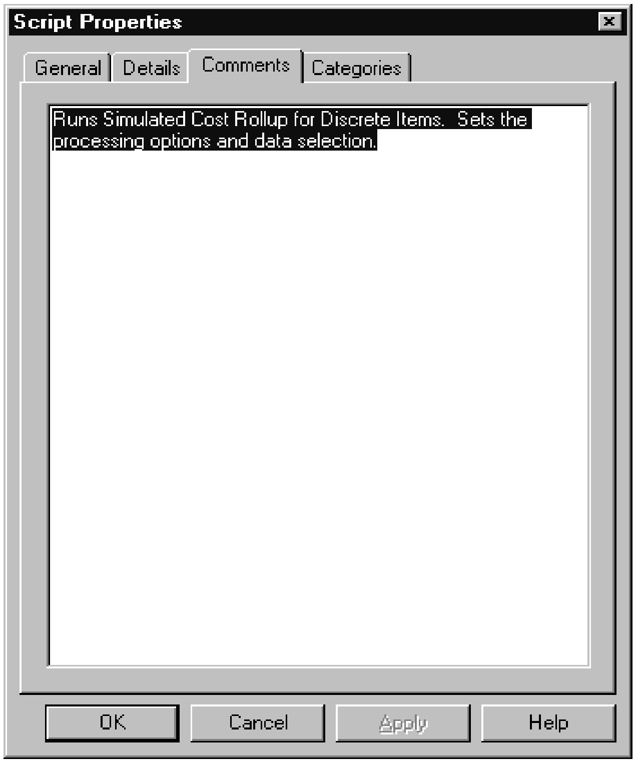
- Report
- Screen Prints
- Log Comments

Choosing one or more of these options reminds those who run the script to manually review the chosen output after the script runs.

Finally, you can indicate, using two more options, whether your script resets changes you made to constants or to master file data, such as additions to or deletions from Item Master.

Comments Tab

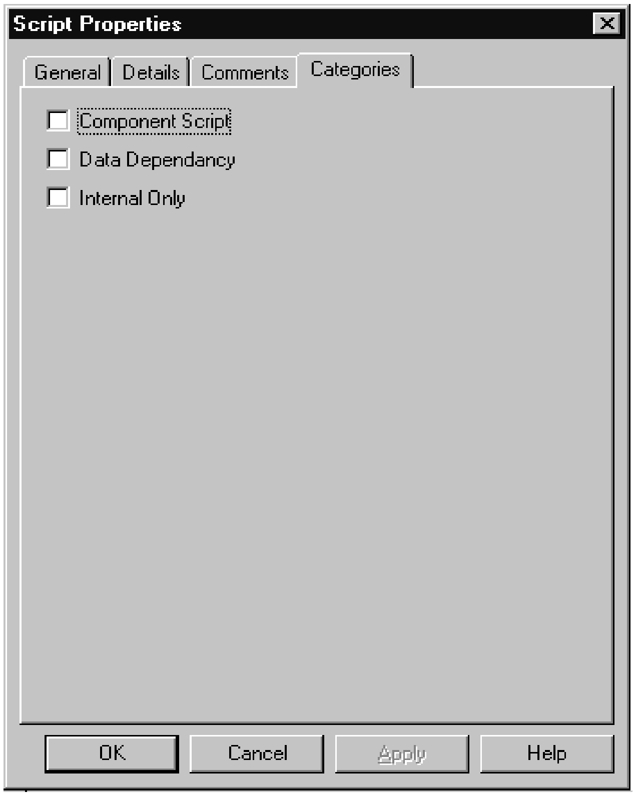
Under the Comments tab, you can enter additional descriptive information about the test and state in detail the intention of the script and any other information you deem relevant.



Runs Simulated Cost Rollup for Discrete Items. Sets the processing options and data selection.

Categories Tab

The Categories tab allows you to further define the object of your testing. Your system administrator creates user-defined testing categories, which display in the form as options. For example, you might define a category of testing as package verification. Options under the Categories tabs, such as Daily Build or Weekly Package, could indicate the type of verification testing the script performs.



□ Component Script
□ Data Dependancy
□ Internal Only

Note: Reserves the first three fields.

Naming Conventions for Saved Scripts

There are no predetermined rules for naming scripts that you intend to add to the repository. However, you should choose a specific naming convention for all scripts. For example, you might give your script a title that identifies the application, release, or function that it tests. In addition, you might want to specify whether the test is new or a retest. Following a naming convention also can make identifying the purpose of a script easier for those who retrieve it from the repository. The most important purpose of assigning properties to the script is to accurately subcategorize it, and adhering to a naming convention merely helps to accomplish that goal.

See Also

- *Script Retention and Reuse*

Add to Repository Command

Once you have created a script, established its properties, given it a unique title, and saved it locally, you can add it to the repository, using the Add Script to Repository form. You can make changes to the script's properties using this form before you add the script to the repository, but you cannot specify the owner, as OneWorld Scripting Tool automatically assigns the owner ID, closes the script to prevent further changes, and adds it to the repository.

Browse Repository Scripts Command

The Browse Repository Scripts command allows you to search for scripts in the repository. You can make your search as narrow or as broad as you like, and once you have found the script that you are looking for, you can get a copy of it, or you can check it out of the repository.

You use two forms to complete the Browse Repository Scripts command:

❑ Select Script form

❑ Browse Scripts form

You use the Select Script form to establish search criteria. The Browse Scripts form contains the titles of and information about the scripts in the repository whose properties match the criteria you specified in the Select Script form.

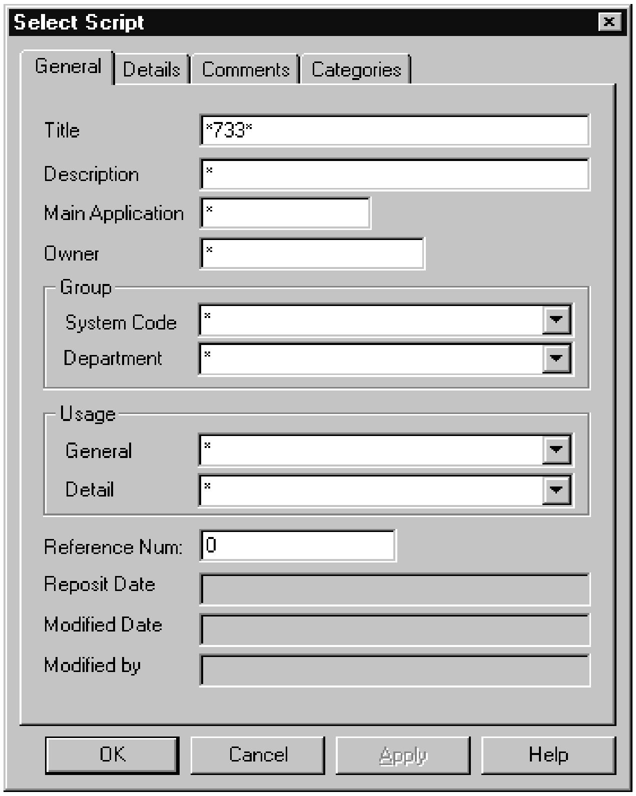
Select Script Form

The Select Script form contains the same tabs and controls as does the Script Properties form. However, it functions as a query form rather than as a form for entering script properties.

You can click any of the tabs, except Comments. and enter data to the controls or choose options to establish search criteria for a type of script that you are looking for, or you can enter the exact title of a script. OneWorld Scripting Tool matches the criteria you set to the entries you and others made to the Script Properties pages before adding scripts to the repository. If you do not enter any control information, OneWorld Scripting Tool includes all the scripts in the repository.

All of the controls under the General and Detail tabs in the Select Script form, with the exception of the SAR number and Script Origin controls, contain asterisks that serve as search wildcards, which you can use to have OneWorld Scripting Tool include all scripts in its search.

You can use the asterisks alone or with an entry to a control. For example, if you want to find all scripts that tested the 73.3 release, you enter *733* to the Title control. OneWorld Scripting Tool includes in its search all scripts that contain 733 in the title, regardless of any words that come before or after 733.



Title	*733*
Description	*
Main Application	*
Owner	*
System Code	*
Department	*
General	*
Detail	*
Reference Num:	0
Reposit Date	
Modified Date	
Modified by	

If you make an entry to a control, OneWorld Scripting Tool automatically appends the wildcard to the text string. For example, if you enter P0911 to the Main Application control, OneWorld Scripting Tool returns all scripts with a main application property that includes P0911: P0911A, P0911B, and so on.

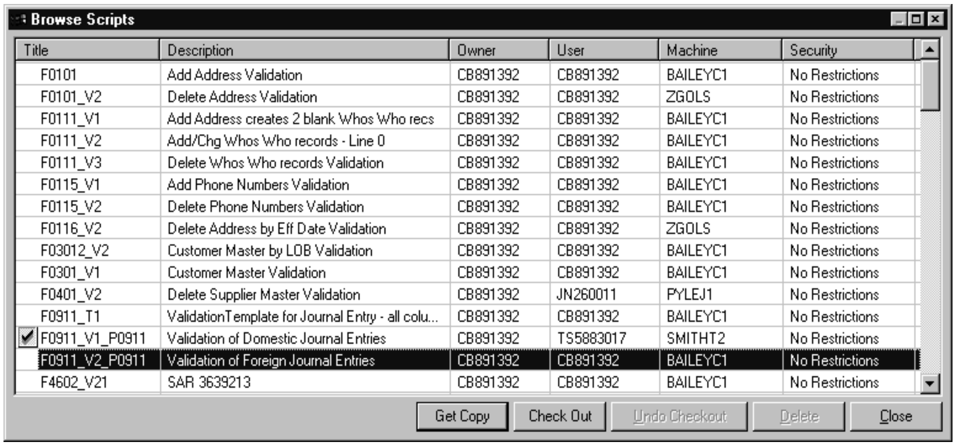
The Reference Number control contains a 0, indicating that you have not entered a SAR number. If you leave the 0 in this control, OneWorld Scripting Tool does not use a reference number as a search criterion and includes all reference numbers, such as SARs, in its search.

If you enter a reference number, you limit your search to those scripts that tested a particular SAR or tests defined by another reference number.

Browse Scripts Form

When you click OK on the Select Script form, the Browse Scripts form appears. This form contains summaries of any scripts that match the data you specified in the Select Script form. The Browse Scripts form contains six column headings that identify a script:

- Title
- Description
- Owner
- User
- Machine
- Security



	Title	Description	Owner	User	Machine	Security
	F0101	Add Address Validation	CB891392	CB891392	BAILEYC1	No Restrictions
	F0101_V2	Delete Address Validation	CB891392	CB891392	ZGOLS	No Restrictions
	F0111_V1	Add Address creates 2 blank Whos Who recs	CB891392	CB891392	BAILEYC1	No Restrictions
	F0111_V2	Add/Chg Whos Who records - Line 0	CB891392	CB891392	BAILEYC1	No Restrictions
	F0111_V3	Delete Whos Who records Validation	CB891392	CB891392	BAILEYC1	No Restrictions
	F0115_V1	Add Phone Numbers Validation	CB891392	CB891392	BAILEYC1	No Restrictions
	F0115_V2	Delete Phone Numbers Validation	CB891392	CB891392	BAILEYC1	No Restrictions
	F0116_V2	Delete Address by Eff Date Validation	CB891392	CB891392	ZGOLS	No Restrictions
	F03012_V2	Customer Master by LOB Validation	CB891392	CB891392	BAILEYC1	No Restrictions
	F0301_V1	Customer Master Validation	CB891392	CB891392	BAILEYC1	No Restrictions
	F0401_V2	Delete Supplier Master Validation	CB891392	JN260011	PYLEJ1	No Restrictions
	F0911_T1	ValidationTemplate for Journal Entry - all colu...	CB891392	CB891392	BAILEYC1	No Restrictions
☑	F0911_V1_P0911	Validation of Domestic Journal Entries	CB891392	TS5883017	SMITHT2	No Restrictions
	F0911_V2_P0911	Validation of Foreign Journal Entries	CB891392	CB891392	BAILEYC1	No Restrictions
	F4602_V21	SAR 3639213	CB891392	CB891392	BAILEYC1	No Restrictions

The script owner is the person designated in the Script Properties form. The user is the last person who checked out or checked in the script. The machine identifies the workstation that the user was on at the time of script checkout. Security indicates the level of security that the owner attaches to the script. The security levels and their meanings are:

Security Level of Reposited Script	**Meaning**
No restrictions	Anyone can check out and make changes to the script; all properties can be changed except security level.
Owner Locked	Anyone can check out and make changes to the script, but owner and security level can't be changed.
No Checkout/in	Those who do not own the script can only get a copy of it; if any changes are made, they can only be saved locally.
No Access	Those who do not own the script can only see that it resides in the repository.

