

The Complete E-Commerce Solution for Everyone!

WebShop™ Designer

For Windows 95, 98 & NT

Version 6.0

For Windows 95, 98, 2000, & NT

Dynamic Pages

Supplemental User Guide

Boomerang Software, Inc.
90 Concord Ave.
Belmont, MA 02478
USA

Sales:

Telephone: 617-489-3000
Fax: 617-489-1486
Email: sales@boomerangsoftware.com

Technical Support:

Telephone: 617-484-4664
Fax: 617-489-1486
Email: techsupport@boomerangsoftware.com

Company URLs:

www.boomerangsoftware.com
www.secureshuttle.com
www.photoring.com

Copyright © 1999 - Boomerang Software, Inc.

Boomerang Software, Inc.

90 Concord Avenue
Belmont, Massachusetts 02478

Printed in the United States of America
November, 1999

WebShop 2000 User Guide
Edited by Tom Keydel and the staff at Boomerang Software

WebShop Designer™ is a trademark of Boomerang Software, Inc.

The WebShop User Guide contains copyrighted materials of Boomerang Software, Inc. All rights reserved. No part of this publication may be reproduced by any means without written permission from Boomerang Software, Inc. Making copies of any part of this book (or the accompanying software) for any purpose other than your own personal use is a violation of United States copyright laws and international treaty.

The WebShop program, its screens, icons, sequence, and organization are copyrighted materials of Boomerang Software, Inc. They may not be copied or duplicated in any medium without the written authorization of Boomerang Software, Inc.

The information contained in this manual is subject to change. Revisions to this manual or new editions may be issued as are deemed necessary by Boomerang Software, Inc. The information herein is believed to be, and is presented to be, accurate in all respects. However, Boomerang Software, Inc. cannot assume any responsibility or liability for any consequences resulting from the use or misuse of the software or the manual.

WebShop 2000, WebShop Designer, Internet/Intranet Design Shop GOLD, GOLD 2000, and Boomerang Software are trademarks of Boomerang Software, Inc. Windows, NT, and Microsoft are trademarks of Microsoft Corporation. Ulead GIF Animator is produced by and is a registered trademark of Ulead Corp. MGI Photo Suite is produced by and is a registered trademark of MGI. IBM, AOL, Sun Micro, and Netscape are trademarks of IBM Corp., America Online, Sun Micro Systems and Netscape Communications, successively.

All other product names identified in this book are trademarks or registered trademarks of their respective companies. They are used in this book in an editorial fashion only. No such use, or the use of any trade name, is intended to convey endorsement or other affiliation with this book or Boomerang software products.

Part Number: WSUG60-DP/PDF

Contents

Chapter 1. Dynamic Pages and ASP Technology	16-1
Introduction	16-1
Dynamic Page Window Display	16-1
Associated FrameSet Window Display	16-3
Dynamic Page Toolbar and Menu Options	16-4
Dynamic Page Actions and Icons	16-5
Dynamic Elements	16-7
Getting Started	16-7
<i>Property Sheet for Dynamic Page</i>	<i>16-8</i>
Specifying Your Dynamic Page Actions	16-11
Specifying the ASP Operations to be Performed	16-13
Query Action	16-13
Constructing Queries	16-16
<i>Delete Query</i>	<i>16-16</i>
<i>Insert Query</i>	<i>16-18</i>
<i>Select Query</i>	<i>16-19</i>
<i>Update Query</i>	<i>16-23</i>
Criteria Tab Operations	16-25
Send Mail Action	16-28
Feedback Section Action	16-29
Request Action	16-30
Redirect to Action	16-32
Condition Action	16-33
Statement Action	16-35
Cookie Action	16-36
Chapter 2. Sending Emails from a Form	2-1
Introduction	2-1
Registering WebShop's Active-X	2-1
Creating Your Email Function Using Dynamic Pages	2-2
<i>Creating a Dynamic Page</i>	<i>2-3</i>
<i>Creating a Form for Use with a Dynamic Page</i>	<i>2-3</i>
Creating a Dynamic Page with Send Mail Capability	2-4
<i>Defining Parameters on a Dynamic Page</i>	<i>2-4</i>
<i>Defining Variables on a Dynamic Page</i>	<i>2-5</i>
Chapter 3. Adding Form Input into a Database	3-1
Introduction	3-1
Create a Database	3-1
Updating Your Database Using Dynamic Pages	3-2
<i>Creating a Dynamic Page</i>	<i>3-2</i>
<i>Creating a Form for Use with a Dynamic Page</i>	<i>3-3</i>



<i>Importing the Database into the Project</i>	3-4
Creating Dynamic Page Elements For Updating a Database	3-5
<i>Defining Parameters for a Dynamic Page</i>	3-5
<i>Defining Variables for a Dynamic Page</i>	3-5
<i>Creating a Field Validation Test</i>	3-6
<i>Writing Information to the Database</i>	3-6
<i>Creating a Feedback Page</i>	3-7
<i>Creating an Error Page</i>	3-7
<i>Creating a Success Page</i>	3-8
Chapter 4. Creating a Query/Search Page	2-1
Introduction	2-1
Create a Database	2-1
Create the Test Data	2-2
Create a New Web Project to Perform a Query/Search	2-2
Importing the Database into the Project	2-2
Creating a Dynamic Page	2-3
<i>Defining Parameters for a Dynamic Page</i>	2-4
<i>Working with the Query Function</i>	2-4
<i>Creating Navigation Links For Feedback Pages</i>	2-8
Chapter 5. Redirecting URLs	5-1
Introduction	5-1
<i>Requesting ISP Information and Setting Up</i>	5-1
<i>Getting Started</i>	5-2
Create a New Project	5-2
<i>Assign "var_name" a value</i>	5-5
<i>Assign "var_any_error" a value</i>	5-5
<i>Redirect for Product 1</i>	5-8
<i>Redirect for Product 2</i>	5-8
Chapter 6. Cookies and Requesting Information	6-1
Introduction	6-1
Creating Your Project	6-2
<i>Creating a Dynamic Page</i>	6-2
<i>Creating a Form for use with a Dynamic Page</i>	6-2
Creating a Cookie	6-3
<i>Creating a Field Validation Test</i>	6-3
<i>Creating a Error Page</i>	6-4
<i>Writing a Cookie</i>	6-4
<i>Creating a Feedback Page</i>	6-5
Reading and Requesting Cookies	6-5
Requesting Information from the Server	6-6

Chapter 1.

Dynamic Pages and ASP Technology

Introduction

Dynamic Pages are the Web pages that perform database operations based on parameters and conditions that you define. Dynamic Pages contain Active Server Page (ASP) technology that lets you dynamically present information based on visitor interaction. The terms Dynamic Page and ASP Page are used interchangeably. Many Dynamic Pages use forms that ask a visitor for information. Based on visitor input, certain data can be displayed as feedback or special processing can be executed. The most frequent Dynamic Page applications include (1) sending email, (2) redirecting a viewer to a URL, (3) adding form input into a database, (4) creating a query/search page, or (5) issuing a cookie and requesting data from the server database based on a cookie that as been retrieved. However, the applications that you can design are limited only by your imagination.

Note: For more information on these specific applications, see the Supplemental User Guide found on the program CD titled "Dynamic Pages."

Topics covered in this chapter include:

- Dynamic Page Window Display
- Associated FrameSet Window Display
- Dynamic Page Toolbar and Menu Options
- Dynamic Page Actions and Icons
- Dynamic Elements
- Getting Started
- Specifying Your Dynamic Page Actions
- Specifying the ASP Operations to be Performed

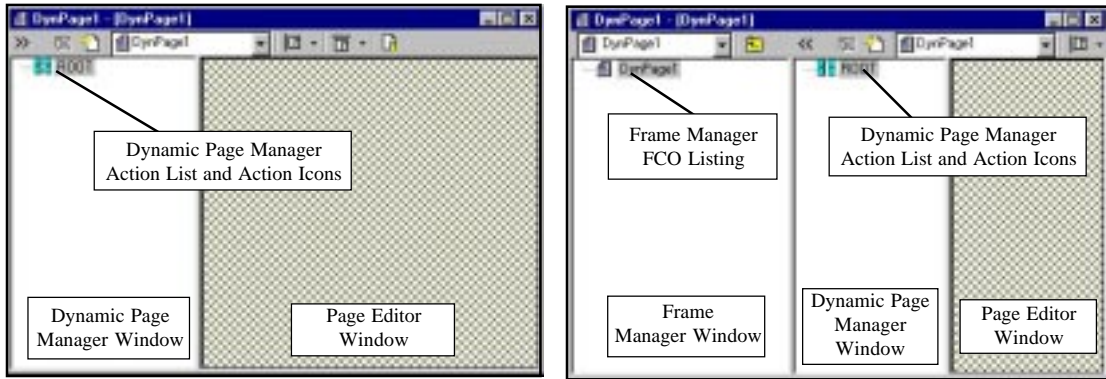
Dynamic Page Window Display

To manage the operations performed by a Dynamic Page, the Dynamic Page window that appears when you first open a Dynamic Page includes as at least two parts: (1) the Dynamic Page Manager window that is used to specify the Dynamic Page actions and (2) the Page Editor window that is used to design a Feedback Section that will conditionally display information to a Web site visitor.



Frame and Dynamic Page Manager windows are not visible to Web site visitor. For development purposes only

Page Editor windows are visible to Web site visitor only when Feedback Section is specified. Cross-hatch indicates that a Feedback Section is not specified.



Dynamic Page Window for a Single Dynamic Page

Dynamic Page Window for a Single Dynamic Page with Frame Manager Open

Figure 1.1 Dynamic Page Window for a Single Dynamic Page

It is important to keep in mind that a Dynamic Page contains both action specifications for controlling Dynamic Page operations and text and graphic specifications for controlling what appears in the browser window whenever a page display is specified. The Dynamic Page Manager window displays the actions that the Dynamic Page will perform and their sequence (from top to bottom). The Page Editor window displays the text and graphic specifications for Feedback Sections that appear *only when a Feedback Section action is specified*. When constructing a Dynamic Page, most likely, you will specify more than one Feedback Section action in your action list and will consequently define more than Feedback Section display. That's because a Dynamic Page lets you set up different information displays based upon varying conditions. That's the whole point of a Dynamic Page!

Unlike a Plain Page, a Dynamic Page lets you *dynamically* present information based upon varying conditions. It's up to you to define the conditions that will trigger each information display using the available Dynamic Page actions and then define the information display that the visitor will see using the page design tools that you have already learned. Every information display that you present must be specified as a Feedback Section before it can be viewed by a Web site visitor.

Logical Layout Note: An information display defined as Feedback Section is an resource to a Dynamic Page and does not appear on Logical Layout. Only the Dynamic Page icon will appear not the individual information displays. Clicking on a Dynamic Page icon from Logical Layout, lets you access the Dynamic Page action listing from which you can then access any of individual Feedback Sections that are associated with that Dynamic Page.

Physical Layout Note: An information display defined as Feedback Section does not appear on Physical Layout - only the *.ASP resource file that contains the individual Feedback Sections. If you use a Dynamic Page as a frame resource (FCO) for a FrameSet Page each Dynamic Page FCO becomes a separate *.ASP resource file. You may create





ASP resource files directly from Physical Layout. These resource files can then be used independently as Frame Content Overlays (FCOs). For more information see the chapter titled “WebShop Frames.”

FrameSet Note: If you create a Dynamic Page from Logical Layout and then click either Split Horizontal or Split Vertical on the Frames and Page Editor toolbar, the Dynamic Page is automatically converted into a FrameSet Page. By clicking either of the two Split buttons, you in effect are preparing a FrameSet Page that uses ASP resource files—not a Dynamic Page that stands alone as a Logical Page. If you continue to divide the frame areas, you will be creating separate Dynamic Page resource files. Each ASP resource file, when displayed as part of a FrameSet Page, contains its own action listing. Any FrameSet Page can make use of ASP resource files as individual Frame Content Overlays. For more information see the topic “Associated FrameSet Window Display” later in this chapter.

To resize the Dynamic Page Manager window or the Page Editor window

1. Position the cursor over the window border between the Dynamic Page Manager window and the Page Editor window. The cursor changes shape.
2. Click and drag the window border to a new position. The window border moves with the cursor.
3. Release the mouse at the new position. The window is resized.

Associated FrameSet Window Display

Any FrameSet Page that uses a Dynamic Page as a Frame Content Overlay for one of the specified frames will display the Dynamic Page Manager window and Page Editor window in the frame area specified.

Remember! ASP resource files when opened in Page Editor always generate a double window display that includes a Page Editor window and a Dynamic Page Manager window. To view the page as it will appear in a browser, you must minimize the Dynamic Page Manager window.

Normally, a FrameSet Page when opened in Page Editor displays a separate “window” for each frame area. Each frame area can have one or more separately assigned page resources (or Frame Content Overlays) that can appear in that frame area. The “windows” defined by each frame area can be resized by repositioning the window border between frame areas. By repositioning the window border between frame areas, you change the Frame Definition associated with the page.

When a Dynamic Page is displayed in a frame, the frame area assigned to the ASP resource file is divided into a Page Editor window and a Dynamic Page Manager window. These ASP resource file “windows” can be resized by repositioning the window border between the Page Editor window and a Dynamic Page Manager window. Resizing these window areas does *not* change the Frame Definition associated with the page. To affect the Frame Definition associated with the page, one of the frame “windows” between frame areas must be repositioned.