The form also contains the following buttons:

- Get Copy allows you to secure a copy of a script from the repository
- Check Out allows you to check out a script from the repository
- Undo Checkout allows you to undo a script checkout
- Delete allows you to delete a script from the repository, provided you are authorized to do so
- Cancel allows you to exit the Browse Scripts form

OneWorld Scripting Tool disables the Get Copy and Checkout buttons until you click the title of a script. Once you click a title, these two buttons are enabled.

You can view the properties of any script that you want to check out on the Repository Script Properties form, which contains the four tabs that you find in the Select Properties and Script Properties forms. You cannot change script properties using the Repository Script Properties form.

You can get a copy of or check out a single script. You can also get multiple copies or check out more than one script You can open the checked-out copy, check it in, or cancel out of the form, in which case the script remains checked out. If you select a mix of scripts that you checked out and scripts that someone else has checked out, you cannot open a script or check it in.

Script Deletion

You can delete from the repository any script that you you own, However, OneWorld Scripting Tool places several restrictions on your ability to delete scripts:

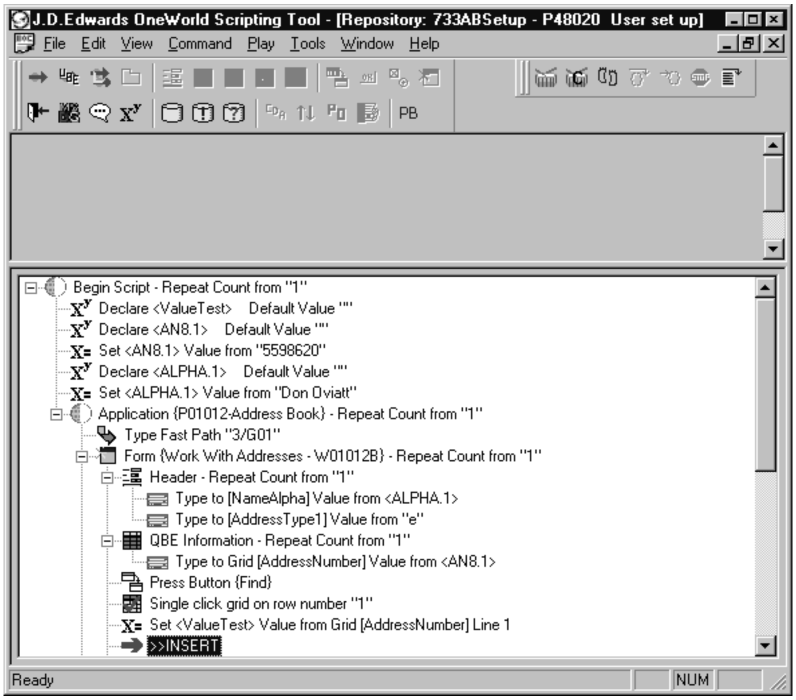
- You cannot delete a script that you do not own unless you have authorization.
- You can delete script you did not add but have authorization to only after entering a password that changes each day.
- You cannot delete a script that is included in another script.

Get Copy Command

While you are in the Browse Scripts form, you can get a copy of a script from the repository. You might run the Get Copy command when you want to view a particular script with the idea of using it as a template for another script or if you simply want to run the script. OneWorld Scripting Tool allows you to view and to run a copy of a script you get from the repository, but you can not make any permanent changes to it without first saving it as a local copy in your .ats directory.

OneWorld Scripting Tool enables the Get Copy button in the Browse Scripts form when you select a script that is not checked out. If you select a checked-out script, the Get Copy command is not available. Instead, you can use the Open command to access a copy of the checked-out script.

When you get a copy of a reposited script, the copy opens in OneWorld Scripting Tool. The form caption bar contains the word "Repository," the title and description of the reposited script.


J.D.Edwards OneWorld Scripting Tool - [Repository: 733ABSetup - P48020 User set up]
File Edit View Command Play Tools Window Help
PB
Ready
NUM

```
Begin Script - Repeat Count from "1"
    Declare <ValueTest>    Default Value ""
    Declare <AN8.1>    Default Value ""
    Set <AN8.1> Value from "5598620"
    Declare <ALPHA.1>    Default Value ""
    Set <ALPHA.1> Value from "Don Oviatt"
    Application {P01012-Address Book} - Repeat Count from "1"
        Type Fast Path "3/G01"
        Form {Work With Addresses - W01012B} - Repeat Count from "1"
            Header - Repeat Count from "1"
                Type to [NameAlpha] Value from <ALPHA.1>
                Type to [AddressType1] Value from "e"
            QBE Information - Repeat Count from "1"
                Type to Grid [AddressNumber] Value from <AN8.1>
            Press Button {Find}
            Single click grid on row number "1"
            Set <ValueTest> Value from Grid [AddressNumber] Line 1
            >>INSERT
```

If you make changes to the copy, then click Save, OneWorld Scripting Tool displays an error message advising you that the script is not checked out and that you must either check it out or save it as a local file for your changes to take effect.

Checkout Command

You might retrieve a script from the repository with the intention of making changes to it. To do so, you must check out the script, make and save the changes, and then check it back into the repository. OneWorld Scripting Tool returns the script to the repository with the changes intact. The repository now contains a new version of the script.

When you check out a script that has been saved and reposited OneWorld Scripting Tool checks out to you the latest version of the script. A script cannot be checked out to more than one person at a time, which prevents two or more people from making changes to the script simultaneously. If the script is already checked out from the repository, OneWorld Scripting Tool prevents you from checking out the script.

You can also run the Check Out command if you are browsing scripts. In the Select Script form, you limit your script search as much or as little as you want by choosing options and making entries under the various tabs.

Undo Checkout Command

You can undo your checkout command if, for example, you make changes to an existing script but decide that they are not the changes you want. If you check in the script, the changes take effect, and the repository has a new version of the script. If you undo the checkout command, none of the changes you made to the script while it was checked out take effect in the original version. To undo the checkout, you click File in the menu bar, then Repository, and Undo Check Out. A form confirms the undo checkout.

My Checkouts

You can find scripts you have checked out in the My Checkouts form. The My Checkouts form helps you keep track of the scripts that you have checked out so that you make sure to check them back in. The My Checkouts form contains the same headings that the Browse Script form contains. Each document icon next to the script title contains a green check mark, indicating that you have checked out the script on the machine you are currently using.

You cannot check out a script on one machine and check it in on another. The My Checkouts form shows all your current machine checkouts. If you made a checkout on a machine different from the one on which you are currently working, the document icon next to the script title contains a red X, indicating that you cannot check in the script from the current machine.

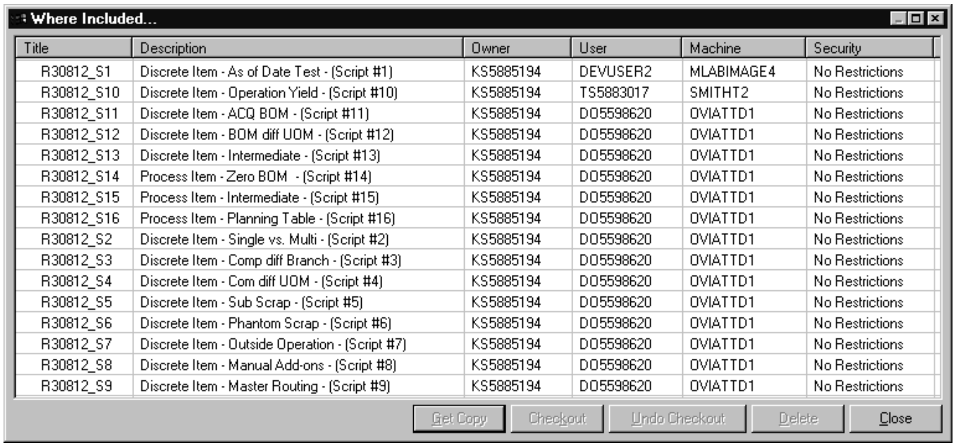
Check In Command

After you have checked out a script, you can work with it and make whatever changes that you want to add to the reposited copy. When you are satisfied with those changes, you must check in the script in order for them to take effect. With the script checked into the repository, the new version that you created is available to others who might want to view it or work with it. When you check in a script, it automatically closs and goes into the repository, just as when add a script to the repository.

Query by Include

You might want to know if a script is included with another script, because you can not delete an included script. You can search the repository for all scripts that include another script using the Where Included command.

When you enter the title of a script, OneWorld Scripting Tool displays on the Where Included form the titles and descriptions of any reposited scripts that include the script title.



Where Included...

Title	Description	Owner	User	Machine	Security
R30812_S1	Discrete Item - As of Date Test - (Script #1)	KS5885194	DEVUSER2	MLABIMAGE4	No Restrictions
R30812_S10	Discrete Item - Operation Yield - (Script #10)	KS5885194	TS5883017	SMITHT2	No Restrictions
R30812_S11	Discrete Item - ACQ BOM - (Script #11)	KS5885194	DO5598620	OVIATTD1	No Restrictions
R30812_S12	Discrete Item - BOM diff UOM - (Script #12)	KS5885194	DO5598620	OVIATTD1	No Restrictions
R30812_S13	Discrete Item - Intermediate - (Script #13)	KS5885194	DO5598620	OVIATTD1	No Restrictions
R30812_S14	Process Item - Zero BOM - (Script #14)	KS5885194	DO5598620	OVIATTD1	No Restrictions
R30812_S15	Process Item - Intermediate - (Script #15)	KS5885194	DO5598620	OVIATTD1	No Restrictions
R30812_S16	Process Item - Planning Table - (Script #16)	KS5885194	DO5598620	OVIATTD1	No Restrictions
R30812_S2	Discrete Item - Single vs. Multi - (Script #2)	KS5885194	DO5598620	OVIATTD1	No Restrictions
R30812_S3	Discrete Item - Comp diff Branch - (Script #3)	KS5885194	DO5598620	OVIATTD1	No Restrictions
R30812_S4	Discrete Item - Com diff UOM - (Script #4)	KS5885194	DO5598620	OVIATTD1	No Restrictions
R30812_S5	Discrete Item - Sub Scrap - (Script #5)	KS5885194	DO5598620	OVIATTD1	No Restrictions
R30812_S6	Discrete Item - Phantom Scrap - (Script #6)	KS5885194	DO5598620	OVIATTD1	No Restrictions
R30812_S7	Discrete Item - Outside Operation - (Script #7)	KS5885194	DO5598620	OVIATTD1	No Restrictions
R30812_S8	Discrete Item - Manual Add-ons - (Script #8)	KS5885194	DO5598620	OVIATTD1	No Restrictions
R30812_S9	Discrete Item - Master Routing - (Script #9)	KS5885194	DO5598620	OVIATTD1	No Restrictions

From the Where Included form, you can choose a script and get a copy of it or check it out. The script you select from this form is a master script, meaning it is the parent of the script for which you initially searched, as well as any other scripts that might be included with it. If no scripts appear in the Where Included form, you can delete the script on which you conducted the search.

Working with the Script Repository

The OneWorld Scripting Tool script repository provides a resource from which you can retrieve scripts for study, playback, and modification. You can add to the repository scripts that you create; others might retrieve copies to see what functions you tested or to use a script as a template for another script. In turn, you can retrieve others' scripts for the same purpose. You can also check out scripts from the repository, make modifications to them, and return the new version by checking it in. These changes might be important to reflect modifications to OneWorld functions.