DynPages 1, 2, 3, and 4 are separate Frame Resources (FCOs) for one FrameSet page identified as FrameSet1 on Logical Layout

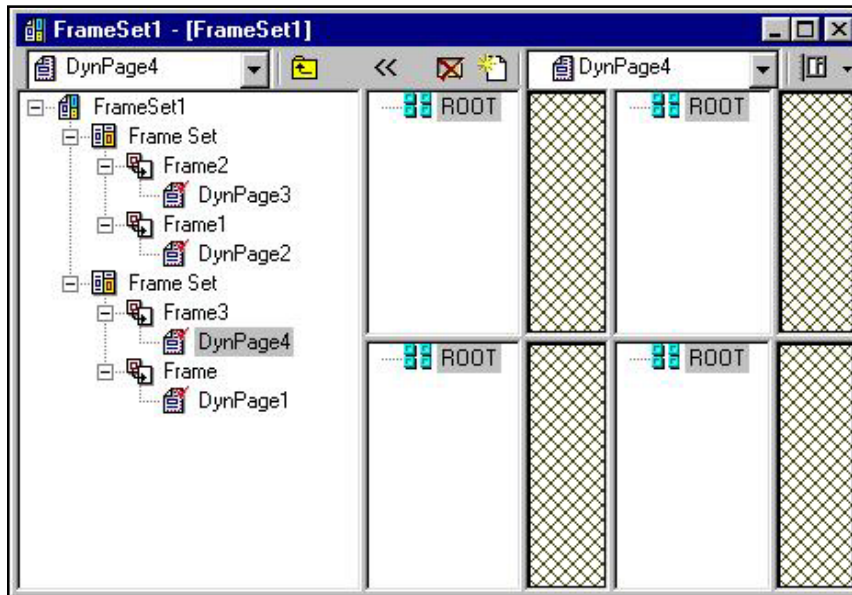


Figure 1.2–Frame Set Page Window with Multiple Dynamic Pages Specified

To resize a FrameSet window with an ASP resource file so that it is WYSIWYG

1. Position the cursor over the window border between the Dynamic Page Manager window and the Page Editor window. The cursor changes shape.
2. Click and drag the window border so that the Dynamic Page Manager window is completely obscured. The window is now resized to approximate what will be seen in the browser.

Frame Manager Note: By clicking the Frame Manager button on the Frames and Page Editor toolbar, the Frame Manager window appears. This window is not visible to a Web site visitor nor does it affect the Frame Definition associated with a page. To insure that you screen display is WYSIWYG minimize the Frame Manager window by clicking on the Frame Manager button.

Dynamic Page Toolbar and Menu Options

The Dynamic Page toolbar and menu appears whenever the cursor is placed inside a Dynamic Page. To access the menu options, use the **Dynamic Page** menu located on the Menu bar. The toolbar operations and menu commands are conditional and become active only when you click on an action list icon or when you place your cursor in the Page Editor window. If you right-click within the Dynamic Page Manager window, a pop-up menu appears. Many of the options listed below are available from this pop-up menu as well (Figure 1.11).

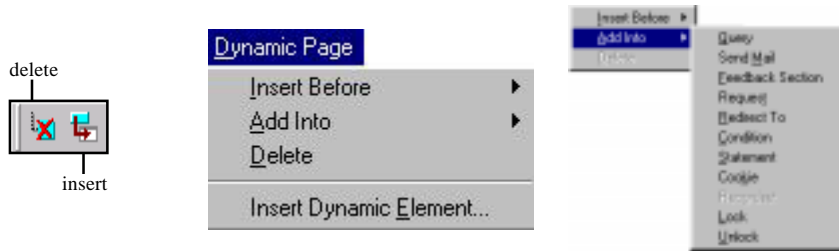


Figure 1.3 – Accessing Dynamic Page Operations

Toolbar Buttons and Menu Options

- **Delete.** Deletes the selected item from the resource tree in the Dynamic Page Manager window.
- **Insert (or Insert Dynamic Element).** Inserts a Dynamic Element into the Page Editor area to the right of the Dynamic Page Manager window. To activate the Insert feature, you must select one of the Frame-Content Overlay icons in the Dynamic Page Manager window.
- **Add Into.** Inserts a subordinate action into the Dynamic Page action list. The inserted action is subordinate to the selected action and is placed at the end of the action's subordinate list.
- **Insert Before.** Inserts an action into the Dynamic Page action list. The inserted action is placed before selected action and on the same level as the selected action

Dynamic Page Actions and Icons

A Dynamic Page is composed of actions that you specify using the Dynamic Page Manager. Everytime you insert an action into the Dynamic Page action listing, an action icon appears in the Dynamic Page Manager window. You manage the dynamic page actions by clicking the action icons and altering the action specifications. The action icons displayed in the Dynamic Page Manager window are not visible to a Web site visitor. They are presented for development purposes only.

Once your site is generated, the actions that you specify are compiled into ASP script. This ASP script is created by WebShop. Your action listing is organized in a hierarchical fashion. The ASP scripting generated by WebShop is compiled from the top down, with subordinate actions generating subordinate ASP script from left to right.

Note: The actions you specify for your Dynamic Page are executed only when the Dynamic Page is opened by a visitor to your Web site.

One of the hallmarks of an ASP page is that temporary values

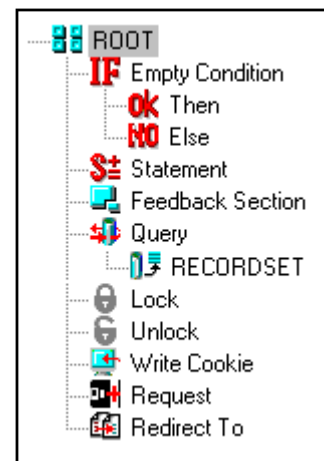


Figure 1.4 – Dynamic Page Icons



are created during the operation of the Dynamic Page. These temporary value can come from many sources. They can come from the Web server, if the action that obtains the data is defined as a Query. They can come from the visitor's PC, if the defined action is a Request. They can also be internally generated by the Dynamic Page itself. Keep in mind, however, that these temporary values are just that—temporary. All of your Dynamic Page values will return to their initial settings once the Web site visitor leaves the page. The only values that are permanent are the ones uploaded to your server after generating the Dynamic Page or the ones that you write out to the server database. This concept is important particularly when creating visitor IDs or record IDs.

The following actions can be executed from a Dynamic Page. Each action has a separate icon that indicates the action performed.

- **Query** - queries a database and returns a result, or executes a database operation on a selected record. There are only four operations that can be performed using **Query**: Select, Insert, Delete, or Update. To establish database connectivity, you must add the **RecordSet** operation into a **Query** operation before specifying the type of operation to be performed.
- **Send Mail** - issues email to one or more email addresses. Specify **From**, **To**, **Cc**, or **Subject**. Also specify an SMTP server address as necessary. In the body of the email to be sent, you can design a Plain Page fragment that can include images, tables, or dynamic elements.
- **Feedback Section** - displays feedback from a Plain Page that you design. A Feedback Section can be designed with (or without) Dynamic Elements. Use this action whenever you wish display text or graphics to the viewer.

Remember! Feedback Section defines the action of displaying information and while more than one Feedback Section can be associated with a Dynamic Page; only one resource file is created for the Dynamic Page, not several independent resource files. This means that a Feedback Section is not independent of its associated action listing nor can a Feedback Section make use of "predesigned" resources in the same way that a FrameSet Page can make use of independent page resources. If you want to create a Dynamic Page that uses "predesigned" resource files, create a FrameSet Page and use an ASP resource file as one of the Frame Content Overlays.

- **Request** - establishes the value for a Dynamic Page variable by obtaining data from: (1) a form, which is completed by the user during the current session, (2) a query string, which is an ASP-generated dynamic value, (3) a cookie, which is an generated value that is stored on the user's computer and identifies the user, (4) a client certificate, which is server value that enables the transmission of secure communication, (5) a server variable, which is a temporary (or permanent) value stored on the server, (6) an application element, which is a value that references a .DLL file or some other application element, or (7) a session element, which is a server value that references the current session's status. This action is used whenever you wish request information for subsequent ASP processing.
- **Redirect to** - redirects the viewer to different URL address or HTML page within the site. A redirect action can be specified using a parameter or by specifying the URL location directly.
- **Condition**- establishes two processing alternatives based upon a true or false condition. Condition statements can be nested within each other to provide multiple processing alternatives. Each processing result must be separately specified. Each condition statement



is composed of three parts: If-Then-Else. Use this action to make branching evaluations.

- **Statement** - sets (or resets) a variable, value, parameter, query output parameter, or cookie value as necessary. Use this action direct the processing of a Dynamic Page by assigning values directly.
- **Cookie** - writes a small text file to the user's hard disk so that he/she can be identified at a later date. This small text file (or cookie) can be retrieved by the server and matched to a cookie record. Use this action to create a unique identifier for visitors to your Web site.
- **RecordSet** - establishes a database connectivity so that query results can be displayed to a Web site visitor. To initiate this operation, you must first add a **Query** action and then add a **RecordSet** into the **Query**. The **Query** action opens the database and accesses current values whereas the RecordSet action allows you to manipulate the values.
- **Lock** - [investigate this more]
- **Unlock** - [investigate this more]

Dynamic Elements

A Dynamic Element is an element that has a changeable value based on the processing of a Dynamic Page. A Dynamic Element can be defined as a parameter, variable, or a query record field. To enable a field to display as a Dynamic Element, the value assigned to the parameter, variable, or query record field must be obtained from a defined source such as a database, an input form, a request action, or other Dynamic Page action.

To create a Dynamic Element based on a query, the database, table, and field must be specified through a query Select operation. Once the Dynamic Element has been specified, it can be inserted onto a Feedback Section as a query output field. Query output fields, by default, assume the field characteristics assigned by the source database.

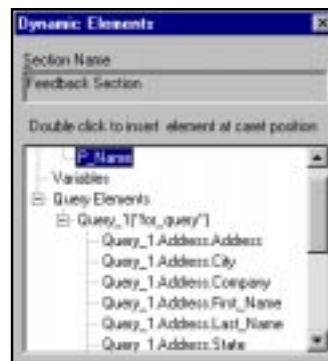


Figure 1.5 – Dynamic Elements Dialog

Getting Started

To get started you first need to create a Dynamic Page and set the properties for the Dynamic Page using the **Property Sheet for Dynamic Page** dialog box. The settings that you establish here are used later when performing Dynamic Page operations. However, you can add and modify these settings at any time.



To set up your Dynamic Page properties

1. Double-click the **ROOT** icon. The **Property Sheet for Dynamic Page** dialog box appears with the **Input Parameter** fields displayed.
2. Decide what you want to do.
 - To add a parameter, click inside the **New Parameter** field, type a parameter name, and click **Add**.

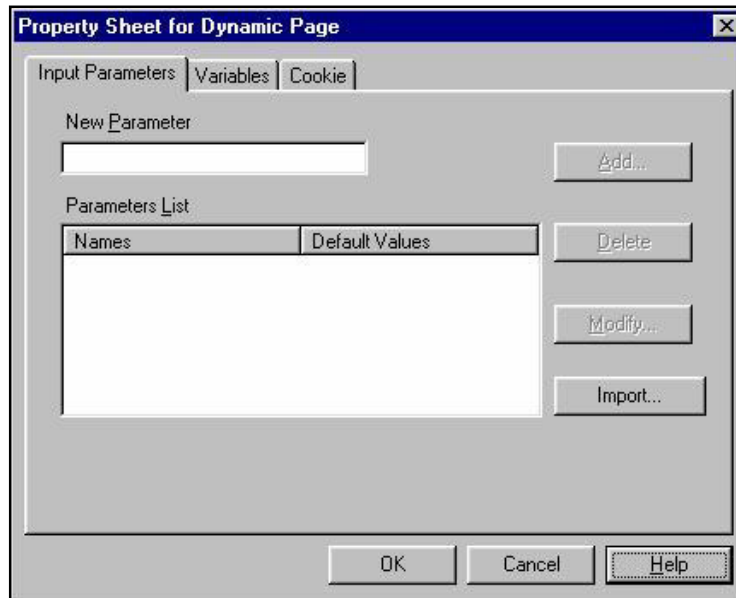


Figure 1.6 – Property Sheet for Dynamic Page Dialog Box

- To add a variable, click **Variables** tab. The **Variables** fields appear. Click inside the **New Variable** field, type a variable name, and click **Add**.
- To add a cookie key, click **Cookie** tab. The **Cookie** fields appear. Click inside the **New Key** field, type a key name, and click **Add**.

Note: To specify a cookie name, expiration date, processing for check security domain or path, as necessary, use the fields that display in the General section.

3. When done, click **OK**.

Property Sheet for Dynamic Page

The Property Sheet for Dynamic Page dialog box contains the following tabs: **Input Parameter**, **Variables**, and **Cookie**. Use this dialog box to define the input parameters, variables and cookie keys used when performing Dynamic Page operations. To save your specifications, you must click **OK** when exiting this dialog box.



Use the following definitions when use the Property Sheet for Dynamic Page dialog box.

- A *parameter* is an identified value that you define for use with a Dynamic Page. A parameter must have two parts: a name and a value. Parameter values are either assigned directly or imported and are defined as alphanumeric strings. You can leave a parameter empty or assign it a default value.
- A *variable* is very similar to a parameter *except that* a variable is used to perform an operation whereas the a parameter is used to hold a result. A variable must have two parts: a name and a value. Variable values are either assigned directly or imported and can defined either as and alphanumeric string or as a number. You can leave a variable empty or assign it a default value.
- A *cookie* is a small file located on the client's computer. These files contain persistent state information, such as user information. The server, where the Web site resides, can then request this information from the client browser to perform a task. Two tasks are required when using cookies. First, they must be sent and written to the client computer. Second they must be retrieved/requested by the server from the client computer. Each cookie must have a name and value (*cookie key*). Complex cookies, like cookie dictionaries, have more than one key associated with a page. Keys can have any of the following attributes:
 - *Domain*–This makes sure that when the cookie is requested it will only go back to its own domain.
 - *Expiration Date*–This is a date that the cookie will expire and remove itself from the client computer
 - *Path*–This is the file path that identifies the page from which the cookie issued; it also limits server requests for cookie information to the original page that issued it.
 - *Secure*–This feature will require a secure connection over the Internet before the cookie is sent.

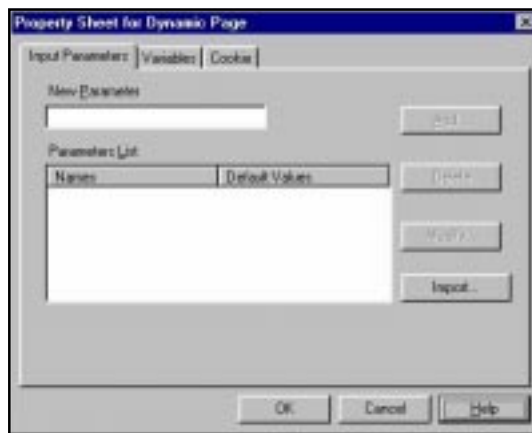


Figure 1.7 – Property Sheet for Dynamic Page Dialog Box (Input Parameters Tab)

To add a parameter

1. Click the **Input Parameters** tab.
2. Click inside the **New Parameter** field, type a parameter name, and click **Add**. The new parameter is listed in the **Parameter List**.

To assign a default value to a listed parameter

1. Select the parameter from the **Parameter List**, click the **Modify** button. The **Modify** dialog box appears.
2. Type the default value in the **Change Value** field and click **OK**. The default parameter



value is assigned and appears in the **Parameter List**.

Note: You can also use to Modify dialog box to change the parameter name.

To import a parameter or variable from another Dynamic Page

1. Click the **Import** button. The **Import List** dialog box appears.
2. Specify the Dynamic Page that currently contains the parameter or variable.
3. Select the parameter or variable from the **List of Parameters/Variables**.
4. Click the **Arrow** button to transfer your selection to the **Import List**.
5. To complete the import, click **Import**. The imported parameters or variables now appear in the **Parameter List**.

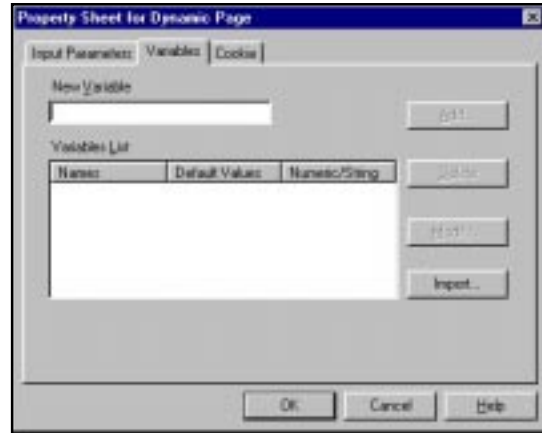


Figure 1.8 – Property Sheet for Dynamic Page Dialog Box (Variables Tab)

To add a variable

1. Click the **Variables** tab.
2. Click inside the **New Variable** field, type a variable name, and click **Add**. The new variable is listed in the **Variable List**.

To add a cookie key

1. Click the **Cookie** tab.
2. Click inside the **New Key** field, type a key name, and click **Add**. The cookie key is added the **Key Name List**.

To specify cookie key attributes

1. Click inside the **Cookie Name** field, type the cookie name, press **TAB** to move between fields, type values for: **Expires**, **Domain**, and **Path** as necessary. To specify the use of secure transmission, check the **Secure** box.
2. To save your attributes, click **OK**. The cookie attributes are specified.

Note: Cookie attributes apply to all Cookie Keys for the current Dynamic Page. These attributes cannot be individually assigned.

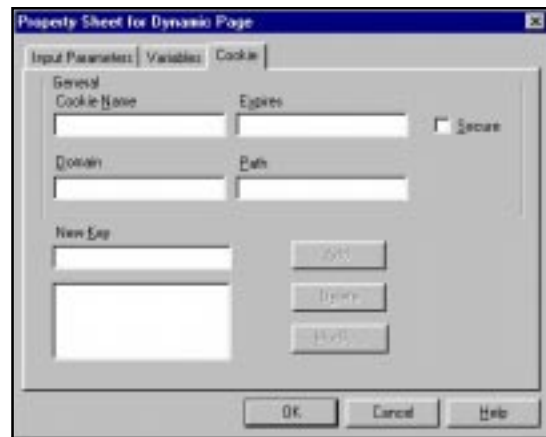


Figure 1.9 – Property Sheet for Dynamic Page Dialog Box (Cookie Tab)



Specifying Your Dynamic Page Actions

Once you have defined your parameters and variable, you are ready to start specifying your Dynamic Page actions. To specify an action, you can either insert an action *before* a selected action or add an action *into* the selected action. The distinction is important since your action listing is organized in a hierarchical fashion. Actions inserted *before* a selected action are on the same hierarchical level and will be performed top down. Actions added *into* the selected action are subordinated to the selected action and will be performed from left to right, one the initiating actions has been performed.

The menu option **Insert Before** places actions on the same hierarchy level as the selected action. The menu option **Add Into** places actions on the next subordinate level *at the end* of any previously entered subordinate actions.

To begin adding actions

1. Click the **Root** icon to select it.
2. Right-click and select **Add Into**. A fly-out menu appears.
3. Select the action that you want. The action icon is inserted at the first hierarchy level below the **Root** icon.

To add additional actions

1. Decide if you want to add the action on the same level in front of a selected action or as a subordinate action OR if you want to add a new subordinate action on the same level or a new action that is one hierarchy level up
 - *To add an action on the same level*, select an action, right-click and select **Insert Before**. A fly-out menu appears. Select the action that you want. An action icon is inserted before of the action selected at the same hierarchy level as the selected action.
 - *To add an action as a subordinate action*, select the action that begins the subordinate list, right-click and select **Add Into**. A fly-out menu appears. Select the action that you want. An action icon is inserted at the end of the subordinate list as a first-level subordinate action.
 - *To add a new subordinate action on the same level*, select a subordinate action right-click and select **Insert Before**. A fly-out menu appears. Select the action that you want. An action icon is inserted before of the action selected at the same hierarchy level as the selected action.
 - *To add a new that is an action that is one hierarchy level up*, select the next closest action that is one-level higher, right-click and select **Add Into**. A fly-out menu appears. Select the action that you want. An action icon is inserted at the end of the subordinate list as a first-level subordinate action.

Frequent Uses and Associated Conditions or Restrictions

The following table identifies the frequent uses associated with each action and any Associated Conditions or Restrictions.

Action	Frequent Use and Associated Conditions or Restrictions
--------	--



Query	Used to obtain information from the server database; no specific conditions or restrictions apply. This action can be inserted onto a Dynamic Page at any point. There are four query operations—Delete, Insert, Select, and Update; use the Criteria tab to limit the execution of the query to specific conditions. The Criteria tab is available with Delete, Select and Update queries. When reporting Query Results back to a Web site visitor using a Feedback Section, insert a RecordSet action after the Query action and before the Feedback Section action.
Send Mail	Used to send email notification over the Internet; no specific conditions or restrictions apply. This action can be inserted onto a Dynamic Page at any point.
Feedback Section	Used to display information to the Web site visitor; no specific conditions or restrictions apply. This action can be inserted onto a Dynamic Page at any point. When used to report Query Results to a site visitor, insert a RecordSet action after the Query action and before the Feedback Section. When inserting Dynamic text, the Dynamic elements must be defined as a parameter, variable or query output element using either the Property Sheet for the Dynamic Page or Query Properties dialog boxes prior to being inserted onto a Feedback Section. If used with a Condition action, the Feedback Section will only display if the initiating “OK” or “NO” action is performed.
Request	Used to retrieve and store data in a Dynamic Page; no specific conditions or restrictions apply. This action can be inserted onto a Dynamic Page at any point. Variables used to initiate and store the requested data must be defined using the Property Sheet for the Dynamic Page dialog box prior to specifying the Request action.
Redirect to	Used to retrieve and store data in a Dynamic Page; no specific conditions or restrictions apply. This action can be inserted onto a Dynamic Page at any point. Variables used to initiate and store the requested data must be defined prior to specifying the Request actions.
Condition	Used to define two processing alternatives based upon a true or false condition; no specific conditions or restrictions apply. This action can be inserted onto a Dynamic Page at any point. The processing alternatives specified by a condition statement are only executed when (and if) the initiating condition is evaluated; the resulting actions are subordinate to the initiating action. If a subordinating action statement (either “OK” or “NO”) is left unspecified, then the next initiating action is performed. If you wish to make multiple evaluations that specify actions for <u>only</u> true (or false) conditions, you can “nest” Condition actions within one another. The last subordinated action statement in the “nested” series, if left unspecified, causes the next initiating action to be performed. This enables you can chain together “one-sided” condition statements if you want.
Statement	Used to assign a value to a variable while a Dynamic Page is completing its processing; no specific conditions or restrictions apply. This



	action can be inserted onto a Dynamic Page at any point. To assign a value, you must specify an operation that will construct the variable's new value given the variable's current value.
Cookie	Used to assign a value to a variable and then write that value to the visitor's PC. To assign a value, you must specify both the cookie key that will store the temporary value and the variable that contains the value that you wish to write onto the visitor's PC; no specific conditions or restrictions apply. This action cannot be inserted if a Feedback Section has already inserted. The Cookie action must be performed prior to the display of a Feedback Section.
RecordSet	Used establishes database connectivity so that query results can be displayed to a Web site visitor; no specific conditions or restrictions apply. This action can only be inserted once the initiating Query action has specified which database will be queried.
Lock	[needs to be clarified]
Unlock	[needs to be clarified]

Specifying the ASP Operations to be Performed

Once you have specified one or more Dynamic Page actions, you are ready to start specifying the ASP operations to be performed. Most actions have at least one dialog box that you use set up the desired operation. The only exceptions are RecordSet, Lock, and Unlock. These actions do not require the use of a dialog box.

Action	Associated Dialog Box
Query	Choose Data Source for New Query Dialog Box
Send Mail	Send Mail Properties Dialog Box
Feedback Section	Feedback Section Dialog Box
Request	Request from User Dialog Box
Redirect to	Property Sheet for Redirect Action Dialog Box
Condition	Set Condition Dialog Box
Statement	Set Variable To Dialog Box
Cookie	Write Cookie Dialog Box

To specify or modify properties of any action

- Double-click on the action icon that appears in the Dynamic Page Manager window. The associated dialog box appears.

Query Action

A **Query** action establishes the database connectivity between the Dynamic page and the server database. With Query, you can Select, Insert, Delete, or Update records in the database.



To specify a new Query

1. Double-click the **Query** icon to open the **Choose Data Source for New Query** dialog box.
2. Using the resource tree, navigate to the database that you want and select it. The **Query Type** field becomes active.
3. Select the type of query you want using the **Query Type** drop-down list.
4. Click **Edit Query**. The **Query Properties** dialog box appears.

Note: Every query is assign a name that includes a sequential query number and database. This query name appears in the title bar of the Query Properties dialog box. It also appears in the Dynamic Page Manager window next to the Query icon. Use this query name to identify the source of any data elements that you create by accessing the selected database.

5. Specify your query properties, as necessary, and click **OK**. The Query action is specified.

Note: For more information about this dialog box, see the topic "Query Properties Dialog Box" below.

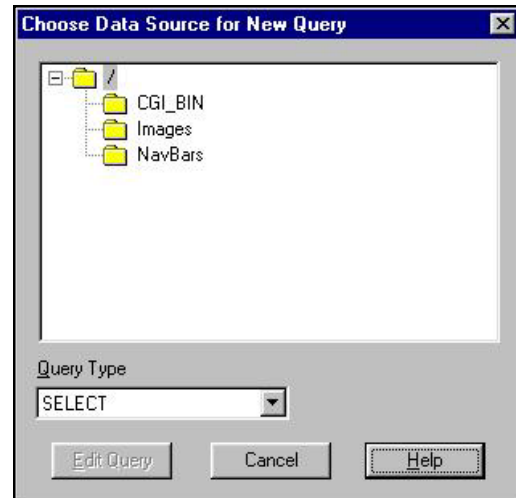


Figure 1.10 – Choose Data Source for New Query Dialog Box

To display query records on a Feedback Section

1. Add a **RecordSet** into the **Query** and add a **Feedback Section** into the RecordSet.

Note: You will not be able to add a RecordSet into the query until the database and type of query are specified.

2. Double-click on the Feedback Section icon. The Page Editor window becomes active.
3. Click the **Dynamic Element Insertion** button on the Dynamic Page toolbar. The **Dynamic Text Element** dialog box appears. The data elements associated with the query appear.

Note: The Dynamic Text Element dialog box organizes dynamic elements by Parameters, Variables, and Query Elements. Each of these groupings can be expanded or collapsed using the resource tree controls. To identify the query elements associated with a specific query, the query name is listed next to each query element.

4. Position the cursor in the Page Editor window where you want the dynamic element to appear and click. A blinking insertion cursor appears.
5. Position the cursor over the name of the element to be inserted and double-click. The dynamic element is inserted.



Query Properties Dialog Box

The **Query Properties** dialog box lets you set properties for a database query. The options displayed in this dialog box are based on the Query Type you have selected in the **Choose Data Source for New Query** dialog box. Each query type will display a different set of tabs. The Query types are Delete, Insert, Select, or Update.

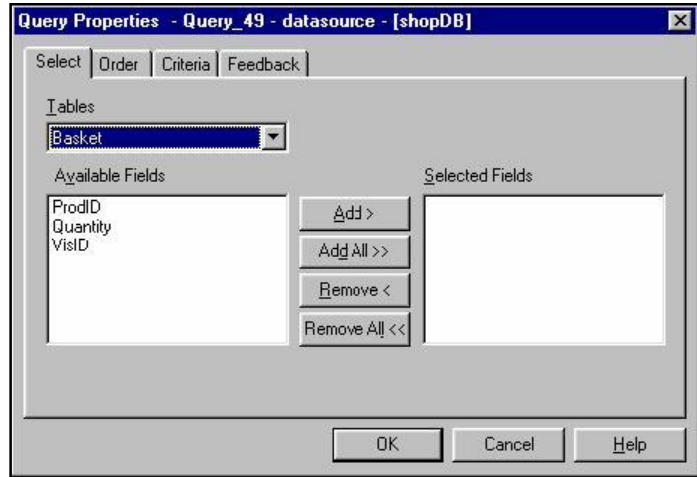


Figure 1.13 – Query Properties Dialog Box (Select Tab)

Delete Query

A Delete query deletes one or more records from the database, according to the condition(s) you specify. If the Delete query is selected, the available tabs are:

- **Delete tab.** Select the table associated with the record(s) to be deleted.
- **Criteria tab.** Specify the criteria necessary for the query to be performed.

Insert Query

An Insert query adds a new record to the specified table of the database. If an Insert query is selected the available tab is:

- **Insert tab.** Select the table and assign field values to the record that will be inserted.

Select Query

A Select query displays the records from the specified database, according to the condition(s) you specify. You may order the order the data display and specify how many records to display. If a Select query is selected, the available tabs are:

- **Select tab.** Select the table where the record(s) are stored and select the field(s) that you wish to query.
- **Order tab.** Select the table and the field order that you want to appear when the query results are displayed.
- **Criteria tab.** Specify the conditions necessary for the query to be performed.
- **Feedback tab.** Specify how many records should appear when the query results are displayed.



Update Query

An Update query records new data into the database, according to the condition(s) you specify. If an Update query is selected, the available tabs are:

- **Update tab.** Select the table where the record(s) are stored, select the field(s) that you wish to update and assign values to fields.
- **Criteria tab.** Specify the conditions necessary for the query to be performed.