The script repository works in conjunction with the other components of OneWorld Scripting Tool. After you have written a script, you can assign properties to it and save it locally. When you add it to the repository, you make it available to others in a centralized storage location, and it becomes a controlled version that can only be changed by following prescribed procedures.

This chapter covers the following tasks necessary for utilizing the script repository:

❑ Assigning properties to a script

❑ Adding a script to the repository

❑ Browsing for repository scripts

❑ Deleting a script from the repository

❑ Assigning security to a reposited script

❑ Getting a copy of a script

❑ Checking out a script

❑ Undoing script checkout

❑ Checking in a script

❑ Querying for included scripts

❑ Using a command line to load a repository script

Assigning Properties to a Script

Before you add a script to the repository, you might want to assign properties that remain with it when you save the script. Properties include the script title, description, main application tested, and other parameter values that you have assigned to it These identifying features make it easier for you and your colleagues to conduct searches of the repository for scripts of a specified type. You save the the property pages locally along with the script.

> **Note:** Remember that four controls under the General tab and one under the Details tab have scroll buttons you can click to access combo boxes. These combo boxes contain user-defined values. Choose from these values to help in categorizing your script.

▶ To assign properties to a script

1. From the OneWorld Scripting Tool form, open a script.
2. From the File menu, choose Properties.

   The Script Properties form appears.

3. In the Script Properties form, make entries to or choose any or all of the controls or options under each of the four tabs:
   - General
   - Details
   - Comments
   - Categories
4. Click OK.
5. From the File menu, click Save or Save As and assign to the script a title that follows the naming convention that your group or organization uses.

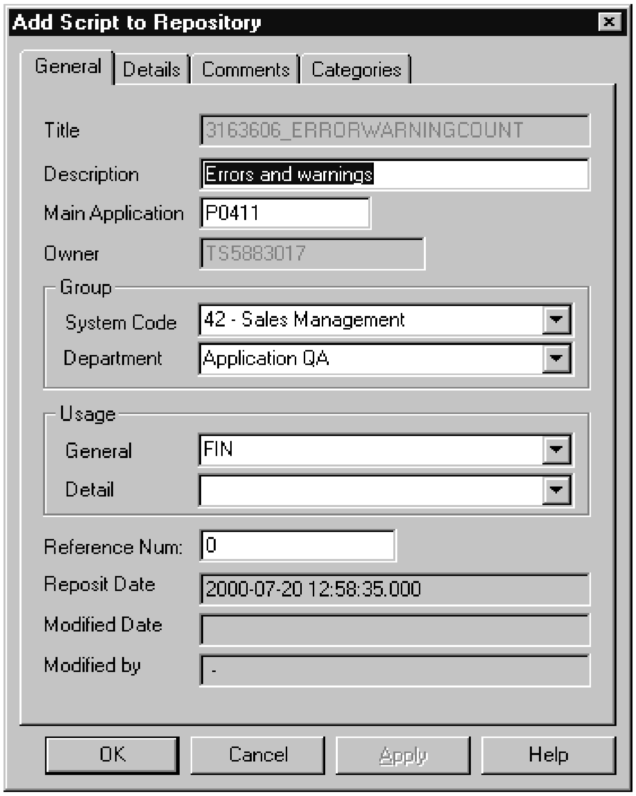
Adding a Script to the Repository

Once you have saved a script, along with any properties you've assigned, you can add it to the repository. The script must be open for you to do so. When you choose the Add to Repository command, OneWorld Scripting Tool allows you to assign script properties for the first time or to add to the properties you have already assigned. After you add the script to the repository, OneWorld Scripting Tool identifies the script by the title and description that you entered.

▶ To add a script to the repository

1. In the OneWorld Scripting Tool form, open the script that you want to add to the repository.
2. From the File menu, choose Repository and Add to Repository.

   The Add Script to Repository form appears.



Add Script to Repository

Title	3163606_ERRORWARNINGCOUNT
Description	Errors and warnings
Main Application	P0411
Owner	TS5883017
Group	
System Code	42 - Sales Management
Department	Application QA
Usage	
General	FIN
Detail	
Reference Num:	0
Reposit Date	2000-07-20 12:58:35.000
Modified Date	
Modified by	-

3. Enter any script properties that you want to assign to the script by making entries to the Add Script to Repository form and Click OK.

   OneWorld Scripting Tool automatically closes and reposits the script. A controlled copy of the script now exists in the repository. You can still make any changes you want to the local copy.

Browsing for Repository Scripts

You can browse the repository for scripts that you may want to run, use as a template, or modify. You can view all of the scripts in the repository, or enter to the Select Script form search criteria to locate scripts of a certain type. You make entries to the controls and choose options to establish search criteria. OneWorld Scripting Tool uses the criteria to display any scripts with matching properties.

▶ To browse for scripts

1. From the File menu of the OneWorld Scripting Tool form, choose Repository and Browse Repository Scripts.

   The Select Script form appears. The form contains the same four tabs that are in the Script Properties form and the Add Script to Repository form.

2. In the Select Script form, establish criteria for the kind of scripts you want.
3. Click OK.

   The Browse Scripts form appears, displaying the titles of any scripts that matched the criteria you specified in the Select Script form.

Deleting a Script from the Repository

You can delete a script from the repository, provided you are its owner or you have the proper authorization. As a precautionary measure, OneWorld Scripting Tool asks you to confirm the delete before you make it.

Note: You cannot delete a script if its included in another script.

▶ To delete a script from the repository

1. From the File menu of the OneWorld Scripting Tool form, choose Browse Repository Scripts.
2. In the Select Script form, enter your user ID to the Owner control under the General tab and click OK.

   Note: If you are authorized to delete scripts other than those you own, you can use other selection criteria by making entries to additional controls and options under the tabs in the Select Script form.

3. In the Browse Scripts form, choose the title of at least one script that you want to delete from the repository and click the Delete button.

   A OneWorld Scripting Tool form asks if you are sure that you want to delete the chosen script. If you click Yes, and the script is open, OneWorld Scripting Tool closes the script.

4. Click OK.

   Note: The deletion fails if the script you want to delete is included in another script.

Assigning Security to a Reposited Script

After you add a script to the repository, you can assign security to it, or you can leave it with no restrictions. Assigning security to a script restricts the ability of others to access and to make changes to the script.

You can assign security to a script if you are not its original owner, but only if the original script had no restrictions. To change the security in this situation, however, be sure to make yourself the owner of the script.

> **Note:** You can also assign security to a script from the My Checkouts form if you have checked the script out of the repository.

▶ To assign security to a reposited script

1. From the File menu of the OneWorld Scripting Tool form, choose Browse Repository Scripts.
2. In the Select Script form, enter your user ID to the Owner control under the General tab and click OK.
3. In the Browse Repository Scripts form, choose one or more scripts and right click.

   A pop-up menu containing four security level choices appears.

4. From the pop-up menu, click one of the following to choose a security level for the checked-out script:
   - No Restrictions
   - Owner Locked
   - No Checkout/in
   - No Access
5. Click the Check In button.

Getting a Copy of a Script

You can use the Browse Scripts form to get a copy of a script that you want to play back or to use as a template for creating another script. Important points to remember about script copies are:

- You can change the copy, but you can only save the changes to a separate local copy of the script, not to the reposited script.
- You cannot get copies of more than one script if one of the scripts you choose is checked out.

- If you choose only the checked-out script, you get a copy of the script, including the changes made since the last checkout.

▶ To get a copy of a script

1. From the File menu of the OneWorld Scripting Tool form, choose Repository and Browse Repository Scripts.

   The Select Script form appears.

2. In the Select Script form, establish criteria for the kind of scripts you want.
3. Click OK.

   The Browse Scripts form appears.

4. In the Browse Scripts form, choose at least one script.
5. Click the Get Copy button if the script you choose is not checked out. Click the Open button if the script is checked out.

Checking Out a Script

You might want to check out the original script that was added to the repository. Only one person at a time can check out a script. You can make changes to a script that you check out, then check it back into the repository. OneWorld Scripting Tool saves all changes, creating a new version, without asking you to add the script to the repository.

If someone has checked out a script, the document icon next to the script title in the Browse Scripts form contains a check mark or an X. An X appears if a script has been checked out to another machine. Note again that if you attempt to choose one or more scripts and one of them has already been checked out, OneWorld Scripting Tool disables the Get Copy and Check Out commands. If you choose the checked-out script only, you can open the script to check it into the repository or to undo the checkout.

▶ To check out a script

1. From the File menu of the OneWorld Scripting Tool form, choose Repository and Browse Repository Scripts.

   The Select Script form appears.

2. In the Select Script form, establish criteria for the kind of scripts you want.

   The Browse Scripts form appears.

3. Choose one or more scripts and click Check Out.

Undoing Script Checkout

You might want to undo a script checkout if, for example, you make changes to the script but then decide that you do not want the changes to take effect.

▶ To undo a script checkout

1. From the File menu of the OneWorld Scripting Tool form with a checked-out script open, choose Repository and Check In/Check Out.
2. Click Undo Checkout.

   A OneWorld Scripting Tool form appears asking you if you want to proceed.

3. Save any changes to the script that you want to preserve and click the Yes button.

Checking in a Script

If you check out a script from the repository and make changes to it, you check it back in if you want to reposit the new version of the script. OneWorld Scripting Tool automatically saves the changes. The new version returns to the repository.

▶ To check in a script

1. From the File menu of the OneWorld Scripting Tool form, with a checked-out script open, choose Repository and Check In/Check Out.

   Note: You can identify all the scripts you have checked out by clicking File, then Repository, and My Checkouts.

2. Choose Check In.

   The script automatically closes. OneWorld Scripting Tool checks the new script version into the repository.

Querying for Included Scripts

You can search the repository for any scripts in which a given script is included by using the Where Included form. If a script is included within another reposited script, you cannot delete it from the repository.

▶ To query for included scripts

1. From the File menu of the OneWorld Scripting Tool form, choose Repository and Where Included.
2. In the Where Included form, enter the title of a script and click OK.

   OneWorld Scripting Tool displays the titles of all scripts that include the script whose title you entered. You can check out or get a copy of any of these scripts.

Using a Command Line to Load a Script a Repository Script

You can load any repository script into AutoPilot from a command line. The script command line parameter is passed in with an .ATR extension. This extension is used to designate repository scripts.

The command line calls the AutoPilot executable and identifies the specific repository script that you want to load into AutoPilot. The .ATR extension operates as a flag to AutoPilot, notifying AutoPilot that it must got to the repository to retrieve and load the script designated in the command line.

```
C:\AutoPilot.exe MyRepositoryScript.ATR
```

Where MyRepositoryScript = the the title of the repository script to be loaded.

Understanding Script Reporting

The script repository forms a base of knowledge about scripts that test particular OneWorld applications and processes. The architecture of OneWorld Scripting Tool also contains a results repository, the OneWorld F98214 table. This repository forms a base of knowledge about the actual OneWorld Scripting Tool and OneWorld events that occur during script playback.

If you configure OneWorld Scripting Tool to capture and store playback events, it records each event, using internally placed code and code in OneWorld. At the completion of playback, OneWorld Scripting Tool sends the record of events, called the event stream, to the repository. You can view each event stream in the Test Results form.

This chapter discusses two key components of OneWorld Scripting Tool script reporting:

- ❑ Event stream
- ❑ Test Results form

Event Stream

The event stream provides a snapshot of what occurred during script playback. For example, you can see which OneWorld tables were opened, which business functions were called, which event rules were invoked, and how long it took to complete each event. You can also identify OneWorld error and warning messages that were generated. This information can help you to troubleshoot problems that might have occurred during playback.

Capturing an event stream is also necessary if you want to use J.D. Edwards OneWorld Analyzer Tool for further analysis of script playback or J.D. Edwards OneWorld Virtual User Tool to generate a virtual script that can be used to simulate multiple users on a single workstation.