Constructing Queries

Delete Query

To specify a new Delete Query

1. Click the **Root** icon to select it.
2. Right-click and select **Add Into**. A fly-out menu appears.
3. Select **Query**. The **Query** icon is inserted at the first hierarchy level below the **Root** icon.
4. Double-click the **Query** icon to open the **Choose Data Source for New Query** dialog box.
5. Using the resource tree, navigate to the database that you want and select it. The **Query Type** field becomes active.
6. Select **Delete**.
7. Click **Edit Query**. The **Query Properties** dialog box appear.

To select the table associated with the record(s) to be deleted

1. Click the **Tables** tab. The **Query Table** fields appear.
2. Click the **Tables** drop-down list and select the table associated with the record(s) to be deleted.
3. When done, decide what criteria must be satisfied before this query will be executed.

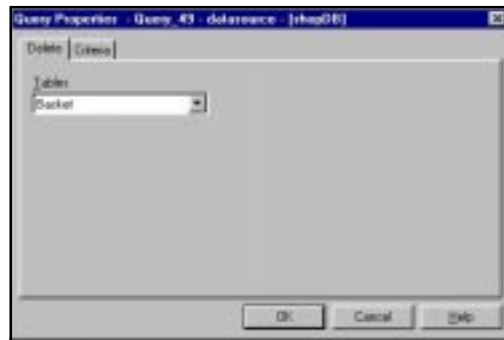


Figure 1.14 – Query Properties Dialog Box

To specify the criteria necessary for the query to be performed

1. Click the **Criteria** tab. The **Query Criteria** fields appear.
2. Click either **Insert Logic** or **Add Logic**. The **Choose Logic Operation** dialog box appears.
3. Select either AND or OR from the **Logic Type** drop-down list and click **OK**. The specified operation is inserted as the **Root** operation.
4. Select the Root operation and insert any additional operations. The insertion cursor appears to the right of the *selected* operation icon.



Note: To insert additional operations, you must decide (1) if you want to insert an operation at the same hierarchy (use Add Logic) or (2) if you want to insert an operation at the same hierarchical level and subordinate the selected operation to the operation that you are about to insert (use Insert Logic). An inserted operation is always placed after the last operation at the same hierarchy level. You may click and drag an operation (and all of its subordinate operations) to any position in the hierarchy. Selecting an operation is different from selecting a condition. When you select a condition, the condition statement is highlighted. When you select an operation, an insertion cursor appears.



Figure 1.15 – Query Properties Dialog Box (CriteriaTab)

5. Click either **Insert Logic** or **Add Logic**. The **Choose Logic Operation** dialog box appears.
6. Select either AND or OR from the **Logic Type** drop-down list and click **OK**. The specified operation is inserted. The insertion cursor appears to the right of the *inserted* operation icon.
7. Select an operation icon and insert condition statements as necessary to specify the query criteria. The insertion cursor appears to the right of the operation icon.
8. Click **Add Condition**. The **Add Condition** dialog box appears.
9. Click on the **Available Fields** drop-down arrow and select the field that you want.
10. Click on the **Binary Operation** drop-down arrow and select the operation that you want.
11. Select the tab that you want to specify the right operand.
 - To specify a fixed value, click the **Value** tab. The **Value** input line appears. Type a fixed value.
 - To specify a form control, click the **Control** tab. The **Form controls** input line appears. Click the drop-down arrow and select the control that you want to use.
12. When done, click **OK**. The specified condition statement appears as a subordinate statement under the selected operation. You may add as many condition statements to an operation as necessary.



Figure 1.16 –Choose Logic Operation Dialog Box (Use with QueryCriteria)



Figure 1.17 – Add Condition Dialog Box



13. When done, click **OK**. The Delete Query is specified.

To execute the query, a user must be taken to the Dynamic Page and the query criteria must be met. Normally, you will want to provide a delete warning prior to executing a Delete Query and some confirmation feedback that the Delete Query was completed successfully.

Insert Query

To specify a new Insert query

1. Click the **Root** icon to select it.
2. Right-click and select **Add Into**. A fly-out menu appears.
3. Select **Query**. The **Query** icon is inserted at the first hierarchy level below the **Root** icon.
4. Double-click the **Query** icon to open the **Choose Data Source for New Query** dialog box.
5. Using the resource tree, navigate to the database that you want and select it. The **Query Type** field becomes active.
6. Select **Insert**.
7. Click **Edit Query**. The **Query Properties** dialog box appear.



Figure 1.18 – Query Properties Dialog Box (Insert Tab)

To select the table and assign field values to the record that will be inserted

1. Click the **Insert** tab. The **Query Insert** fields appear.
2. Click the **Tables** drop-down list and select the table that will receive the inserted records. The fields that are currently defined for that table appear in the **Fields and Values** section.

Note: By default all fields have the value "empty" until specifically assigned a value.

3. Select a field and click **Edit**. The **Set Value** dialog box appears. Use this dialog box to insert values explicitly, as numbers or alphanumeric strings, or to select a form control that will dynamically insert a value.
 - To insert values explicitly, click the **Value** tab and type the value that you want to assign. Click **OK**. The entered value appears as the field value in the **Fields and Values** section.
 - To select a form control, click the **Control** tab and click the drop-down arrow. A scrollable list of controls appears. Select the control you want to assign. Click **OK**. The selected control appears as the field value in the **Fields and Values** section.



Figure 1.19 – Set Value Dialog Box



- To omit a field from the query, select it and click **Clear**. The word “empty” appears as the field value in the **Fields and Values** section. Omitted fields receive a null value.
- When done, click **OK**. The Insert query is specified.

To execute the query, a user must be taken to the Dynamic Page where the Insert Query has been specified. Normally, an Insert Query is specified after some data has been collected, either through a form using the **Submit** button. You should provide an insert warning prior to executing an Insert Query and also provide some confirmation feedback that the Insert Query was completed successfully.

Select Query

To specify a new Select query

- Click the **Root** icon to select it.
- Right-click and select **Add Into**. A fly-out menu appears.
- Select **Query**. The **Query** icon is inserted at the first hierarchy level below the **Root** icon.
- Double-click the **Query** icon to open the **Choose Data Source for New Query** dialog box.
- Using the resource tree, navigate to the database that you want and select it. The **Query Type** field becomes active.
- Select **Select**.
- Click **Edit Query**. The **Query Properties** dialog box appear.

To select the table

- Click the **Select** tab. The **Query Select** fields appear.
- Click the **Tables** drop-down list and select the table that contains the records that you wish to select. The fields that are currently defined for that table appear in the **Available Fields** section. To select a field, you must insert the desired fields into the **Selected Fields** section from the **Available Fields** section.

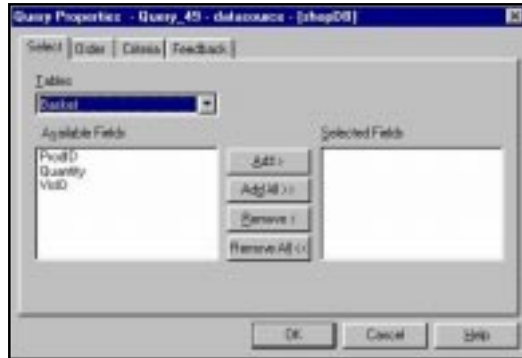


Figure 1.20 – Query Properties Dialog Box (CriteriaTab)

To specify the fields that query will select

- To select fields, decide how you wish to insert fields into the **Selected Fields** section.
 - To insert fields one by one, select a field and click **Add >**. Only the selected field is added to the **Selected Fields** section.



- To select more than one field, hold down the **CNTR** key as you select your fields and click **Add >**. All of the selected fields are added.
 - To insert all of the **Available Fields**, click **Add All >>**. All of the table fields are added to the **Selected Fields** section.
2. To remove fields, decide how you wish to remove fields from the **Selected Fields** section.
 - To remove fields one by one, select the field and click **Remove <**. The selected field is removed.
 - To remove more than one field, hold down the **CNTR** key as you select your fields and click **Remove <**. All of the selected fields are removed.
 - To remove all of the fields from **Selected Fields**, click **Remove All <<**. The fields are removed.
 3. When done, decide how you want your query results to be organized.

To select the display order that you want to appear

1. Click the **Order** tab. The **Query Order** fields appear.
2. Click the **Tables** drop-down list and select the table that contains the fields that you want to use to order your query results. To sequence your query result in a specific order, you must insert a field sorting order into the **Fields to Order By** section. These fields are used sort the query feedback on the Feedback Section that displays the query result.

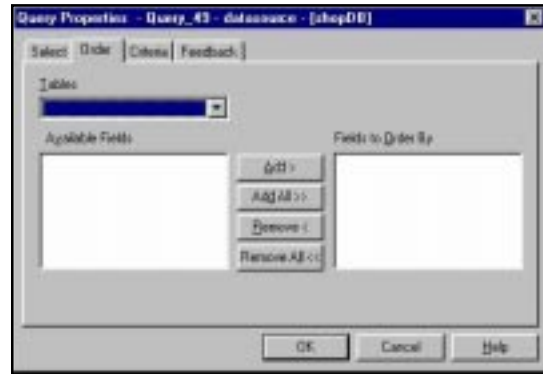


Figure 1.22 – Query Properties Dialog Box (Order Tab)

Note: When inserted, field names display a small arrow to the left of the field name. The arrow's direction indicates whether the sorting will be ascending or descending. If the arrow points upward, the sort is ascending. If the arrow points downward, the sort is descending. To change the arrow's direction, click on the arrow icon. The direction is reversed.

3. To select fields, decide how you wish to insert fields into the **Fields to Order by** section.
 - To insert fields one by one, select a field and click **Add >**. Only the selected field is added to the **Fields to Order by** section.
 - To select more than one field, hold down the **CNTR** key as you select your fields and click **Add >**. All of the Fields to Order by are added.
 - To insert all of the **Available Fields**, click **Add All >>**. All of the table fields are added to the **Fields to Order by** section.
4. To remove fields, decide how you wish to remove fields from the **Fields to Order by** section.
 - To remove fields one by one, select the field and click **Remove <**. Only the selected field is removed.



- To remove more than one field, hold down the **CNTR** key as you select your fields and click **Remove <**. All of the Fields to Order by are removed.
 - To remove all of the fields from **Fields to Order by**, click **Remove All <<**. The fields are removed.
5. When done, decide what criteria must be satisfied before this query will be executed.

To specify the criteria necessary for the query to be performed

1. Click the **Criteria** tab. The **Query Criteria** fields appear.
2. Click either **Insert Logic** or **Add Logic**. The **Choose Logic Operation** dialog box appears.
3. Select either AND or OR from the **Logic Type** drop-down list and click **OK**. The specified operation is inserted as the **Root** operation.
4. Select the Root operation and insert any additional operations. The insertion cursor appears to the right of the *selected* operation icon.

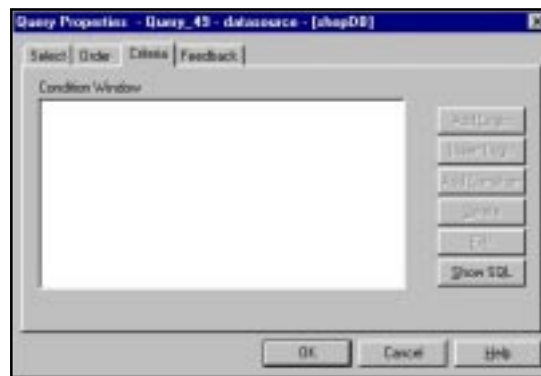


Figure 1.21 – Query Properties Dialog Box (CriteriaTab)

*Note: To insert additional operations, you must decide (1) if you want to insert an operation at the same hierarchy (use **Add Logic**) or (2) if you want to insert an operation at the same hierarchical level and subordinate the selected operation to the operation that you are about to insert (use **Insert Logic**). An inserted operation is always placed after the last operation at the same hierarchy level. You may click and drag an operation (and all of its subordinate operations) to any position in the hierarchy. Selecting an operation is different from selecting a condition. When you select a condition, the condition statement is highlighted. When you select an operation, an insertion cursor appears.*

5. Click either **Insert Logic** or **Add Logic**. The **Choose Logic Operation** dialog box appears.
6. Select either AND or OR from the **Logic Type** drop-down list and click **OK**. The specified operation is inserted. The insertion cursor appears to the right of the *inserted* operation icon.
7. Select an operation icon and insert condition statements as necessary to specify the query criteria. The insertion cursor appears to the right of the operation icon.
8. Click **Add Condition**. The **Add Condition** dialog box appears.
9. Click on the **Available Fields** drop-down arrow and select the field that you want.
10. Click on the **Binary Operation** drop-down arrow and select the operation that you want.
11. Select the tab that you want to use to specify the right operand.
 - To specify a fixed value, click the **Value** tab. The **Value** input line appears. Type a fixed value.



- To specify a form control, click the **Control** tab. The **Form controls** input line appears. Click the drop-down arrow and select the control that you want to use.
 - To specify a database field value, click the **Join** tab. The **Field to join by** input line appears. Click the drop-down arrow and select the database field that you want to use.
13. Repeat Steps 7–11. You may add as many condition statements to an operation as necessary.
 14. When done, decide what size you want the query feedback “page” to be.

To specify how many records will appear when the query results are displayed

1. Click the **Feedback** tab. The **Feedback** fields appear.
2. Decide what you want to do:
 - To display all result records on one page, leave the **Show All** radio button selected.
 - To display result recorder over multiple pages, click the **Custom** radio button and specify the number of record lines to display.
3. When done, click **OK**. The Select query is specified.



Figure 1.23 – Query Properties Dialog Box (Feedback -Tab)

To execute the query, a user must be taken to the Dynamic Page where the Select query has been specified. Normally, a Select query is specified after a user has specified what data he/she wishes to see. Most likely, you will specify this query after some data has been collected through a form using the **Submit** button. If you use a **Custom** feedback page, you must be sure to insert navigation links on to each Feedback Section that displays the query result. Query results are displayed as a “rolling window,” that is, as a subset of the total record number. Therefore you must evaluate the relative position of subset window to determine how the text links should display.

To insert the necessary text links, use the query variables, **Navigation_Forward** and **Navigation_Backward**. To evaluate your query results, use the query elements, **Low_Range** and **High_Range** to identify the relative position of “window” given the total record number that must be displayed.

Note: The query element, Low_Range, equals 0 or 1 depending on whether results have been found and the query element, High_Range, equals the specified number of display lines to display on each feedback page. These query elements appear on the Query tab drop-down list from the Set Condition dialog box. Use this dialog box to specify the evaluation criteria for each feedback page display.

Since your evaluations are sequential, you are setting up conditions to test for first page display, middle page(s) display, and last page display.

Note: Since evaluations are sequential, there is no test for a single page result that does not



include any navigation links- only first page, middle age, and last page.

You evaluation tests should include

- **Low_Range** = 0 (no result condition)
- **Low_Range** = 1 (feedback required) AND **High_Range** is less than the total record number (first result page; use Forward Link only)
ELSE
- **Low_Range** = 1 (feedback required) AND **High_Range** is less than the total record number (more than on page required)
ELSE
- **Low_Range** = 1 (feedback required) AND **High_Range** greater than or equal to the total record number (last page display; use Backward Link only)

Your feedback pages should be set up to provide a link that allows the user to input a new search.

Update Query

To specify an Update query

1. Click the **Root** icon to select it.
2. Right-click and select **Add Into**. A fly-out menu appears.
3. Select **Query**. The **Query** icon is inserted at the first hierarchy level below the **Root** icon.
4. Double-click the **Query** icon to open the **Choose Data Source for New Query** dialog box.
5. Using the resource tree, navigate to the database that you want and select it. The **Query Type** field becomes active.
6. Select **Update**.
7. Click **Edit Query**. The **Query Properties** dialog box appear.

To select the table and assign field values to the record that will be inserted

1. Click the **Update** tab. The **Query Update** fields appear.
2. Click the **Tables** drop-down list and select the table that will receive the inserted records. The fields that are currently defined for that table appear in the **Fields and Values** section.

Note: By default all fields have the value "empty" until specifically assigned a value.

3. Select a field and click **Edit**. The

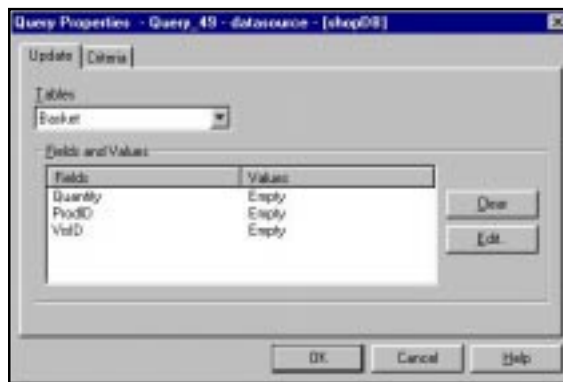


Figure 1.24 – Query Properties Dialog Box (Update Tab)



Set Value dialog box appears. Use this dialog box to insert values explicitly, as numbers or alphanumeric strings, or to select a form control that will dynamically insert a value.

- To insert values explicitly, click the **Value** tab and type the value that you want to assign. Click **OK**. The entered value appears as the field value in the **Fields and Values** section.
 - To select a form control, click the **Control** tab and click the drop-down arrow. A scrollable list of controls appears. Select the control you want to assign. Click **OK**. The selected control appears as the field value in the **Fields and Values** section.
 - To omit a field from the query, select it and click **Clear**. The word “empty” appears as the field value in the **Fields and Values** section. Omitted fields receive a null value.
3. When done, decide what criteria must be satisfied before this query will be executed.

To specify the criteria necessary for the query to be performed

1. Click the **Criteria** tab. The **Query Criteria** fields appear.
2. Click either **Insert Logic** or **Add Logic**. The **Choose Logic Operation** dialog box appears.
3. Select either AND or OR from the **Logic Type** drop-down list and click **OK**. The specified operation is inserted as the **Root** operation.
4. Select the Root operation and insert any additional operations. The insertion cursor appears to the right of the *selected* operation icon.



Figure 1.25 – Query Properties Dialog Box (Criteria Tab)

*Note: To insert additional operations, you must decide (1) if you want to insert an operation at the same hierarchy (use **Add Logic**) or (2) if you want to insert an operation at the same hierarchical level and subordinate the selected operation to the operation that you are about to insert (use **Insert Logic**). An inserted operation is always placed after the last operation at the same hierarchy level. You may click and drag an operation (and all of its subordinate operations) to any position in the hierarchy. Selecting an operation is different from selecting a condition. When you select a condition, the condition statement is highlighted. When you select an operation, an insertion cursor appears.*

5. Click either **Insert Logic** or **Add Logic**. The **Choose Logic Operation** dialog box appears.
6. Select either AND or OR from the **Logic Type** drop-down list and click **OK**. The specified operation is inserted. The insertion cursor appears to the right of the *inserted* operation icon.
7. Select an operation icon and insert condition statements as necessary to specify the query criteria. The insertion cursor appears to the right of the operation icon.
8. Click **Add Condition**. The **Add Condition** dialog box appears.



9. Click on the **Available Fields** drop-down arrow and select the field that you want.
10. Click on the **Binary Operation** drop-down arrow and select the operation that you want.
11. Select the tab that you want to use to specify the right operand.
 - To specify a fixed value, click the **Value** tab. The **Value** input line appears. Type a fixed value.
 - To specify a form control, click the **Control** tab. The **Form controls** input line appears. Click the drop-down arrow and select the control that you want to use.
 - To specify a database field value, click the **Join** tab. The **Field to join by** input line appears. Click the drop-down arrow and select the database field that you want to use.
13. Repeat Steps 7–11. You may add as many condition statements to an operation as necessary.
14. When done, click **OK**. The Update query is specified.

To execute the query, a user must be taken to the Dynamic Page where the Select query has been specified. Normally, an Update query is specified after a user has specified what data he/she wishes to see.

Criteria Tab Operations

Because the Query Properties **Criteria** tab is both difficult and important, the specific criteria operations that you can set-up using the Criteria tab are covered here.

The **Criteria** tab fields includes the Condition window and several buttons from managing the binary operations that specify your criteria.

When first displayed with a new query, the Condition window is empty. As you build your criteria's binary operations, they appear in Condition window as a logical condition tree. The condition that comprises your query criteria is logically connected by either an AND operation or an OR binary operation. By specifying an AND operation, you are saying that the query operation can only be performed when one or more conditions are true.

By specifying an OR operation, you are saying that the query operation can only be performed when any of the listed conditions are true.

Remember! When building your criteria, you are inserting two different kinds of selectable objects, a logical operation object and a condition statement object.

The first Logical Operation that you add is the **Root** operation. You cannot delete the **Root** operation once it is inserted, however you can change both its associated logical operation and its conditions.

Logical operations are organized hierarchically as selectable objects. Operations that contain either a subordinate operation or condition statement appear with a display control. Use this display control to expand or collapse operation. When the control displays a plus (+) sign, the operation is collapsed. When the control displays a minus (-) sign, the operation is expanded.

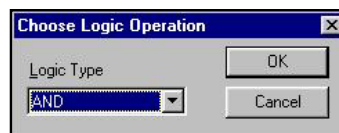


Figure 1.26 –Choose Logic Operation Dialog Box (Use with QueryCriteria)



When adding condition statements, the **Add Condition** dialog box appears. This dialog box has three input areas: a left operand, a center operation and a right operand. Use these input areas to construct your evaluation condition. The left operand contains a drop-down box that identifies the available fields for the selected table. The center operation contains a drop-down box that identifies the available binary operations. These operations are used to evaluate for a true conditions and are used to compare the left and right operands.



Figure 1.27 – Add Condition Dialog Box

When used, the available operations will generate a true result:

- < - if the left operand is less than the right operand
- <= - if the left operand is less than or equal to the right operand
- <> - if the left operand is not equal to the right operand
- = - if the left operand is equal to the right operand
- > - if the left operand is more than the right operand
- >= - if the left operand is more than or equal to the right operand

BEGINS WITH - if the left operand begins with the character(s) of the right operand

BETWEEN - if the left operand is in the range defined by the minimum and maximum specified in the Value section. In this case you must enter two values in the Value section.

CONTAINS - if the left operand contains the character(s) displayed in the Value section;

ENDS WITH - if the left operand ends with the character(s) displayed in the Value section.

The right operand input area lets you select between an explicitly entered value, a form control value, or a database field value. To specify your right operand, click any one of available tabs.

Note: The Join tab is only available with a Select query. Also, if you choose the BETWEEN Operation, two value specifications are required instead of one. Set the upper and lower limits using the addition input lines created with this operation.

Value Tab. Click the input line and type a fixed value.

Control Tab. Click the drop-down arrow to display the form control names defined for use with this query. Select the control that you want to use.

Join Tab. Click the drop-down arrow to display the available fields associated with the database table specified in the **Query Properties** dialog box under the **Select** tab.



Note: It is recommended that you only use the Equals operation when specifying a Join condition.

Entering Criteria Operations and Conditions

To insert the Root operation

1. Click either **Insert Logic** or **Add Logic**. The **Choose Logic Operation** dialog box appears.
2. Select either AND or OR from the **Logic Type** drop-down list and click **OK**. The specified operation is inserted as the Root operation.

To select the Root operation or any inserted operation

- Click on it. The insertion cursor appears to the right of the operation icon.

Note: Selecting an operation is different from selecting a condition. When you select a condition, the condition statement is highlighted. When you select an operation, an insertion cursor appears.

To insert a condition

1. Click **Add Condition**. The **Add Condition** dialog box appears.
2. Click on the **Available Fields** drop-down arrow and select the field that you want.
3. Click on the **Binary Operation** drop-down arrow and select the operation that you want.
4. Select the tab that you want to use to specify the right operand.
 - To specify a fixed value, click the **Value** tab. The **Value** input line appears. Type a fixed value.
 - To specify a form control, click the **Control** tab. The **Form controls** input line appears. Click the drop-down arrow and select the control that you want to use.
 - To specify a database field value, click the **Join** tab. The **Field to join by** input line appears. Click the drop-down arrow and select the database field that you want to use.
5. When done, click **OK**. The specified condition statement appears as a subordinate statement under the selected operation. You may add as many condition statements to an operation as necessary.

Inserting Additional Operations

To insert additional operations, you must decide (1) if you want to insert an operation at the same hierarchy *or* (2) if you want to insert an operation at the *same* hierarchical level and subordinate the selected operation to the operation that you are about to insert.

All operations are inserted *end* of the logical operations tree (or segment) *after* the last operation *at the same hierarchy level*. This means that:

- If you want to insert an operation at the *same* hierarchical level, the inserted operation will be *after* the last operation *at the same hierarchy level*, regardless of which operations *at that hierarchy level* you select.

Example: If three operations, at the same hierarchy level are listed, then any one of them could be selected and the result would be the same.



- If you insert an operation at the same hierarchy and subordinate the selected operation to the operation that you are about to insert, then the selected operation and *all of its subordinated operations and conditions* are positioned at the *end* of the logical operations tree (or segment) *after* the last operation *at the selected hierarchy level*.

Example: If three operations, at the same hierarchy level are listed, and the top one contains subordinate conditions, then if the topmost operation is selected that operation and its subordinate conditions will move to the bottom of the listing.

You can reposition an inserted operation and *all of its subordinated operations and conditions* at any time by clicking and dragging it to its new position.

To insert an operation at the same hierarchical level

1. Select an operation that is at the *same* hierarchical level as the one you want to add. The insert cursor appears on the right.
2. Click **Add Logic**. The **Choose Logic Operation** dialog box appears.
3. Select either AND or OR from the **Logic Type** drop-down list and click **OK**. The specified operation is inserted *after* the last operation *at the selected hierarchy level*.

Note: The insertion cursor moves to indicate where the new operation has been inserted.

4. Click and drag the inserted operation to the location where you want it.

To insert an operation at the same hierarchical level *and* subordinate the selected operation to one being inserted

1. Select an operation that you wish to make subordinate. The insert cursor appears on the right.
2. Click **Insert Logic**. The **Choose Logic Operation** dialog box appears.
3. Select either AND or OR from the **Logic Type** drop-down list and click **OK**. The specified operation is inserted and the selected operation and *all of its subordinated operations and conditions* are moved to the *end* of the logical operations tree (or segment) *after* the last operation *at the selected hierarchy level*.

Note: The insertion cursor moves to indicate where the new operation has been inserted.

4. Click and drag the inserted operation and *all of its subordinated operations and conditions* to the location where you want it.

Send Mail Action

A **Send Mail** action lets you send email from a Dynamic Page. Once the **Send Mail** icon is inserted into the Dynamic Page action listing. Double-click on it. The **Send Mail Properties** dialog box appears. Use this dialog box to enter **From**, **To**, **Cc**, and **Subject**. You may also send an email directly to a SMTP server address.

Note: To use Send Mail, you must ensure that a special ActiveX component is available on your Internet server. Contact your Internet provider or system administrator and provide them with required ActiveX component from your installation CD (the location is \Server Components\Send Mail\ folder).



You can also specify your email text by clicking on the Page Editor window. Email text is not displayed to your Web site visitor, only to the recipient of the email. In the body of the mail to be sent, you can design an HTML fragment like any HTML page, including images, tables, and Dynamic Text Elements. Keep in mind that the Send Mail action is a server-side action that is invisible to the Web site visitor.

To specify a Send Mail action

1. Click the **Root** icon to select it.
2. Right-click and select **Add Into**. A fly-out menu appears.
3. Select **Send Mail**. The **Send Mail** icon is inserted at the first hierarchy level below the **Root** icon.
4. Double-click the **Send Mail** icon to open the **Send Mail Properties** dialog box.
5. Decide how you want to assign a field value. Click on the tab that you want to use. Your choices include Value, Variables, Parameters, Query Element, and Cookie.
 - Click on the drop-down arrow and select a dynamically created value, or
 - Type a value directly.

Note: The only required field is TO.

6. When done, decide what information you want to include with the email text.

To specify the email text to be sent

1. With the **Send Mail** icon selected, click inside the Page Editor window. A blinking insertion cursor appears.
2. Decide what kind of text elements you want to insert
 - To insert standard HTML elements, use the Insert Object toolbar to create your Plain Page design.
 - To insert one or more dynamic elements, click the **Dynamic Element Insertion** button on the Dynamic Page toolbar. The **Dynamic Text Element** dialog box appears. Position the cursor in the Page Editor window where you want the Dynamic Element to appear and click. A blinking insertion cursor appears. Using the **Dynamic Text Element** dialog box, double-click on the name of the Dynamic Element. A Dynamic Element is inserted.

Note: When inserted, Dynamic Elements appear as greyed out text elements.