Test Results Form

If you have configured script playback to capture, save and display results, the Test Results form appears at the completion of playback. The Test Results form displays playback data, such as the event stream, which is a chronological listing of each event that occurred during playback. You can filter the list for test time, type, or text. You can also view previous test results and you can select one or more to view details of the results.

For a discussion of setting up OneWorld Scripting Tool to capture script playback results see

The Test Results form contains details about script playback under each of the following:

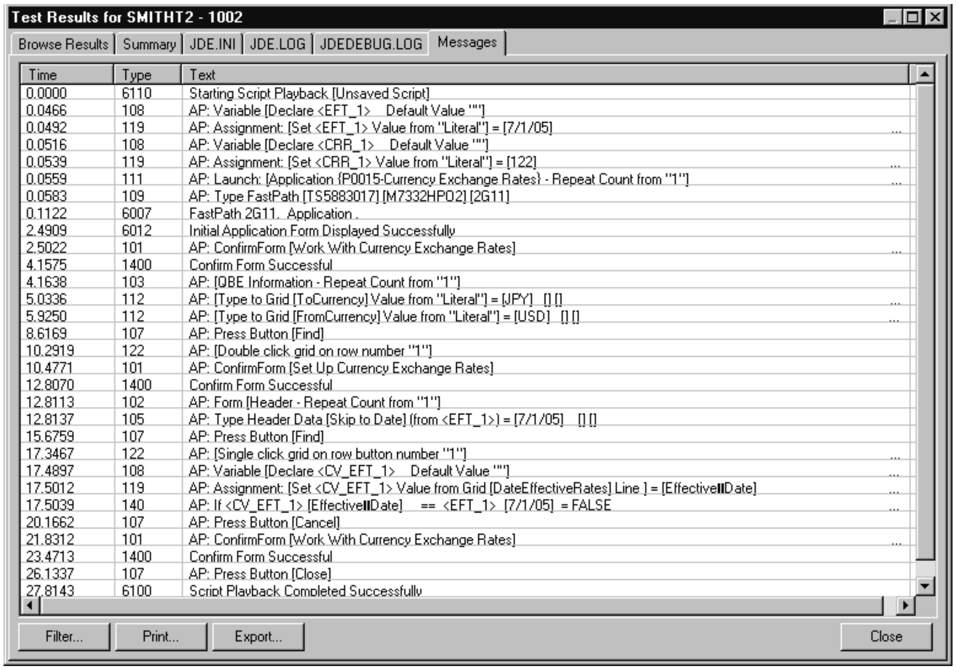
- The Messages tab
- The Summary tab
- The Browse Results tab
- The jde.ini tab
- The jde.log tab
- The jdedebug.log tab

See Also

- *Options for Configuring OneWorld Scripting Tool*
- *Results.*

The Messages Tab

Data displayed under the Messages tab of the Test Results form summarize the script that OneWorld Scripting Tool played back.



Time	Type	Text
0.0000	6110	Starting Script Playback [Unsaved Script]
0.0466	108	AP: Variable [Declare <EFT_1> Default Value ""]
0.0492	119	AP: Assignment: [Set <EFT_1> Value from "Literal"] = [7/1/05] ...
0.0516	108	AP: Variable [Declare <CRR_1> Default Value ""]
0.0539	119	AP: Assignment: [Set <CRR_1> Value from "Literal"] = [122] ...
0.0559	111	AP: Launch: [Application {P0015-Currency Exchange Rates} - Repeat Count from "1"] ...
0.0583	109	AP: Type FastPath [TS5883017] [M7332HPO2] [2G11]
0.1122	6007	FastPath 2G11. Application .
2.4909	6012	Initial Application Form Displayed Successfully
2.5022	101	AP: ConfirmForm [Work With Currency Exchange Rates] ...
4.1575	1400	Confirm Form Successful
4.1638	103	AP: [QBE Information - Repeat Count from "1"]
5.0336	112	AP: [Type to Grid [ToCurrency] Value from "Literal"] = [JPY] [] [] ...
5.9250	112	AP: [Type to Grid [FromCurrency] Value from "Literal"] = [USD] [] [] ...
8.6169	107	AP: Press Button [Find]
10.2919	122	AP: [Double click grid on row number "1"]
10.4771	101	AP: ConfirmForm [Set Up Currency Exchange Rates]
12.8070	1400	Confirm Form Successful
12.8113	102	AP: Form [Header - Repeat Count from "1"]
12.8137	105	AP: Type Header Data [Skip to Date] (from <EFT_1>) = [7/1/05] [] []
15.6759	107	AP: Press Button [Find]
17.3467	122	AP: [Single click grid on row button number "1"] ...
17.4897	108	AP: Variable [Declare <CV_EFT_1> Default Value ""] ...
17.5012	119	AP: Assignment: [Set <CV_EFT_1> Value from Grid [DateEffectiveRates] Line] = [Effective▮Date] ...
17.5039	140	AP: If <CV_EFT_1> [Effective▮Date] == <EFT_1> [7/1/05] = FALSE ...
20.1662	107	AP: Press Button [Cancel]
21.8312	101	AP: ConfirmForm [Work With Currency Exchange Rates] ...
23.4713	1400	Confirm Form Successful
26.1337	107	AP: Press Button [Close]
27.8143	6100	Script Playback Completed Successfully

You can review each context command and action command that you wrote in the script, as well as any error messages that OneWorld Scripting Tool may have generated. In addition, you can review any error messages that OneWorld may have generated during playback, such as those displayed in the status bar of OneWorld forms.

Under the Messages tab you can filter, print, and export test results. You can filter test results for:

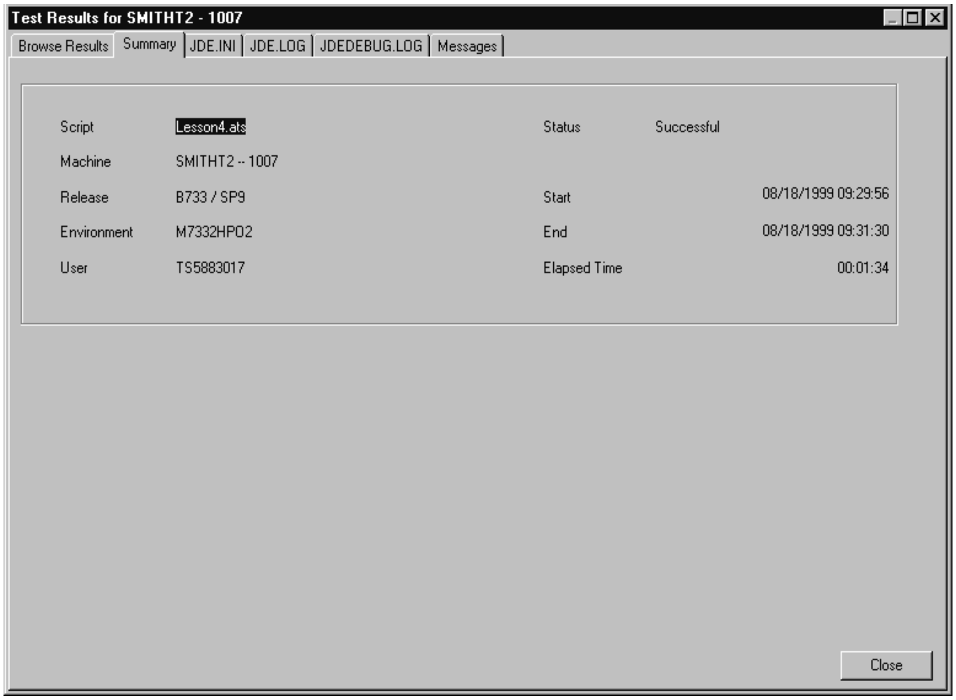
- A particular point during playback that an event occurred
- A particular kind of event, such as a message or an action in OneWorld Scripting Tool
- A text description of the event in the Test Results form

You can print your test results, provided you have set up your default printer to do so. Using the Export button, you can export your test results to an Excel spreadsheet.

The Summary Tab

Clicking the Summary tab displays the properties of each test you run:

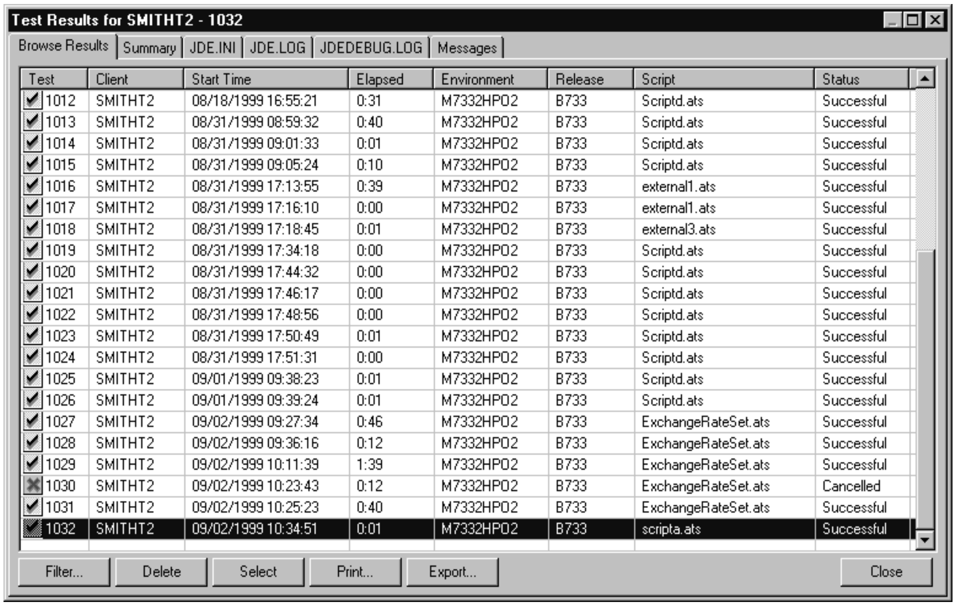
- Path
- Machine
- OneWorld release
- Environment
- User
- Start time
- End time
- Elapsed playback time
- Status of the playback



Script	Lesson4.ats	Status	Successful
Machine	SMITHT2 -- 1007		
Release	B733 / SP9	Start	08/18/1999 09:29:56
Environment	M7332HPO2	End	08/18/1999 09:31:30
User	TS5883017	Elapsed Time	00:01:34

The Browse Results Tab

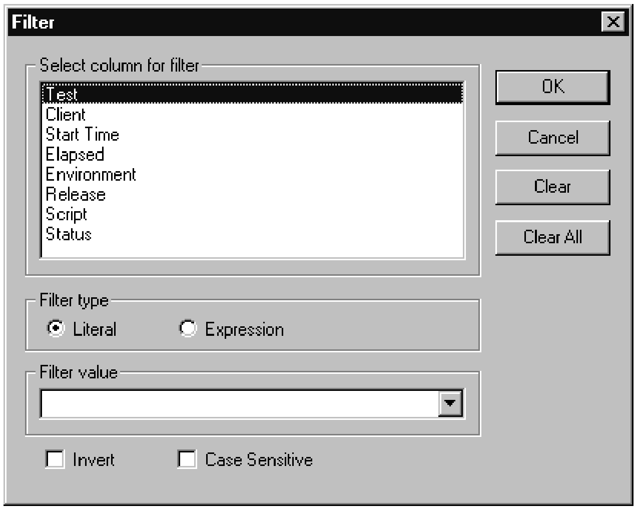
Under the Browse Results tab, you find summaries of all the tests whose results you have saved. You can also view the events in an individual test. A checkmark beside a test indicates that it was successful; an X indicates a test that failed or was cancelled.



Test	Client	Start Time	Elapsed	Environment	Release	Script	Status
✓ 1012	SMITHT2	08/18/1999 16:55:21	0:31	M7332HPO2	B733	Scriptd.ats	Successful
✓ 1013	SMITHT2	08/31/1999 08:59:32	0:40	M7332HPO2	B733	Scriptd.ats	Successful
✓ 1014	SMITHT2	08/31/1999 09:01:33	0:01	M7332HPO2	B733	Scriptd.ats	Successful
✓ 1015	SMITHT2	08/31/1999 09:05:24	0:10	M7332HPO2	B733	Scriptd.ats	Successful
✓ 1016	SMITHT2	08/31/1999 17:13:55	0:39	M7332HPO2	B733	external1.ats	Successful
✓ 1017	SMITHT2	08/31/1999 17:16:10	0:00	M7332HPO2	B733	external1.ats	Successful
✓ 1018	SMITHT2	08/31/1999 17:18:45	0:01	M7332HPO2	B733	external3.ats	Successful
✓ 1019	SMITHT2	08/31/1999 17:34:18	0:00	M7332HPO2	B733	Scriptd.ats	Successful
✓ 1020	SMITHT2	08/31/1999 17:44:32	0:00	M7332HPO2	B733	Scriptd.ats	Successful
✓ 1021	SMITHT2	08/31/1999 17:46:17	0:00	M7332HPO2	B733	Scriptd.ats	Successful
✓ 1022	SMITHT2	08/31/1999 17:48:56	0:00	M7332HPO2	B733	Scriptd.ats	Successful
✓ 1023	SMITHT2	08/31/1999 17:50:49	0:01	M7332HPO2	B733	Scriptd.ats	Successful
✓ 1024	SMITHT2	08/31/1999 17:51:31	0:00	M7332HPO2	B733	Scriptd.ats	Successful
✓ 1025	SMITHT2	09/01/1999 09:38:23	0:01	M7332HPO2	B733	Scriptd.ats	Successful
✓ 1026	SMITHT2	09/01/1999 09:39:24	0:01	M7332HPO2	B733	Scriptd.ats	Successful
✓ 1027	SMITHT2	09/02/1999 09:27:34	0:46	M7332HPO2	B733	ExchangeRateSet.ats	Successful
✓ 1028	SMITHT2	09/02/1999 09:36:16	0:12	M7332HPO2	B733	ExchangeRateSet.ats	Successful
✓ 1029	SMITHT2	09/02/1999 10:11:39	1:39	M7332HPO2	B733	ExchangeRateSet.ats	Successful
✗ 1030	SMITHT2	09/02/1999 10:23:43	0:12	M7332HPO2	B733	ExchangeRateSet.ats	Cancelled
✓ 1031	SMITHT2	09/02/1999 10:25:23	0:40	M7332HPO2	B733	ExchangeRateSet.ats	Successful
✓ 1032	SMITHT2	09/02/1999 10:34:51	0:01	M7332HPO2	B733	scripta.ats	Successful

The Test Results form for saved tests also permits you to print results and to export them to an Excel spreadsheet, using buttons at the bottom of the form.

The filter button allows you to filter the saved test results, by using the columns in the form as criteria. You use the Filter form to choose a filter criterion.


Filter

Select column for filter

Test
Client
Start Time
Elapsed
Environment
Release
Script
Status

Filter type

● Literal ○ Expression

Filter value

□ Invert □ Case Sensitive

The jde.ini Tab

The jde.ini tab,allows you to view the initialized settings for OneWorld that existed before OneWorld Scripting Tool ran the script playback. OneWorld Scripting Tool captures the file from C:\Winnt40\jde.ini, then displays its contents under the tab. You can troubleshoot the file to see, for example, if paths in the jde.ini setting point to the correct database or drive. You could also use data under the jde.ini tab to duplicate the results of one test in another.

The jde.log Tab

OneWorld Scripting Tool captures the jde.log file from C:\jde.log after script playback and displays it under the jde.log tab of the Test Results form. You can view the contents of the file in order to track error messages that might have occurred during OneWorld processing.

The jdedebug.log Tab

The jdedebug.log tab displays the jdedebug.log file that OneWorld Scripting Tool captures from the C:\jdedebug.log file after it completes script playback. You can troubleshoot the file to determine, for example, when normal execution of the script stopped. You can also review the timing of all OneWorld processes that occurred during script playback.

Understanding OneWorld Scripting Tool Test Manager

OneWorld Scripting Tool Test Manager is an executable that allows you to test multiple OneWorld Scripting Tool scripts simultaneously, to gather test results for archiving, and to review test results quickly. You use Test Manager to create a playlist that contains scripts that you saved on your local drive, that you retrieved from the script repository, or a combination of both.

After you assemble a playlist, you run it. Test Manager launches OneWorld Scripting Tool, which in turn launches OneWorld. Test Manager runs the playlist to completion, closing OneWorld Scripting Tool each time a script completes, and opening OneWorld Scripting Tool without delay when the next script emerges in the queue. Test Manager displays a test status message of failure, success, or incomplete for each script that runs.

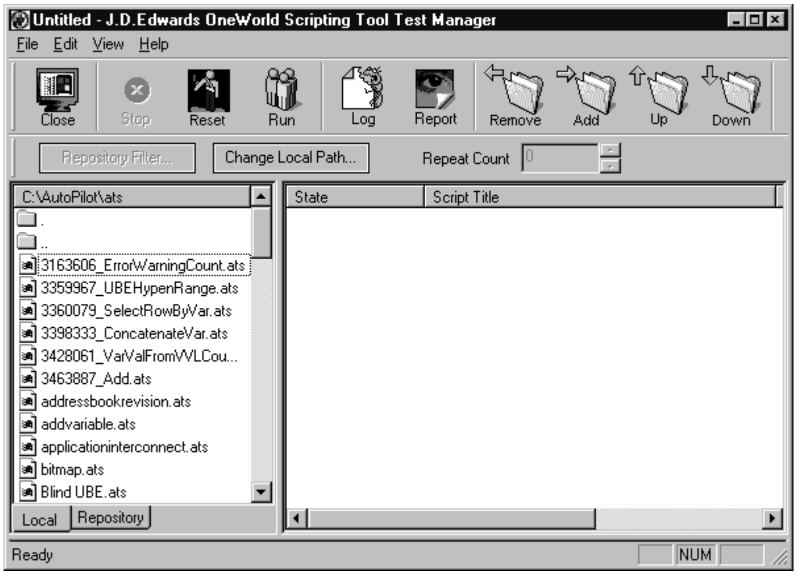
You can view the results of each script playback. These results include messages that help you to analyze the cause of any script failure that might have occurred. Finally, you can save the playlists you create, edit them, and replay them.

This chapter discusses the following components of the OneWorld Scripting Tool Test Manager user interface:

- ❑ Script display pane
- ❑ Script storage pane
- ❑ Test results pane
- ❑ OneWorld Scripting Tool Test Manager toolbar

Script Display Pane

The script display pane in the OneWorld Scripting Tool Test Manager is the area where you choose scripts to assemble your playlist. The pane contains two tabs, Local and Repository. When you choose the Local tab, Test Manager displays all the scripts that you have stored on your local drive.

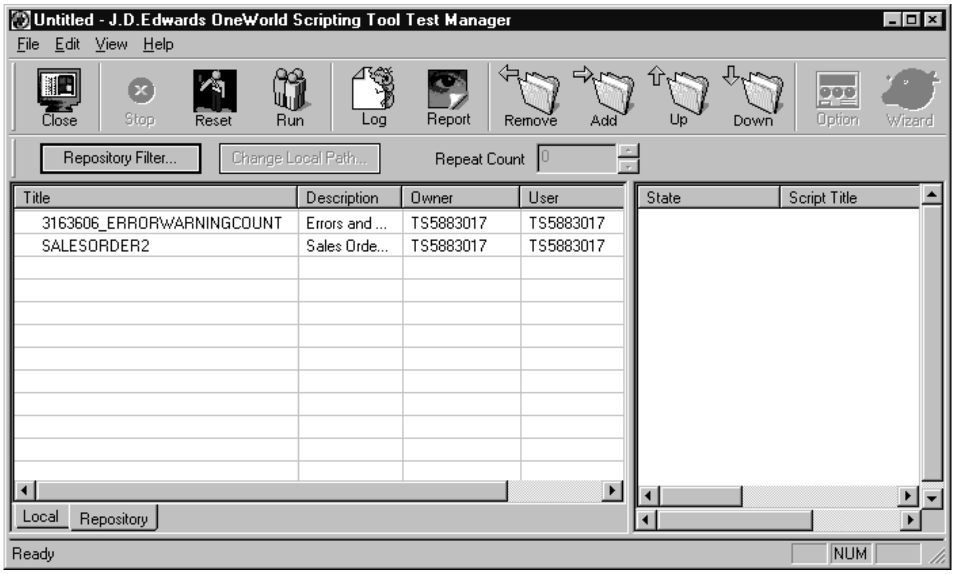


C:\AutoPilot\ats
- .
- ..
- 3163606_ErrorWarningCount.ats
- 3359967_UBEHypenRange.ats
- 3360079_SelectRowByVar.ats
- 3398333_ConcatenateVar.ats
- 3428061_VarValFromVVLCou...
- 3463887_Add.ats
- addressbookrevision.ats
- addvariable.ats
- applicationinterconnect.ats
- bitmap.ats
- Blind UBE.ats

State	Script Title

When you choose the Repository tab and Repository Filter button, Test Manager launches the Select Script form, which you can use to enter search criteria for scripts that have been checked into the repository. For details on using the script repository, see *Understanding the Script Repository* and *Working with the Script Repository* in this section.

Test Manager populates the script display pane with the names of the scripts that match the property criteria you enter in the Select Script form, and you can add these scripts to the playlist in the script storage pane.



Title	Description	Owner	User
3163606_ERRORWARNINGCOUNT	Errors and ...	TS5883017	TS5883017
SALESORDER2	Sales Orde...	TS5883017	TS5883017

State	Script Title

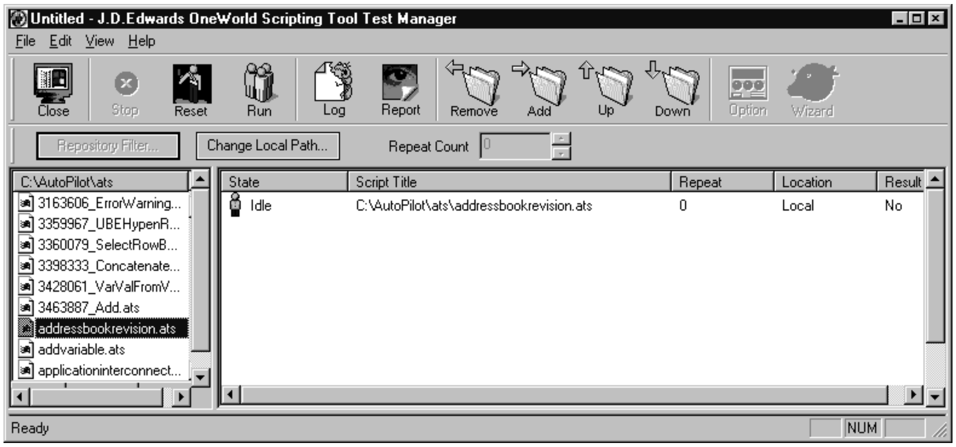
Note: Test Manager creates copies of the repository scripts. It does not check out scripts from the repository. Therefore, adding a reposited script to the Test Manager playlist does not prevent other users from checking out the script from the repository and making changes to it.

See Also

- *Property Pages for Scripts*
- *Assigning Properties to a Script*

Script Storage Pane

You add scripts from the script display pane to the script storage pane to create a playlist. When you initially add scripts to the script storage pane, Test Manager displays the state of the script as Idle, meaning that you have added it to the playlist, but have not yet run it.