4. When done, close the **Dynamic Text Element** dialog box and click **OK**. The Send Mail action is specified.

Feedback Section Action

The **Feedback Section** action lets you display a feedback page to the Web site visitor based upon conditions that you specify. This is probably the most important Dynamic Page action that you will use. You can specify as many Feedback Sections for the dynamic page as you like. When you insert a Feedback Section, you must specify a Feedback Section name. Each Feedback Section name must be unique to the Dynamic Page; however, all Feedback Sections



associated with a Dynamic Page are assigned the same FCO.

Remember! A Dynamic Page (or an ASP resource file) is a single resource. Feedback Sections that are inserted onto a Dynamic Page are part of Dynamic Page resource file. Feedback Sections are not stand alone FCO resources.

To specify a Feedback Section action

1. Click the **Root** icon to select it.
2. Right-click and select **Add Into**. A fly-out menu appears.
3. Select **Feedback Section**. The **Feedback Section** icon is inserted at the first hierarchy level below the **Root** icon.
4. Double-click the **Feedback Section** icon to open the **Feedback Section** dialog box.
5. Type the Feedback Section name and click **OK**. The Dynamic Page Manger icon is updated.
6. Decide what information you want to display to the Web site visitor.

To specify the text to be displayed

1. With the **Feedback Section** icon selected, click inside the Page Editor window. A blinking insertion cursor appears.
2. Decide what kind of text elements you want to insert.
 - To insert standard HTML elements, use the Insert Object toolbar to create your Plain Page design.
 - To insert one or more dynamic elements, click the **Dynamic Element Insertion** button on the Dynamic Page toolbar. The **Dynamic Text Element** dialog box appears. Position the cursor in the Page Editor window where you want the Dynamic Element to appear and click. A blinking insertion cursor appears. Using the **Dynamic Text Element** dialog box, double-click on the name of the Dynamic Element. A Dynamic Element is inserted.

Note: When inserted, Dynamic Elements appear as greyed out text elements.

4. When done, close the **Dynamic Text Element** dialog box. The Feedback Section action is specified.

Request Action

The **Request** action lets you retrieve and store data in a Dynamic Page so that it can be used in some way. This action is different from a query because data can be obtained from the client or server machine rather than from a database. When you specify a variable name, it is used to identify the retrieved value. When you specify a request action, you are assigning the variable name a value; however, you are also specifying both the source of the data and the means required to retrieve the data.



Request from User Dialog Box

This dialog box has two parts: a variable key that initiates the request and a key result which stores the data once it is obtained. When making request, first identify the type of information that you want. Your request options are identified by the dialog box tabs that appear on the right. The data types that you can request include form data, cookie data, query string data, server variables, and others. By selecting a data type, you are specifying both the data source and the means required to complete the request. WebShop automatically creates the necessary ASP scripting to complete the request and obtain the data.

Next specify the variable name. The variables names from which you select are defined using the **Property Sheet for the Dynamic Page**. Before you can request data, you must set up the variables that will hold the data obtained by the request. The variable names that you assign must agree with the data names used by the data source. *For example:* A form field defined as **First_Name**, must be defined as **First_Name** on the Dynamic Page before a request can be initiated.

Note: Variable names that do not agree will generate a server error. If you are requesting the following data Client Certificate, Server Variables, Application Element, or Session Element, then you must set up a variable that stores the appropriate ASP element identification and use this variable as the variable key that initiates the request.

To specify a Request action

1. Click the **Root** icon to select it.
2. Right-click and select **Add Into**. A fly-out menu appears.
3. Select **Request**. The **Request** icon is inserted at the first hierarchy level below the **Root** icon.
4. Double-click the **Request** icon. The **Request from User** dialog box appears.
5. Click the drop-down arrow below the label **Variable to store the result of request**. The currently defined variables for the Dynamic Page appear.
6. Select the variable that you want.
7. Click on the tab which identifies the type of information that you want to assign to the variable. Your choices are: Form, Query String, Cookie, Client Certificate, Server Variables, Application Element, or Session Element.
8. Click the drop-down arrow below the label **Variable to store the key of Request**. The currently defined variables for the Dynamic Page appear.
9. Select the same variable as in Step 6.

Note: The variable value will be replaced with the retrieved value only while the Dynamic page is executing. Retrieved values are not permanently saved. The initial values that you specify are restored after the processing is complete.

10. When done, click **OK**. The request is specified.

Normally, you will want to initiate a request for information prior to completing any actions on the data. Many users initiate requests at the start of the top of the Dynamic Page action listing.



Redirect to Action

The **Redirect to** action lets you re-direct the viewer's browser to different page. There are two options when specifying a Redirect to action: **Local** or **External**. A local redirect takes the viewer to different page within the current Web site. An external redirect takes the viewer to different page outside the current Web site. You can also pass parameter values from the current Dynamic Page to the Redirect page. To pass values, you must specify the parameter name and value. The value that you specify can be entered directly or can be a dynamically generated. When specifying a dynamically-generated values, choose between Parameters, Query Elements, Cookies, or Variables.

Property Sheet for Redirect Action Dialog Box

This dialog box has two tabs: **Local** and **External**.

Local Tab

The **Local** tab lets you select a page within the current Web site and specify the parameter names and values to pass when the redirect action is performed. The **Site Pages** combo box displays the available Web pages for the current project. The **Parameters List** displays the available parameters for the selected page. The **Chosen parameters and values** display area lists the parameters and values that will be passed when the redirect action is performed.

To specify a local redirect

1. Click the **Local** tab.
2. Click the drop-down arrow next to **Site Pages**. A list of available Web pages appears.
3. Select the page where the viewer is to be taken.
4. Click the drop-down arrow next to **Parameters List**. A list of available parameters for the selected page appears.
5. Decide if you want to pass specific parameter values.
 - To add a parameter to the **Chosen parameters and values** display area, select the parameter from the **Parameters List** of the new page, and click the **Add** button.
 - To set a value to pass when the redirect action is performed, select the parameter by clicking on it, and click the **Modify** button. The **Modify** dialog box appears. Decide how you want to assign a value.
 - To assign a value directly, click the **Value** tab and type the value to be assigned.
 - To assign a dynamically, select any one of the other tabs and specify the data name that contains the value that you want to assign.
 - When done, click **OK**. The parameter value is assigned and appears in the **Chosen parameters and values** display area.
6. When done, click **OK**. The **Redirect to** action is specified.

Note: Each tab provides a list of available data names. These names are the currently defined data names associated with the current Dynamic Page. To assign a dynamically-generated value, the data name must be defined on the Dynamic Page and a value must be assigned. If you assign a query element, you may only select a query output element.



External Tab

The **External** tab lets you specify an external URL location outside the current Web site and lets you specify the parameter names and values to pass when the redirect action is performed. You can only pass parameter values by directly creating parameter names to hold your values. Since the specified URL location is outside the current project site, WebShop cannot determine the currently defined parameters. For data to pass effectively, you must obtain the parameter names directly from the URL location.

To specify an external redirect

1. Click the **External** tab.
2. Click the **Custom URL** input line and type (or paste) the exact URL address where the viewer will be taken.
3. Decide if you want to pass specific parameter values.
 - To add a parameter to the **Chosen parameters and values** display area, click the **Parameter Name** input line and type (or paste) the parameter name and click the **Add** button.
 - To set a value to pass when the redirect action is performed, select the parameter by clicking on it, and click the **Modify** button. The **Modify** dialog box appears. Decide how you want to assign a value.
 - To assign a value directly, click the **Value** tab and type the value to be assigned.
 - To assign a dynamically, select any one of the other tabs and specify the data name that contains the value that you want to assign.
 - When done, click **OK**. The parameter value is assigned and appears in the **Chosen parameters and values** display area.

Note: Each tab provides a list of available data names. These names are the currently defined data names associated with the current Dynamic Page. To assign a dynamically-generated value, the data name must be defined on the Dynamic Page and a value must be assigned. If you assign a query element, you may only select a query output element.

7. When done, click **OK**. The **Redirect to** action is specified.

Condition Action

The **Condition** action lets you establish two processing alternatives based upon a true or false condition. When you insert this action, three icons appear:

- **“IF” statement icon.** Use this icon to evaluate a condition against an expected value. When first inserted, the words “Empty Condition” appear to the right of the icon. These are replaced once the evaluation condition is specified.
- **“OK” statement icon.** Use this icon to initiate an action when the “IF” statement is true. The word “Then” is always appears to the right of the icon.
- **“NO” statement icon.** Use this icon initiate an action when the “IF” statement is false. The word “Else” is always appears to the right of the icon.

When referring to the icons and labels created by a Condition action, “OK” and “NO” refer to



the true or false condition; “Then” and “Else” refer to the consequential actions.

To initiate an action for use with either “OK” or “NO,” use **Dynamic Page > Add Into** or right-click and select **Add Into**. Actions that are added into either “OK” or “NO” are subordinated actions. You do not *have to* specify actions for either “OK” or “NO” if you don’t want to. You can arrange your condition statement to only cause an action to be performed for only one of the two logical alternatives. If either the “OK” or “NO” statement is left unspecified, the next action performed is whatever follows the initiating “IF” statement.

Remember! Action icons are displayed hierarchically. That means, actions at the same hierarchy are performed sequentially from top to bottom in the order presented. Any subordinated action is performed only when the initiating action is performed. If a subordinating statement (such as “OK” or “NO”) is left unspecified, then the next initiating action is performed. If you wish to make multiple evaluations that specify actions for only true conditions, you can “nest” Condition actions within one another. The last false condition in the “nested” series, if left unspecified, causes the next initiating action to be performed. You can chain together “one-sided” condition statements if you want.

For example: A Condition action can be arranged to see if a cookie is written in the visitor’s computer (“IF” condition). If not, it will display some welcome text and sends an advertizing email (“NO” statement). If the cookie is available, the page reads an information stored in the cookie, and displays another Feedback Section (“OK” statement).

Set Condition Dialog Box

This dialog box has three parts: left operand, operation and right operand. To specify the “If” condition, you must construct a statement for which there can only be a true or a false result.

Left Operand: Select the type of value that you want to enter by clicking on the appropriate tab. Your choices are: Variable, Parameter, Query, Cookie. Each tab contains a single input line. Click the drop-down arrow. A listing of available data names appears. Select the data name that contains the value that you want to test.

Operation: Select one operation from among the following using the drop-down list: (1) BEGINS WITH, (2) CONTAINS, (3) ENDS, (4) EQUAL, (5) NOT EQUAL, (6) GREATER THAN, (7) GREATER OR EQUAL TO, and (8) LESS THAN, (9) LESS OR EQUAL TO

Right Operand: Select the type of value that you want to enter by clicking on the appropriate tab. Your choices are: Value, Variable, Parameter, Query, Cookie. Each tab contains a single input line. Click the drop-down arrow. A listing of available data names appears. Select the data name that contains the value that you want to test.

Note: The data names that appear have been defined for used with the current Dynamic Page either through the Dynamic Page Property Sheet or a Query action.

To specify a Condition action

1. Click the **Root** icon to select it.
2. Right-click and select **Add Into**. A fly-out menu appears.
3. Select **Condition**. The **Condition** icons IF-OK-NO are inserted at the first hierarchy level below the **Root** icon.
4. Double-click the “**IF**” icon. The **Set Condition** dialog box appears.



5. Construct the “**IF**” condition that you want to evaluate.
 - For the left operand, select the type of value that you want to enter by clicking on the appropriate tab. Your choices are: Value, Variable, Parameter, Query, Cookie. Click the drop-down arrow and select a data name.
 - For the operation, click the drop-down arrow and select an operation.
 - For the right operand, select the type of value that you want to enter by clicking on the appropriate tab. Your choices are: Value, Variable, Parameter, Query, Cookie. Click the drop-down arrow and select the data name.

Note: If you leave the right operand unspecified, or blank, the “IF” condition tests for an empty or null value.

- When done, click **OK**. The condition statement appears to the right of the “**IF**” icon.
6. Click the “**OK**” icon to select it and right-click. Use **Add Into** to select the action you want to initiate when the “**IF**” condition is true. An action icon appears. Complete your action specifications, as necessary.
 7. Click the “**NO**” icon to select it and right-click. Use **Add Into** to select the action you want to initiate when the “**IF**” condition is false. An action icon appears. Complete your action specifications, as necessary.

Be sure to test your Dynamic Page evaluations using some sample data. It is important to verify that each part of the IF-OK-NO Condition action are operating as expected.

Statement Action

The **Statement** action lets you assign a value to a variable while a Dynamic Page is completing its actions. To assign a value, you must specify an operation that will construct the variable’s new value given the variable’s current value.

Set Variable To Dialog Box

This dialog box has three parts: left operand, operation and right operand. To construct a statement, specify the variable that is to receive the new value as the left operand, select an operation, and specify the variable or value that will be used to construct the new value as the right operand.

Left Operand: Click the **Choose Variable Name** drop-down arrow; select a variable from the list of available variables. The variables that appear were previously defined using the Dynamic Page Property Sheet.

Operation: Click the drop-down arrow; select one operation from among the following: Equals, Subtract, Concatenate, Add, Multiply, Divide. If you use the operation “Concatenate,” the string values for the left and right operands will be joined. Use this operation to construct dynamic text that will display as a single variable rather than multiple variables.

Right Operand: Select the type of value that you want to enter by clicking on the appropriate tab. Your choices are Value, Variable, Parameter, Query, Cookie. Each tab contains a single input line. Click the drop-down arrow. A listing of available data names appears. Select the data name that you want to use.



Note: The data names that appear have been defined for use with the current Dynamic Page either through the Dynamic Page Property Sheet or a Query action.

To specify a Statement action

1. Click the **Root** icon to select it.
2. Right-click and select **Add Info**. A fly-out menu appears.
3. Select **Statement**. The **Statement** icon is inserted at the first hierarchy level below the **Root** icon.
4. Double-click the **Statement** icon. The **Set Variable To** dialog box appears.
5. Construct the equation that will assign the new value.
 - For the left operand, click the **Choose Variable Name** drop-down arrow and select the variable that you want to receive the new value.
 - For the operation, click the drop-down arrow and select an operation to use.
 - For the right operand, select the type of value that you want to enter by clicking on the appropriate tab. Your choices are Value, Variable, Parameter, Query, Cookie. Each tab contains a single input line. Click the drop-down arrow and select the data name.

Note: To assign a variable an empty or null value, use Equal and leave the left operand blank.