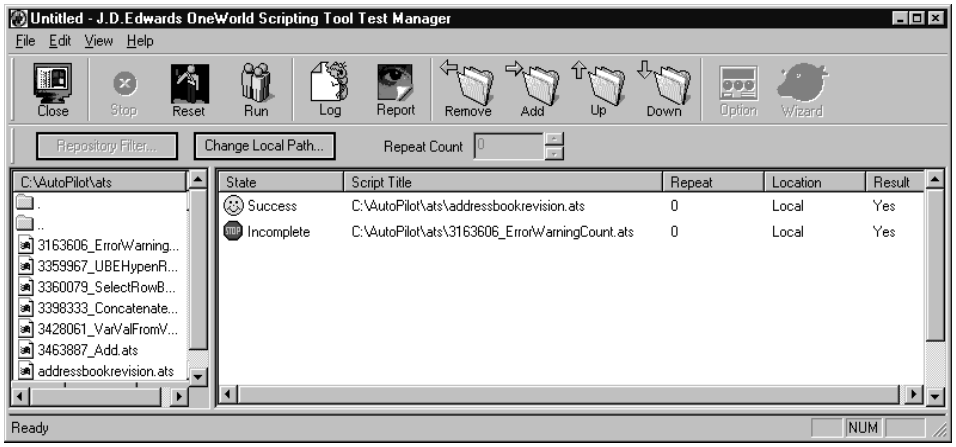
Repeat Count: 0

C:\AutoPilot\ats
3163606_ErrorWarning...
3359967_UBEHypenR...
3360079_SelectRowB...
3398333_Concatenate...
3428061_VarValFromV...
3463887_Add.ats
addressbookrevision.ats
addvariable.ats
applicationinterconnect...

State	Script Title	Repeat	Location	Result
Idle	C:\AutoPilot\ats\addressbookrevision.ats	0	Local	No

You can remove one or more scripts from the script storage pane. If you decide to remove a script from the playlist, Test Manager asks you to confirm that you want to remove the script from the playlist.

Once you have assembled the playlist that you want, you run it by clicking the Run button in the toolbar. Test Manager launches OneWorld Scripting Tool and runs the scripts in the order that you queued them in the script storage pane. After the script runs, Test Manager displays the state of each according to the results of the test: success, failure, cancellation, or incomplete.



C:\AutoPilot\ats

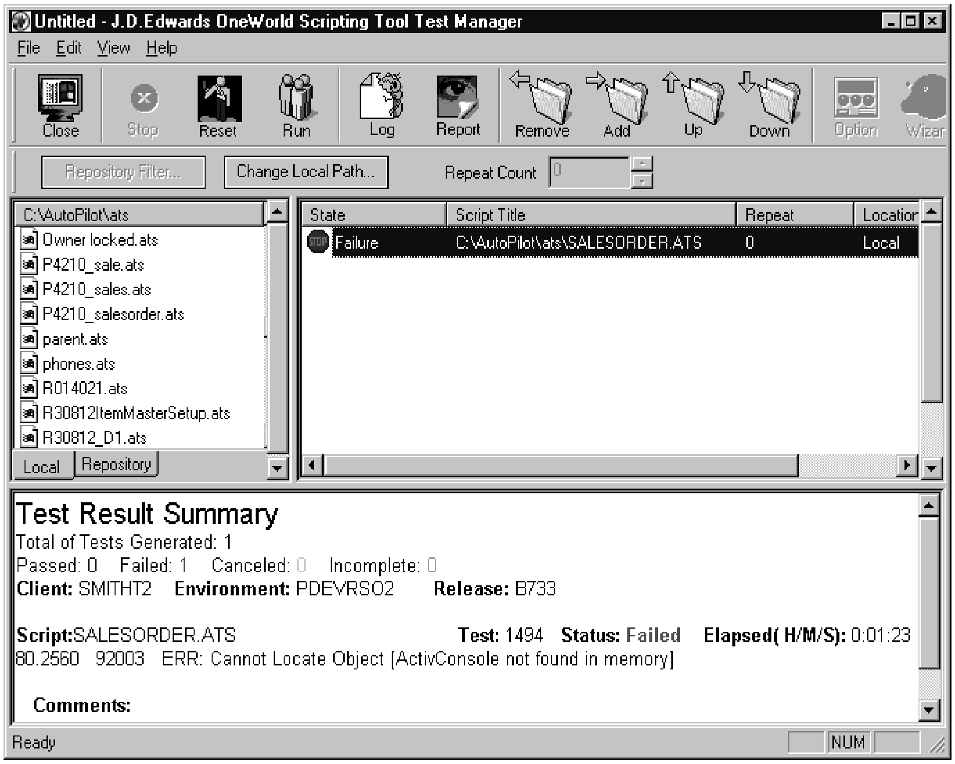
- .
- ..
- 3163606_ErrorWarning...
- 3359967_UBEHypenR...
- 3360079_SelectRowB...
- 3398333_Concatenate...
- 3428061_VarValFromV...
- 3463887_Add.ats
- addressbookrevision.ats

State	Script Title	Repeat	Location	Result
Success	C:\AutoPilot\ats\addressbookrevision.ats	0	Local	Yes
Incomplete	C:\AutoPilot\ats\3163606_ErrorWarningCount.ats	0	Local	Yes

Test Results Pane

After you have assembled and run a playlist, Test Manager summarizes the results in the test results pane. You can view the summary by clicking the Report button in the toolbar. The Test Result Summary offers at a glance the following information about the test:

- Total number of tests generated
- Status breakdown: the number of scripts that failed, succeeded, were cancelled, or did not complete
- Name of the client machine
- OneWorld environment against which the test was run
- OneWorld release against which the test was run
- Name of the script
- Number of the test
- Status of each script run
- Time elapsed for each script run
- Comments you added to the script and designated for logging in Test Manager



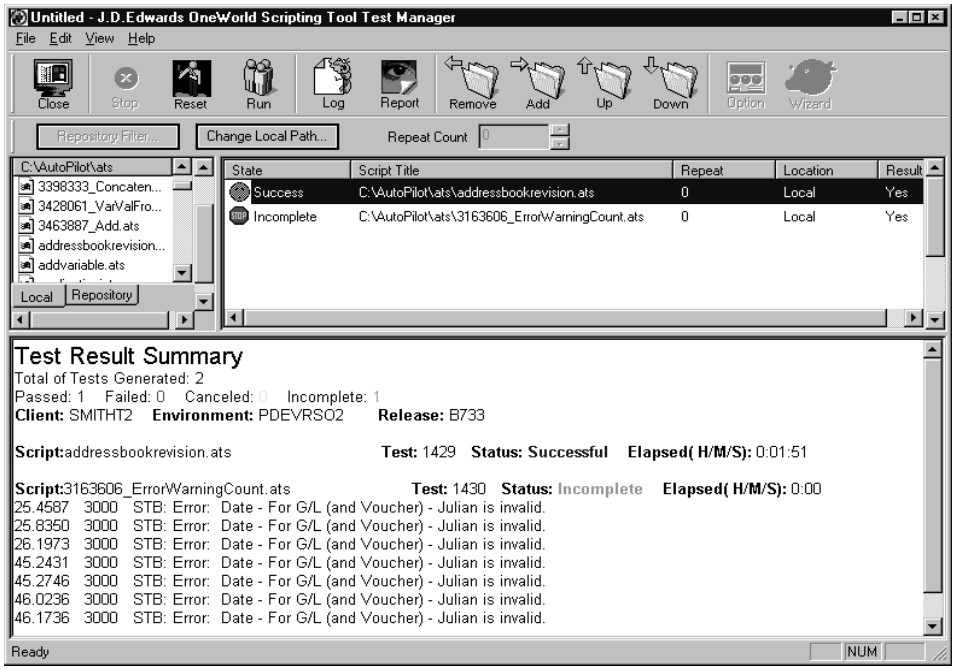
State	Script Title	Repeat	Location
Failure	C:\AutoPilot\ats\SALESORDER.ATS	0	Local

Test Result Summary
Total of Tests Generated: 1
Passed: 0 Failed: 1 Canceled: 0 Incomplete: 0
Client: SMITHT2 **Environment:** PDEVRSO2 **Release:** B733

Script:SALESORDER.ATS **Test:** 1494 **Status:** Failed **Elapsed(H/M/S):** 0:01:23
80.2560 92003 ERR: Cannot Locate Object [ActivConsole not found in memory]

Comments:

If a script fails, Test Manager displays message types in the test results pane, along with the time of the message. These message types provide information on the reason for the script's failure as well as information about warning messages that might have occurred during playback.



State	Script Title	Repeat	Location	Result
Success	C:\AutoPilot\ats\addressbookrevision.ats	0	Local	Yes
Incomplete	C:\AutoPilot\ats\3163606_ErrorWarningCount.ats	0	Local	Yes

Test Result Summary
Total of Tests Generated: 2
Passed: 1 Failed: 0 Canceled: 0 Incomplete: 1
Client: SMITHT2 **Environment:** PDEVRSO2 **Release:** B733

Script:addressbookrevision.ats **Test:** 1429 **Status: Successful** **Elapsed(H/M/S):** 0:01:51

Script:3163606_ErrorWarningCount.ats **Test:** 1430 **Status:** Incomplete **Elapsed(H/M/S):** 0:00

```
25.4587   3000   STB: Error:  Date - For G/L (and Voucher) - Julian is invalid.
25.8350   3000   STB: Error:  Date - For G/L (and Voucher) - Julian is invalid.
26.1973   3000   STB: Error:  Date - For G/L (and Voucher) - Julian is invalid.
45.2431   3000   STB: Error:  Date - For G/L (and Voucher) - Julian is invalid.
45.2746   3000   STB: Error:  Date - For G/L (and Voucher) - Julian is invalid.
46.0236   3000   STB: Error:  Date - For G/L (and Voucher) - Julian is invalid.
46.1736   3000   STB: Error:  Date - For G/L (and Voucher) - Julian is invalid.
```

In addition, Test Manager provides information on OneWorld warning messages. The following table lists the message types that can appear in the test results pane and summarizes their meaning:

Message Type Displayed in Test Results Pane	Meaning
110	Failure status with text ###FAILURE&&&.
138	Warning status. Each message type 138 includes the path for a screen capture that is included in the OneWorld Scripting Tool script.
2607	Failure status: no data returned.
2608	Failure status: Unexpected records found during validation.
2609	Failure status: database validation failed.
3000	Failure status: OneWorld status bar message that contains warnings or error text. Message text that includes “STB: Error...” indicates that the script failed. Warning messages do not indicate that the script has failed. However, Test Manager summarizes all warning messages as an additional aid for the tester.
6016	Failure status: variable not found.
6301	Warning status: OneWorld Scripting Tool failed to set processing option text, which might cause a failure later in the script.

OneWorld Scripting Tool Test Manager Toolbar

The Test Manager toolbar allows you to control a test session and to view its results. The Test Manager toolbar contains the following buttons:

- Close allows you to close out of a script-testing session, after which you can save the playlist.
- Stop allows you to stop a script playback session, after which you can save the playlist.
- Reset allows you to rerun a session, which could overwrite previous test results, if you save the results of the rerun session.
- Run allows you to initiate a test session, which launches OneWorld Scripting Tool
- Log displays the Browse Results form, which contains detailed summaries of each test you have run and saved in Test Manager. If you select a test in the Browse Results form and double click or click the Select button, the Messages form appears, which presents details about each OneWorld Scripting Tool and OneWorld event that occurred during playback.
- Report: Populates the test results pane, after you have run a script playlist, with summary information about the playback.

- Remove: Allows you to remove a script from the playlist in the script storage pane.
- Add: Allows you to add a script from the script display pane to the script storage pane.
- Up: Allows you to move upward in the playlist from one script to another.
- Down: Allows you to move downward in the playlist from one script to another.

Managing Script Testing

You use OneWorld Scripting Tool Test Manager to create a playlist from scripts that reside on your local drive or in the script repository. Test Manager allows you to run multiple times without intervention a playlist that can contain a mixture of local and reposited scripts. You can save a playlist, or you can reset it and play it again from the top. You can view the summarized results of each playback in the test results pane, collected playback results or the events of an individual test.