5. When done, click **OK**. The variable statement appears to the right of the **Statement** icon.

Cookie Action

The **Cookie** action lets you assign a value to a variable and then write that value to the visitor's PC. To assign a value, you must specify both the cookie key that will store the temporary value and the variable that contains the value that you wish to write onto the visitor's PC. One of the purposes associated with cookies is to identify the previous actions that a visitor has taken. To identify the user and the previous actions taken, cookies keys are frequently organized by page so that there is a unique cookie associated with individual pages and with specific actions. A visitor's PC may contain several cookies. Each one identifies information specific to a page and specific to the users actions. To accurately identify where the visitor has been, a cookie must be written before a Dynamic Page page displays any feedback. That way, a running history can be kept which is always accessible.

Write Cookie Dialog Box

This dialog box has three parts: left operand, operation and right operand. To write a cookie to a visitor's PC, specify the cookie key that will receive the new value prior to being written to the visitor's PC, then specify the variable (or value) that will be assigned to the cookie key.

Left Operand: Click the drop-down arrow next to the **Choose Key** field; select a cookie key from the list of available keys. The cookie keys that appear were previously defined using the Dynamic Page Property Sheet.

Operation: When writing a cookie, the Operation is always Equal.

Right Operand: Click the **Choose Variable** field with drop-down list box; select one of the



variables that were previously defined using the Dynamic Page Property Sheet.

To specify a Cookie action

1. Click the **Root** icon to select it.
2. Right-click and select **Add Info**. A fly-out menu appears.
3. Select **Cookie**. The **Cookie** icon is inserted at the first hierarchy level below the **Root** icon.
Note: If a Feedback Section has already been inserted, an error message appears. Be sure that the Cookie action is inserted prior to any Feedback Section.
4. Double-click the **Cookie** icon. The **Write Cookie** dialog box appears.
5. Construct the equation that will assign the cookie value.
 - For the left operand, click the **Choose Key** drop-down arrow and select the cookie key that you want to receive the new value.
 - For the right operand, click the **Choose Variable** drop-down arrow and select the variables that you want to use to assign the cookie value.
6. When done, click **OK**. The cookie assignment statement appears to the right of the **Cookie** icon.

Chapter 2.

Sending Emails from a Form

Introduction

With the use of Active Server Pages (ASP) and an Active-X component, you can send the contents of a form to an email address.

The ASP technology was developed by Microsoft and requires that the pages be used with a Windows NT-server running Internet Information Server (IIS), Windows NT Workstation running Peer Web Server, or, for Windows 95, a PC running Personal Web Server. Since sending email through conventional programming causes problems when connecting to the SMTP server (through the use of CGI scripts), Boomerang Software has created an Active-X component that will contact the mail server and send the mail through the server.

Registering WebShop's Active-X

An Active-X component is a *.DLL file that resides in the Web-server's System folder. To successfully send an automatic email from your Web site, a specially developed Active-X component must be registered on your hosting company's Windows NT-server.

To do this, contact your Web-hosting/ISP provider and ask them to register the Boomerang Software Active-X component. You will need to send them the contents of the folder located on the WebShop program CD **X:\Server Components\Send Mail** (where X designates your CD ROM drive).

The **Send Mail** folder contains the following files:

- **MailX.dll**
- **Readme.txt**
- **Reg.bat**
- **Unreg.bat**

Once you have confirmed the contents of this folder, you must send it to your Web Site/ISP provider.

To register the Active-X

1. Copy the **Send Mail** folder and place it on your hard disk.



- Send your Web-hosting/ISP provider the folder by email and ask them to run the batch file **Reg.bat** or upload your site (including the above files) and ask them to register the Active-X component.

You will want to be sure that your Web-host administrator reads the Readme text file before attempting to register the Boomerang Active-X component. Your Web-hosting/ISP provider can unregister the Active-X component by running the batch file **Unreg.bat**. The Active-X component has been fully tested on Windows NT-servers Running IIS. If there are any questions, from the Web-hosting/ISP provider, regarding the registration of the Active-X component, have them call the Boomerang Software Technical Support Department at (617) 484-4664 or email Boomerang Software, Inc. at either of the following email addresses:

- techsupport@boomerangsoftware.com
- webhostingsupport@boomerangsoftware.com

Note: The Active-X program and associated program and files provided on the CD are properties of Boomerang Software, Inc. and are copyrighted. You and your ISP or Web-hosting company have the right to use them free of charge, with proper acknowledgment given to Boomerang Software. They cannot be sold or transferred to other entities without written permission from Boomerang Software.

Creating Your Email Function Using Dynamic Pages

First, create a new project file. Select **File > New** and select the template with an **X** in the **Preview** area.

Dynamic Page Icons

To create dynamic email function, the following Dynamic Page icons will be visible in the Dynamic Page Manager window. These icons identify dynamic operations that you will use to complete the project.

- **Root Icon.** This is the Dynamic Page root directory. All supporting operation resources are created below this icon.
- **“IF” Statement Icon.** This is an evaluation operation. Use it to evaluate a condition against an expected value.
- **“OK” Statement Icon.** This is an action operation. Use it to initiate an action when the "IF" Statement is true.
- **“NO” Statement Icon.** This is an action operation. Use it to initiate an action when the "IF" Statement is false.
- **Statement Icon.** This is an action operation. Use it to assign a specific value to a parameter or a variable.
- **Feedback Section Icon.** This is a notification operation. Use it to display text and/or dynamically generated values to the Web site visitor.
- **Send Mail icon.** This is a notification operation. Use it to send text and/or dynamically generated values to the Web site visitor or someone other than the Web site visitor.

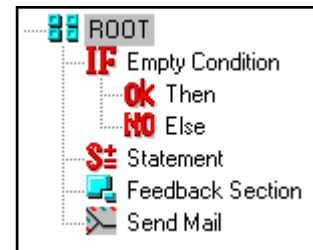


Figure 2.1 – Dynamic Page Icons



Creating a Dynamic Page

Dynamic Pages can be created either from the **Site Manager–Physical Layout** window or the **Site Manager–Logical Layout** window. It is recommended you create your pages from the Logical Layout and then customize the file structure in the Physical Layout.

To create a Dynamic Page from Physical Layout

1. Highlight the **Root** directory (or subfolder) where you want to place the Dynamic Page and right-click. A pop-up menu appears.
2. Select **Create Dynamic Page**.

Note: Pages created in the Physical Layout are not visible in the Logical Layout. To make a page visible in Logical Layout, you must create them in Logical Layout.

To create a Dynamic Page from Logical Layout

1. Highlight the **Logical Page** icon that you want the new Dynamic Page to be under and right-click. A pop-up menu appears.
2. Select **New**. The **Add Logical Page** dialog box appears.
3. Select **Dynamic Page** and click **OK**.

Note: Pages created in Logical Layout are by default placed in the CGI Bin folder. To verify this, go to Site Manager–Physical Layout and open the CGI Bin folder.

From either the Logical or Physical Layout, double-click on the **Dynamic Page** icon you have created. This opens the Page Editor window for that page. The default physical names for Dynamic Pages are DynPage1.asp, DynPage2.asp and so forth. The default logical names for Dynamic Pages are DynPage1, DynPage2 and so forth.



Creating a Form for Use with a Dynamic Page

To enable a site visitor to send an email to a specific address, you must first create a form that collects the necessary information. When creating a form for use with a Dynamic Page, you must first create the form on some page *other than* a Dynamic Page. Once you have created this resource, you must link the form to the Dynamic Page where the information is collected. Create a form with the following fields:

- **First_Name**
- **Last_Name**
- **Company**
- **Address**
- **City**
- **State**
- **Zip**
- **Email**



To create a form on any page (other than a Dynamic Page)

1. In the **Site Manager–Physical Layout** window, double-click on the page that you want. This opens the Page Editor window for that page.
-  2. Click **Show/Hide** on the Standard toolbar. Formatting symbols such as tab and paragraph symbols are visible.
-  3. Select **Insert > Form** or click **Form** on the Insert Object toolbar. Two lines designating the form area appear.
4. Place the cursor inside the two lines and type “First Name:”
5. Select **Insert > Form Fields > Text Line**. The **Text Line Properties** dialog box appears.
6. In the **Name** box, type “First_Name”
7. Specify the **Field Size** and the **Text Max** length properties as necessary and click **OK**.
8. Press **ENTER** to move down to the next line on the form.
9. Repeat Steps 4–8 for every field you need on your form.

After you have completed the above, add a Submit button.

To add a submit button

1. Selecting **Insert > Form Field > Push Button**. The **Push Button Properties** dialog box appears.

Note: The defaults are set to create a Submit push button.

2. Click **OK**. The **Submit** button appears on your form.

Next, link the form to the Dynamic Page you created.

To link the form to a Dynamic Page

1. Right-click somewhere in the form (the white space). A pop-up menu appears.
2. Select **Forms**. The **Form Properties** dialog box appears.
3. In the **Action** section, click **Local** and select physical name for the Dynamic Page that you want to associate this form with. For example, “DynPage1.asp.”
4. When done, click **OK**.

Creating a Dynamic Page with Send Mail Capability

To create your Dynamic Page elements, you must define the parameters associated with each form field and define a variable that tests whether a field was completed.

Defining Parameters on a Dynamic Page

You should define parameters for each of the form fields that you created (except for the Submit button).

At this point, step back and think about which fields are going to be required so you can leave the default values for these parameters empty. This is how to determine whether the user has not entered a value.



This means that for a required form field there should be **NO** default values. For this example you will require all fields except for “Company.”

To define parameters



1. Double-click the **ROOT** icon. The **Property Sheet for the Dynamic Page** appears.
2. On the **Input Parameter** tab, type in the **New Parameter Box**, the field names from the form. For example, type “First_Name” and click **Add**. Do not modify this parameter or specify a value.
3. Repeat this for the rest of the fields.

Note: For “Company,” modify the parameter and specify the value “Not Entered.”

4. To do this, highlight the parameter, “**Company**,” and click **Modify**. Specify the value “**Not Entered**” and click **OK**.

Defining Variables on a Dynamic Page

As you are requiring certain fields to be filled out, you will need to test for errors at the time of submission. To do this, create a variable called “var_any_error.” You will later use this variable to concatenate (join) with the values of the missing fields.



1. Double-click the **ROOT** icon and select the **Variables** tab.
2. Type in the **New Variable Box**, “var_any_error” and click **Add**.
3. Click **OK**.

You will not assign the variable any value. Instead, concatenate it with the error message later.

Creating a Field Validation Test

In this section, create a test to make sure that the required fields were filled out. If a field is not entered, assign the field parameter a name and the variable, “var_any_error,” a value.



1. Highlight the **ROOT** icon and right-click and select **Add Into > Condition**. The **Condition** icons IF-OK-NO are inserted.



2. Double-click the “**IF**” statement icon. The **Set Conditions** dialog box appears.
3. Select the **Parameters** tab
4. For the left operand, select “**First_Name**” from the drop-down box.
5. For the operation, select “**EQUAL**.”
6. For the right operand, select the **Value** tab and enter nothing (leave it blank).
7. Click **OK**.

Note: What this says, is that if the parameter, “First_Name,” has no value when submitted then note this in the variable, “var_any_error.”



8. Highlight the “**OK**” statement icon for that condition, right-click, and select **Add Into > Statement**.



9. Double-click the **Statement** icon. The **Set Variable To** dialog box appears.



10. In the **Choose Variable Name**, select “**var_any_error.**”
11. In the operation drop-down box, select “**Concatenate.**” Concatenate means “to join to.”
12. In the right operand, select the **Value** tab and in the **Insert Value** box, type “First Name.”
13. Click **OK.**
14. Repeat this procedure for all required fields.

Creating a Feedback page and a Send Mail function.

You will now create the Feedback pages for an Error Message and for a Successful Entry.



1. Highlight the **ROOT** icon and right-click and select **Add Into > Condition.**



2. Double-click the “**IF**” statement icon. The **Set Condition** dialog box appears.
3. For the left operand, select the **Variable** tab and select the variable, “**var_any_error.**”
4. For the operation, select “**NOT EQUAL.**”
5. For the right operand, select the **Value** tab and leave the value blank.

This says that “**IF**” ‘var_any_error’ contains any information (from the validation) then (“**OK**”) produces an error message. **IF** the condition is not true (“**NO**” or “**ELSE**”) then assume that all the fields were filled and produce a Success page notice.

Creating an Error Page



1. Highlight the “**OK**” statement icon, right-click and select **Add Into > Feedback Section.**



2. Highlight the **Feedback Section** icon and double-click. Rename it to “Error.”
3. Click **OK.**
4. Place the cursor on the right screen and type “You have not entered the following fields:” Press **ENTER.**
5. Click on the **Dynamic Element Insertion** icon. The **Dynamic Text Element** dialog box appears.
6. Highlight “**var_any_error**” and double-click.
7. Close the dialog box.

Creating a Success Page



1. For the condition you specified above, click the “**NO**” statement icon, right-click, and select **Add Into > Feedback Section.**



2. Highlight the **Feedback Section** icon and double-click. Rename it to “Success.”
3. Click **OK.**
4. Place the cursor on the right screen and type “Thank you for filling out our form. Your message has been sent.”

Setting Up a Send Mail Operation



1. For the Success condition specified above, click the “**NO**” statement icon, right-click and select **Add Into > Send Mail.**



2. Double-click on the **Send Mail** icon. The **Send Mail** dialog box appears.

Note: You will now need to fill out the necessary dialog information. The two critical elements that need to be filled out every time are the SMTP Server and the TO: box. In this case, do a few more.

3. For the “**From:**” box, click the **Parameter** tab and select **Email** from the drop down box. This will place the data from the form field **Email** to the **From Section in the Header** of the email you receive.
4. The “**To:**” box should be your own email address. You can also have the form field have a drop-down box with a few email addresses and you can call them from the **Parameter** tab.
5. In the “**Subject:**” box, type whatever you want to be displayed in the **Subject Box** of your email.

Note: You could have the user type in the subject and have it dynamically posted there.

6. For the SMTP server, type in the name of your mail server. Usually this will look something like this “mail.yourservername.com” or sometimes it may just be an IP Address.
7. Click **OK**.

Creating Dynamic Text to Display as Email Text



1. Highlight the **Send Mail** icon and place the cursor on the screen on the right. This is the body of the email.
2. Type “First Name:” and click the **Dynamic Element Insertion** icon. Select **Parameter(“First_Name”)**.
3. Repeat the procedure for the remaining fields of the form.

You are now ready to upload your site and test it.

Chapter 3.