This chapter discusses the steps required to manage script testing:

❑ Creating a playlist

❑ Saving a playlist

❑ Running a test

❑ Viewing test results

❑ Resetting a test

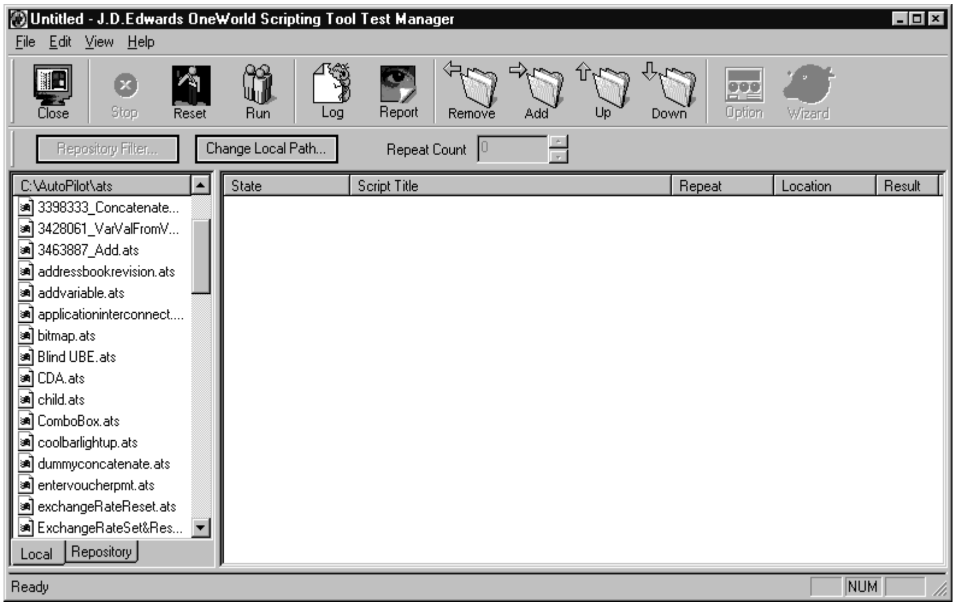
Creating a Playlist

You begin work in Test Manager by creating a playlist. You retrieve the scripts for your playlist from your local drive, from the script repository, or from both sources. The Add button on the toolbar enables you to move scripts from the script display pane to the script storage pane, where the playlist resides.

▶ To create a playlist

1. On your desktop or in the directory where you store OneWorld Scripting Tool Test Manager, click the Test Manager executable.

   The Test Manager splash screen appears, followed by the Test Manager form.



2. In the script display pane of the Test Manager form, click either the Local or the Repository tab.

3. If you click the Local tab, choose a local script from the script display pane and click the Add button in the toolbar.

   Test Manager adds the test you chose to the script storage pane.

   Note: You can choose more than one test by choosing a script in the script display pane, holding down either the Control or the Shift key, and choosing another script or scripts.

4. If you click the Repository tab, click the Repository filter button.

   Test Manager displays the Select Script form.

5. In the Select Script form, choose any script criteria you desire to narrow the number of scripts you want to copy from the repository and click OK.

   Test Manager displays in the script display pane any repository scripts that match your search criteria.

6. Choose one or more repository scripts from the script display pane and click the Add button in the toolbar.

7. Continue adding local and/or repository scripts until you have created the playlist that you desire.

8. To remove a script from the playlist, select it in the script storage pane and click the Remove button in the toolbar.

Saving a Playlist

After you have created a playlist, you might wish to save it, although you can run the test before you save it. Remember that if you do not save the playlist, Test Manager prompts you to do so when you attempt to exit the form.

▶ To save a playlist

1. From the file menu of the Test Manager form, choose Save or Save As.
2. Assign the playlist a name and save it to the drive, directory, and file that you desire and click Save.

   Note: Test Manager assigns to all playlists the default extension of .apl.

Running a Test

After you have created a playlist, you can run the test. Test Manager launches OneWorld Scripting Tool, and then runs each script in the queue according to the test order that you set up in the script storage pane.

Test Manager launches OneWorld Scripting Tool, minimizes the OneWorld Scripting Tool form, and then begins running the first script in the queue. As each test completes, Test Manager displays its result in the script storage pane. When Test Manager completes running a script, it closes OneWorld Scripting Tool, and then relaunches it with the beginning of the next script in the queue.

As each script completes running, Test Manager displays the result: success, failure, canceled, or incomplete.

▶ To run a test

1. In the File menu of the Test Manager form, choose Open.
2. Open the drive, directory, and file in which you store your playlists, choose one or more playlists and click Open.
3. In the toolbar of the Test Manager form, click Run.

   Note: During playback, OneWorld remains open, unless a script contains an Exit OneWorld command, in which case OneWorld closes. In this case, with the beginning of the next script in the queue, Test Manager launches OneWorld Scripting Tool, which in turn launches OneWorld.

4. To stop the test, click the Stop button in the toolbar.

Viewing Test Results

After Test Manager completes the playlist, you can view the playback results in one of two ways. Clicking the Report button enables you to view in the test results pane a summary of the results from each test in the just-completed playlist. To view summaries of the tests from all the playlists you have run and saved, you click the Log button.

▶ To view test results

1. In the Test Manager form, open a saved playlist.
2. After Test Manager has completed the playlist, click the Report button in the toolbar.

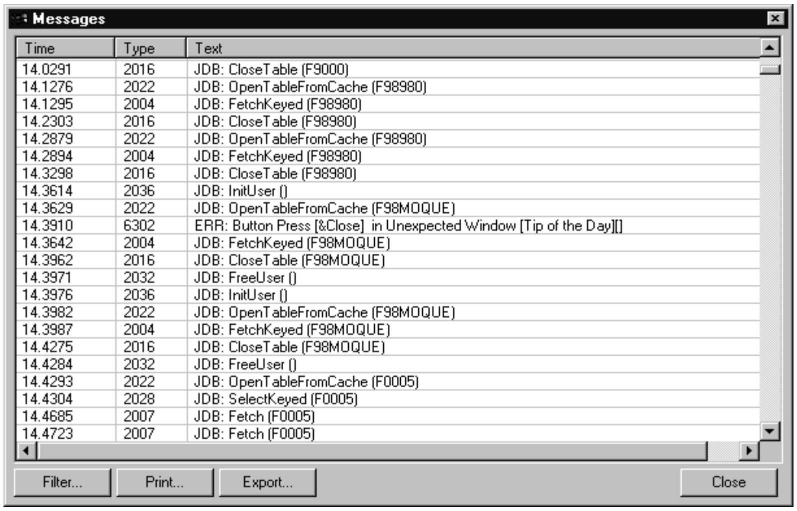
   Test Manager populates the test results pane with a summary of the results for each script in the playlist.

3. To view summaries of all scripts that you have played back, click the Log button in the toolbar.

   Test Manager displays the Browse Results form, which contains summary information of the results of all played-back scripts.

4. To view in detail all the events for the playback of an individual script, select the script in the Browse Results form, and then double click or click the Select button.

   The Messages form displays all the script playback events in chronological order.



Time	Type	Text
14.0291	2016	JDB: CloseTable (F9000)
14.1276	2022	JDB: OpenTableFromCache (F98980)
14.1295	2004	JDB: FetchKeyed (F98980)
14.2303	2016	JDB: CloseTable (F98980)
14.2879	2022	JDB: OpenTableFromCache (F98980)
14.2894	2004	JDB: FetchKeyed (F98980)
14.3298	2016	JDB: CloseTable (F98980)
14.3614	2036	JDB: InitUser ()
14.3629	2022	JDB: OpenTableFromCache (F98MOQUE)
14.3910	6302	ERR: Button Press [&Close] in Unexpected Window [Tip of the Day][]
14.3642	2004	JDB: FetchKeyed (F98MOQUE)
14.3962	2016	JDB: CloseTable (F98MOQUE)
14.3971	2032	JDB: FreeUser ()
14.3976	2036	JDB: InitUser ()
14.3982	2022	JDB: OpenTableFromCache (F98MOQUE)
14.3987	2004	JDB: FetchKeyed (F98MOQUE)
14.4275	2016	JDB: CloseTable (F98MOQUE)
14.4284	2032	JDB: FreeUser ()
14.4293	2022	JDB: OpenTableFromCache (F0005)
14.4304	2028	JDB: SelectKeyed (F0005)
14.4685	2007	JDB: Fetch (F0005)
14.4723	2007	JDB: Fetch (F0005)

Resetting a Test

Once you have assembled a playlist and run it, you can reset the test, which overwrites the previous results. Resetting might be appropriate if scripts in the original test fail and you make changes to correct the failure.

▶ To reset a test

1. In the Test Manager form, open a playlist that you have already run.
2. In the toolbar, click Reset.

   Test Manager displays a form advising you that resetting the test overwrites the existing results.

3. Click Yes.
4. In the Toolbar, click the Run button to rerun the test.

Glossary

Glossary

AAI. See automatic accounting instruction.

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.

activator. In the Solution Explorer, a parent task with sequentially-arranged child tasks that are automated with a director.

ActiveX. A computing technology, based on object linking and embedding, that enables Java applet-style functionality for Web browsers as well as other applications. (Java is limited to Web browsers at this time.) The ActiveX equivalent of a Java applet is an ActiveX control. These controls bring computational, communications, and data manipulation power to programs that can "contain" them. For example, certain Web browsers, Microsoft Office programs, and anything developed with Visual Basic or Visual C++.

advance. A change in the status of a project in the Object Management Workbench. When you advance a project, the status change might trigger other actions and conditions such as moving objects from one server to another or preventing check-out of project objects.

alphanumeric character. A combination of letters, numbers, and symbols used to represent data. Contrast with numeric character and special character.

API. See application programming interface.

APPL. See application.

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. In the computer industry, the same as an executable file. In OneWorld, an interactive or batch application is a DLL that contains programming for a set of related forms that can be run from a menu to perform a business task such as Accounts Payable and Sales Order Processing. Also known as system.

application developer. A programmer who develops OneWorld applications using the OneWorld toolset.

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

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

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 Accounts Payable, Accounts Receivable, Financial Reporting, 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.

batch header. The information that identifies and controls a batch of transactions or records.

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 computer 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 J.D. Edwards 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 applicationfor 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.

BDA. See Business View Design Aid.

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.

Boolean Logic Operand. In J.D. Edwards reporting programs, the parameter of the Relationship field. The Boolean logic operand instructs the system to compare certain records or parameters. Available options are:

EQ	Equal To.
LT	Less Than.
LE	Less Than or Equal To.
GT	Greater Than.
GE	Greater Than or Equal To.
NE	Not Equal To.
NL	Not Less Than.
NG	Not Greater Than.

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

BSFN. See business function.

BSTR. See binary string.

BSVW. See business view.

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. See named 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 ool for creating, modifying, copying, and printing business views. The tool uses a graphical user interface.

category code. In user defined codes, a temporary title for an undefined category. For example, if you are adding a code that designates 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.

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.

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 sub-directories 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/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.

CNC. See configurable network computing.

component. In the ActivEra Portal, an encapsulated object that appears inside a workspace. Portal components

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. 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 the system uses to standardize information processing by associated programs. Some examples of constants are: validating bills of material online and including fixed labor overhead in costing.

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.

core. The central and foundation systems of J.D. Edwards software, including General Accounting, Accounts Payable, Accounts Receivable, Address Book, Financial Reporting, Financial Modeling and Allocations, and Back Office.

CRP. Conference Room Pilot.

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.

data dictionary. The OneWorld method for storing and managing data item definitions and specifications. J.D. Edwards has an active data dictionary, which means it is accessed at runtime.

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 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 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 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.

data warehousing. Essentially, data warehousing involves off-loading operational data sources to target databases that will be used exclusively for decision support (reports and queries). There are a range of decision support environments, including duplicated database, enhanced analysis databases, and enterprise data warehouses.

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

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

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

DCE. See distributed computing environment.

DD. See data dictionary.

default. A code, number, or parameter value that is assumed when none is specified.

detail. The specific pieces of information and data that make up a record or transaction. Contrast with summary.

detail area. A control that is found in OneWorld applications and functions similarly to a spreadsheet grid for viewing, adding, or updating many rows of data at one time.

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. An interactive utility that guides a user through the steps of a process to complete a task.

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.

DLL. See dynamic link library.

DS. See data structure.

DSTR. See data structure.

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.

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.

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 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.

ER. See event rule.

ERP. See enterprise resource planning.

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 Named event rules.

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

exit. 1) To interrupt or leave a computer program by pressing a specific key or a sequence of keys. 2) An option or function key displayed on a form that allows you to access another form.

facility. 1) A separate 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. 2) In Home Builder and ECS, a facility is a collection of computer language statements or programs that provide a specialized function throughout a system or throughout all integrated systems. For example, DREAM Writer and FASTR are facilities.

FDA. See Form Design Aid.

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. An element of OneWorld's graphical user interface that contains controls by which a user can interact with an application. Forms allow the user to input, select, and view information. A OneWorld application might contain multiple forms. In Microsoft Windows terminology, a form is known as a dialog box.

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.

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. See detail area.

GUI. See graphical user interface.

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

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

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

hidden selections. Menu selections you cannot see until you enter HS in a menu's Selection field. Although you cannot see these selections, they are available from any menu. They include such items as Display Submitted Jobs (33), Display User Job Queue (42), and Display User Print Queue (43). The Hidden Selections window displays three categories of selections: user tools, operator tools, and programmer tools.

host. In the centralized computer model, a large timesharing computer system that terminals communicate with and rely on for processing. In 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.

HTML. See hypertext markup language.

hypertext markup language. A markup language used to specify the logical structure of a document rather than the physical layout. Specifying logical structure makes any HTML document platform independent. You can view an HTML document on any desktop capable of supporting a browser. HTML can include active links to other HTML documents anywhere on the Internet or on intranet sites.

index. Represents both an ordering of values and a uniqueness of values that provide efficient access to data in rows of a table. An index is made up of one or more columns in the table.

inheritance. The ability of a class to recieve all or parts of the data and procedure definitions from a parent class. Inheritance enhances developement through the reuse of classes and their related code.

install system code. See system code.

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.

interactive processing. Processing actions that occur in response to commands 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. See also online. Contrast with batch processing.

interface. A link between two or more computer systems that allows these systems to send information to and receive information from one another.

Internet. The worldwide constellation of servers, applications, and information available to a desktop client through a phone line or other type of remote access.

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 connection-less 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.

ISO 9000. A series of standards established by the International Organization for Standardization, designed as a measure of product and service quality.

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 Database Connectivity (JDBC). The standard way to access Java databases, set by Sun Microsystems. This standard allows you to use any JDBC driver database.

JavaScript. A scripting language related to Java. Unlike Java, however, JavaScript is not an object-oriented language and it is not compiled.

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.

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 datasource 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.

JDENET. J.D. Edwards proprietary middleware software. JDENET is a messaging software package.

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 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.

KEY. A column or combination of columns that identify one or more records in a database table.

leading zeros. A series of zeros that certain facilities in J.D. Edwards systems place in front of a value you enter. This normally occurs when you enter a value that is smaller than the specified length of the field. For example, if you enter 4567 in a field that accommodates eight numbers, the facility places four zeros in front of the four numbers you enter. The result appears as: 00004567.

level of detail. 1) The degree of difficulty of a menu in J.D. Edwards software. The levels of detail for menus are as follows:

A	Major Product Directories.
B	Product Groups.
1	Basic Operations.
2	Intermediate Operations.
3	Advanced Operations.
4	Computer Operations.
5	Programmers.
6	Advanced Programmers Also known as menu levels.

2) The degree to which account information in the General Accounting system is summarized. The highest level of detail is 1 (least detailed) and the lowest level of detail is 9 (most detailed).

MAPI. See Messaging Application Programming Interface.

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.

menu. A menu that displays numbered selections. Each of these selections represents a program or another menu. To access a selection from a menu, type the selection number and then press Enter.

menu levels. See level of detail.

menu masking. A security feature of J.D. Edwards systems that lets you prevent individual users from accessing specified menus or menu selections. The system does not display the menus or menu selections to unauthorized users.

Messaging Application Programming Interface (MAPI). An architecture that defines the components of a messaging system and how they behave. It also defines the interface between the messaging system and the components.

middleware. A general term that covers all the distributed software needed to support interactions between clients and servers. Think of it as the software that's in the middle of the client/server system or the "glue" that lets the client obtain a service from a server.

modal. A restrictive or limiting interaction created by a given condition of operation. Modal often describes a secondary window that restricts a user's interaction with other windows. A secondary window can be modal with respect to it's primary window or to the entire system. A modal dialog box must be closed by the user before the application continues.

mode. In reference to forms in OneWorld, mode has two meanings:

- An operational qualifier that governs how the form interacts with tables and business views. OneWorld form modes are: add, copy, and update.
- An arbitrary setting that aids in organizing form generation for different environments. For example, you might set forms generated for a Windows environment to mode 1 and forms generated for a Web environment to mode 2.

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

NER. See named event rule.

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 you use to control the automatic numbering of such items as new G/L accounts, vouchers, and addresses. It lets you specify a numbering system and provides a method to increment numbers to reduce transposition and typing errors.

non–object librarian object. An object that is not managed by the object librarian.

numeric character. 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. You access these objects with the Object Management Workbench.

object librarian object. An object managed by the object librarian.

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 management workbench (OMW). An application that provides check-out and check-in capabilities for developers, and aids in the creation, modification, and use of OneWorld Objects. The OMW supports multiple environments (such as production and development).

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.

OCM. See object configuration manager.

ODBC. See open database connectivity.

OLE. See object linking and embedding.

OMW. Object Management Workbench.

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. 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:

- JFC (0.5.1).
- Generator Package with Generator.Java and JDECOM.dll.
- R2 with interpretive and apllication controls/form.

online. Computer functions over which the system has continuous control. Users are online with the system when working with J.D. Edwards system provided forms.

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.

operand. See Boolean Logic Operand.

output. Information that the computer transfers from internal storage to an external device, such as a printer or a computer form.

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 install 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 it's set of replicated objects. This is usually \\deployment server\release\path_code\package\ package name. The sub-directories 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.

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.

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.

portability. Allows the same application to run on different operating systems and hardware platforms.

portal. A configurable Web object that provides information and links to the Web. Portals can be used as home pages and are typically used in conjunction with a Web browser.

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 list.

processing option. A feature of the J.D. Edwards reporting system that allows you to supply parameters to direct the functions of a program. For example, processing options allow you to specify defaults for certain form displays, control the format in which information prints on reports, change how a form displays information, and enter beginning dates.

program temporary fix (PTF). A representation of changes to J.D. Edwards software that your organization receives on magnetic tapes or diskettes.

project. An Object Management Workbench object used to organize objects in development.

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.

QBE. See query by example.

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.

redundancy. Storing exact copies of data in multiple databases.

regenerable. Source code for OneWorld business functions can be regenerated from specifications (business function names). Regeneration occurs whenever an application is recompiled, eitherfor 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/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 the directory where these objects are located.

run. To cause the computer system to perform a routine, process a batch of transactions, or carry out computer program instructions.

SAR. See software action request.

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.

search/select. A type of form used to search for a value and return it to the calling field.

selection. Found on J.D. Edwards menus, selections represent functions that you can access from a menu. To make a selection, type the associated number in the Selection field and press Enter.

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.

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-side what applets are to the client-side.

software. The operating system and application programs that tell the computer how and what tasks to perform.

software action request (SAR). An entry in the AS/400 database used for requesting modifications to J.D. Edwards software.

special character. A symbol used to represent data. Some examples are *, &, #, and /. Contrast with alphanumeric character and numeric character.

specifications. A complete description of a OneWorld object. Each object has its own specification, or name, which is used to build applications.

Specs. See specifications.

spool. The function by which the system stores generated output to await printing and processing.

spooled table. A holding file for output data waiting to be printed or input data waiting to be processed.

SQL. See structured query language.

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.

structured query language (SQL). A fourth generation language used as an industry standard for relational database access. It can be used to create databases and to retrieve, add, modify, or deleta data from databases. SQL is not a complete programming language because it does not contain control flow logic.

subfile. See detail.

submit. See run.

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.

subsystem job. Within OneWorld, subsystem jobs are batch processes that continually run independently of, but asynchronously with, OneWorld applications.

summary. The presentation of data or information in a cumulative or totaled manner in which most of the details have been removed. Many of the J.D. Edwards systems offer forms and reports that are summaries of the information stored in certain tables. Contrast with detail.

system. See application.

System Code. System codes are a numerical representation of J.D. Edwards and customer systems. For example, 01 is the system code for Address Book. System codes 55 through 59 are reserved for customer development by customers. Use system codes to categorize within OneWorld. For example, when establishing user defined codes (UDCs), you must include the system code the best categorizes it. When naming objects such as applications, tables, and menus, the second and third characters in the object's name is the system code for that object. For example, G04 is the main menu for Acounts Payable, and 04 is its system code.

system function. A program module, provided by OneWorld, available to applications and reports for further processing.

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 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.

TAM. Table Access Management.

TBLE. See table.

TC. Table conversion.

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.

TDA. See table design aid.

TER. See table event rules.

Terminal Identification. The workstation ID number.Terminal number of a specific terminal or IBM user ID of a particular person for whom this is a valid profile.Header Field: Use the Skip to Terminal/User ID field in the upper portion of the form as an inquiry field in which you can enter the number of a terminal or the IBM user ID of a specific person whose profile you want the system to display at the top of the list. When you first access this form, the system automatically enters the user ID of the person signed on to the system.Detail Field: The Terminal/User ID field in the lower portion of the form contains the user ID of the person whose profile appears on the same line.A code identifying the user or terminal for which you accessed this window.

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.

token. A referent to an object used to determine ownership of that object and to prevent non-owners from checking the object out in Object Management Workbench. An object holds its own token until the object is checked out, at which time the object passes its token to the project in which the object is placed.

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.

UBE. Universal batch engine.

UDC Edit Control. Use a User-Defined Code (UDC) Edit Control for a field that accepts only specific values defined in a UDC table. Associate a UDC edit control with a database item or dictionary item. The visual assist Flashlight automatically appears adjacent to the UDC edit control field. When you click on the visual assist Flashlight, the attached search and select form displays valid values for the field. To create a UDC Edit Control, you must:

- Associate the data item with a specific UDC table in the Data Dictionary.
- Create a search and select form for displaying valid values from the UDC table.

uniform resource identifier (URI). A character string that references an internet object by name or location. A URL is a type of URI.

uniform resource locator (URL). Names the address (location) of a document on the Internet or an intranet. A URL includes the document's protocol and server name. The path to the document might be included as well. The following is an example of a URL: http://www.jdedwards.com. This is J.D. Edwards Internet address.

URI. See uniform resource identifier.

URL. See uniform resource locator.

user defined code (type). The identifier for a table of codes with a meaning you define for the system, such as ST for the Search Type codes table in Address Book. J.D. Edwards systems provide a number of these tables and allow you to create and define tables of your own. User defined codes were formerly known as descriptive titles.

user defined codes (UDC). Codes within software that users can define, relate to code descriptions, and assign valid values. Sometimes user defined codes are referred to as a generic code table. Examples of such codes are unit-of-measure codes, state names, and employee type codes.

UTB. Universal Table Browser.

valid codes. The allowed codes, amounts, or types of data that you can enter in a field. The system verifies the information you enter against the list of valid codes.

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 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.

WF. See workflow.

window. See form.

workflow. According to the Workflow Management Coalition, worlflow means "the automation of a business process, in whole or part, during which documents, information, or tasks are passed from one participant to another for action, according to a set of procedural rules.".

workgroup server. A remote database server usually containing subsets of data replicated from a master database server. This server does not performance an application or batch processing. It may or may not have OneWorld running (in order to replicate data).

workspace. In the ActivEra Portal, the main section of the Portal. A user might have access to several workspaces, each one configured differently and containing its own components.

worldwide web. A part of the Internet that can transmit text, graphics, audio, and video. The

World Wide 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