Adding Form Input into a Database

Introduction

Many Web designers want to take data from a form and insert it into a database. To do this in WebShop, you must complete a multi-step process—one that makes use of Dynamic Pages. To achieve database connectivity, you must have a database. WebShop allows you to import a database (Microsoft Access .MDB file) into your Web site project so that data can be queried, added or modified. As the functions and methodologies are complex, the use of these functions should be for advanced users. The following is a simple example of what can be done with the program.

Note: You are using the Active Server Pages (ASP) technology created by Microsoft. This technology currently operates on Windows NT Servers running Microsoft Internet Information Servers (IIS). Windows NT Workstation running Peer Web Server or for Windows 95 running Personal Web Server.

In this example you will create a form on a page and have the information entered placed into the database. You should create this project on a test Web site before incorporating it into your Web site project.

The first thing you should do is to construct a new *.MDB database.

Create a Database

You must create an Access database with a table called **Address**. In this table, insert the following fields:

- **First_Name**
- **Last_Name**
- **Company**
- **Address**
- **City**
- **State**
- **Zip**

Note: It is important that you set any fields that use numbers to a text format.



Updating Your Database Using Dynamic Pages

Now that you have created your database, you will want to create a new project file. Select **File > New** and select the template with an **X** in the **Preview** area.

Dynamic Page Icons

To complete your database project, the following Dynamic Page icons will be visible in the Dynamic Page Manager window. These icons identify dynamic operations that you will use to update your database from a Dynamic Page.

- **Root Icon.** This is the Dynamic Page root directory. All supporting operation resources are created below this icon.
- **“IF” Statement Icon.** This is an evaluation operation. Use it to evaluate a condition against an expected value.
- **“OK” Statement Icon.** This is an action operation. Use it to initiate an action when the "IF" Statement is true.
- **“NO” Statement Icon.** This is an action operation. Use it to initiate an action when the "IF" Statement is false.
- **Statement Icon.** This is an action operation. Use it to assign a specific value to a parameter or a variable.
- **Feedback Section Icon.** This is a notification operation. Use it to display text or dynamically generated values to the Web site visitor.
- **Query Icon.** This is a update or access operation. Use it to send data to or read data from a database.

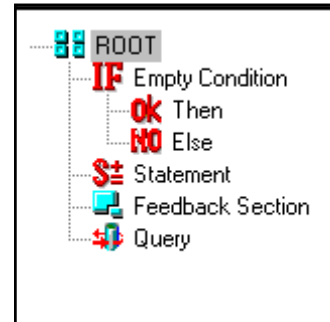


Figure 3.1 – Dynamic Page Icons

Creating a Dynamic Page

Dynamic Pages can be created from either the **Site Manager–Physical Layout** window or the **Site Manager–Logical Layout** window. It is recommended you create your pages from the Logical Layout and then customize the file structure in the Physical Layout.

To create a Dynamic Page from Physical Layout

1. Highlight the **Root** directory (or subfolder) where you want to place the Dynamic Page and right-click. A pop-up menu appears.
2. Select **Create Dynamic Page**.

OR

1. Highlight the **Root** directory (or subfolder) where you want to place the Dynamic Page and click **Dynamic Page** on the Physical Layout toolbar or select **Physical Layout > Create Dynamic Page**.

Note: Pages created in Physical Layout are not visible in Logical Layout. To make a page visible in Logical Layout, you must create them in Logical Layout.



To create a Dynamic Page from Logical Layout

1. Highlight the page that you want the new Dynamic Page to be under and right-click. A pop-up menu appears.
2. Select **New**. The **Add Logical Page** dialog box appears.
3. Select **Dynamic Page** and click **OK**.

OR

1. Highlight the page that you want the new Dynamic Page to be under and click **New Page** on the Logical Layout toolbar. The **Add Logical Page** dialog box appears.
2. Select **Dynamic Page** and click **OK**.

Note: Dynamic Pages created in the Logical Layout are by default placed in the CGI Bin folder. To verify this, go to Site Manager–Physical Layout and open the CGI bin folder.

From either Logical or Physical Layout, double-click on the **Dynamic Page** icon you have created. This opens the Page Editor window for that page. The default physical names for Dynamic Pages are DynPage1.asp, DynPage2.asp and so forth. The default logical names for Dynamic Pages are DynPage1, DynPage2 and so on.

Creating a Form for Use with a Dynamic Page

To collect information, you must create a form. When creating a form for use with a dynamic page, you must first create the form on some page *other than a dynamic page*. Once you have created this resource, link the form to the dynamic page.

To create a form on any page (other than a Dynamic Page)

1. In the **Site Manager–Physical Layout** window, double-click on the page that you want. This opens the Page Editor window for that page
2. Click **Show/Hide** on the Standard toolbar. Formatting symbols such as tab and paragraph symbols are visible.
3. Select **Insert > Form** or click **Form** on the Insert Object toolbar. Two lines designating the form area appear.
4. Place the cursor inside the two lines and type “First Name:”
5. Select **Insert > Form Fields > Text Line**. The **Text Line Properties** dialog box appears.
6. In the **Name** box, type “First_Name”
Note: This name must be the same field name you created in your database.
7. Specify the **Field Size** and the **Text Max** length properties as necessary and click **OK**.
8. Press **ENTER** to move down to the next line on the form.
9. Repeat Steps 4–8 for every field you created in your database.

After you have completed the above, add a **Submit** button.



To add a submit button



1. Selecting **Insert > Form Field > Push Button**. The **Push Button Properties** dialog box appears.

Note: The defaults are set to create a Submit push button.

2. Click **OK**. The **Submit** button appears on your form.

Next, link the form to the dynamic page you created.

To link the form to a dynamic page

1. Right-click somewhere in the form (the white space). A pop-up menu appears.
2. Select **Forms**. The **Form Properties** dialog box appears.
3. In the **Action** section, click **Local** and select the physical name for the dynamic page that you want to associate this form with. For example, “DynPage1.asp.”
4. When done, click **OK**.

Importing the Database into the Project

You may need to import external data into your project to use with your dynamic pages. It is often more efficient to create your data outside WebShop and import it later when you are ready to test one or more of your Active Server Pages.

To import a database

1. From the **Site Manager–Physical Layout** window, right-click in the open area. A pop-up menu appears.
2. Select **Import Database**.
3. Navigate and select the database that you want to use and click **Open**. The database will appear in the Physical Layout.
4. Highlight the database you imported and right-click. A pop-up menu appears.
5. Select **Properties**. The **Properties** dialog box appears.
6. Click the **Database** tab. A diagram of the imported database’s hierarchy appears.

*Note: Opening the database tables allows you to access field-level information. The database hierarchy is **DATABASE > TABLES > FIELDS**. Each table has an icon and a label.*

7. Select the table icon that you want to open and click **Open**.
8. After a few moments, the word “**Open**” appears next the selected table icon. You have now opened the database table and can view, add or remove data entries.
9. Click **OK**.

To view, add, or remove data entries for an open data table

You can view the database from the program and also add and remove entries in the database by double-clicking the database after one or more of the data tables have been opened.

1. From the **Site Manager–Physical Layout** window, double-click the database icon you want. The **Fill Data** dialog box appears.



Note: Only data for “open” database tables is displayed. There is a separate tab for each table opened.

2. View your data by clicking on the tab for the table you want.
To move forward or backward through your data one record at a time, use the **Forward** and **Backward** navigation buttons at the top of the dialog box.
To move to the start or the end of your data records, use the **Go to Start** or **Go To End** navigation buttons at the top of the dialog box.
3. To add a record, click **+** (**plus sign**). A blank record appears. Enter your data as necessary.
4. To delete a record that currently is displayed, click **-** (**minus sign**).

Creating Dynamic Page Elements For Updating a Database

To create your dynamic page elements, you must define the parameters associated with each form field you specified on your form and define a variable that will test whether a field was completed.

Defining Parameters for a Dynamic Page

You should define parameters for each of the form fields that you created (except for the Submit Button).

At this point, step back and think about which fields are going to be required so you can leave the default values for these parameters empty. This is how to determine whether the user has not entered a value.

This means that for each required form field, there should be **NO** default values.

For our example, you will require only that the First Name and Last Name field be filled.



1. Double-click the **ROOT** icon. The **Property Sheet for the Dynamic Page** appears.
2. Click the **Input Parameter** tab, type each of the field names in the **New Parameter** box. For example, type “First_Name” and click **Add**. Do **not** modify this parameter or enter a value.

Note: For all other fields, modify the parameter and enter the value “Not Entered.”

To do this, highlight the parameter, in this case, “Address” and click **Modify**. Type the value “**Not Entered**” and click **OK**. Repeat this procedure for the rest of the parameters.

Defining Variables for a Dynamic Page

As you are going to have more than one required field, you must create an error message. You also must determine the field(s) that were not entered.

To do this, create a variable called “**var_any_error**.” Later, use this variable to join with values of the missing fields. To do this,



1. Double-click the **ROOT** icon and select the **Variables** tab.
2. Type the variable name “var_any_error” into the **New Variable** box.
3. Click **Add** and **OK**.

Do not assign the variable a value. Later, you will concatenate it with the error message.

Creating a Field Validation Test

To insure that users enter data correctly, set up some field validations. These validations must be set up for each of the required fields.

To set up a field validation



1. Highlight the **ROOT** icon and right-click. A pop-up menu appears.
2. Select **Add Into > Condition**. The **Condition** icons IF–OK–NO appear.



3. Double-click the “**IF**” condition icon. The **Set Conditions** dialog box appears.

Note: The Set Condition dialog box has three parts, a left operand, a center operation, and a right operand.

4. Select the **Parameters** tab, on the left operand, and select “First_Name” from the drop-down box. For the operation select “**EQUAL.**” For the right operand select the **Value** tab and leave it blank.
5. Click **OK**.

Note: What is being stated is that “if the parameter, “First_Name,” has no value (left blank) when submitted, then note this in the variable “var_any_error.” Assign the variable, “var_any_error,” a value, if the above statement is true.



6. Highlight the “**OK**” icon for that condition, right-click, and select **Add Into > Statement**.



7. Double-click the **Statement** icon. The **Set Variable To** dialog box appears.
8. In the **Choose Variable Name**, select “var_any_error.”
9. In the **Operation** drop-down box, select “**Concatenate.**” Concatenate means to join to.
10. In the right operand, select the **Value** tab and in the **Insert Value** box type “First Name, ”
11. Click **OK**.
12. Repeat both the above procedures for the other required field (that is, “**Last_Name**”), but in the **Set Variable To** dialog box set the right operand value to **Last Name**.

For the condition statement of the last required field (that is, “Last_Name”) use the “**NO**” (ELSE) part to write the information to the database.

Writing Information to the Database

At this point, you are ready to have your Active Server Page write information to the database. To do this, link together individual data fields in you database to individual form fields on your form.

**To link data fields with form fields**

1. Highlight the “**NO**” statement icon for the last required field (“Last_Name”), right-click, and select **Add Into > Query**. The **Choose Data Source for New Query** dialog box appears.



2. Highlight the **Query** icon that you want to use and, for query type, select “**Insert.**” Click **Edit Query**. The **Query Properties** dialog box appears.

3. Highlight, under the **Fields** section, “**First_Name**” and double-click. The **Set Value** dialog box appears.

4. Select the **Control** tab and assign it the appropriate value from the drop-down box (that is, “First_Name” matches First_Name).

5. Click **OK**.

6. Repeat this procedure for all the fields in the **Fields and Values** list box and click **OK**. The Page Editor window reappears.

Creating a Feedback Page

Create the feedback page to use with two messages: an error message and a successful entry message.

To create a feedback page

1. Highlight the **ROOT** icon, right-click, and select **Add Into > Condition**. The **Condition** icons IF-OK-NO appear.



2. Double-click the “**IF**” statement icon. The **Set Condition** dialog box appears.

3. For the left operand, select the **Variable** tab and select the variable, “**var_any_error.**”

4. For the operation, select “**NOT EQUAL.**”

5. For the right operand, select the **Value** tab and leave the value blank.

This states that “**IF**” ‘var_any_error’ contains any information (from the validation) then produce an error message (“**OK**”). If the condition is not true (“**NO**” or “**ELSE**”) then assume that all the fields were filled and produce a success page.

Creating an Error Page

For the condition you specified above, you must create an Error page.

To create an error page

1. Click the “**OK**” statement icon, right-click, and select **Add Into > Feedback Section**.



2. Highlight the **Feedback Section** icon and double-click. Rename it “Error” and click **OK**.

3. Place the cursor on the right screen and type “You have not entered the following Fields” and press **ENTER**.

4. Click the **Dynamic Element Insertion** icon. The **Dynamic Text Element** dialog box



appears.

5. Select and double-click “**var_any_error.**”
6. Close the dialog box.

Creating a Success Page

For the condition you specified above, you must create a success page.

To create a success page



1. Click the “**NO**” statement icon, right-click, and select **Add Into > Feedback Section.**



2. Highlight the **Feedback Section** icon and double-click. Rename it “Success” and click **OK.**
3. Place the cursor on the right screen and type “Thank you for filling out our form. Have a good day.”

You have successfully created your Form/Database insertion. Save your project, upload it to your site, and test it.

Note: You cannot preview your Web site locally. Once the site is uploaded, access the index page by typing in the appropriate URL. Enter data into the form fields and then press Submit. You will want to download the database back to your hard disk and then open it with Access to verify the new information added.

Chapter 4.

Creating a Query/Search Page

Introduction

It is useful for a visitor to a Web site to be able to search a database, based on certain criteria, and find out the results of the search. Such an example would be for a visitor to be able to input someone's name and display the contact information of all the people with that name. The program allows you to import a database (Microsoft Access .MDB file) into your Web Site project so that data can be queried, added to or modified. You can then create pages that will access information from the database and display the results dynamically. One such example is creating a query page. As the functions and methodologies are complex, the use of these functions should be left to advanced users only. The following is a simple example of what can be done with the program.

Note: You are using the Active Server Pages (ASP) technology created by Microsoft. This technology currently operates on Windows NT Servers running Microsoft Internet Information Servers (IIS), or Windows NT Workstation running Peer Web Server, or Windows 95 running Personal Web Servers.

In this example you will search the database for an individuals' name and display the address. You should create this project on a test Web site before incorporating it into your Web Site project.

Create a Database

You must create an Access database with a table called **Address**. In this table, insert the following fields:

- **First_Name**
- **Last_Name**
- **Address**
- **City**
- **State**
- **Zip**

Note: It is important that you set any fields that use numbers to a text format.



Create the Test Data

You must enter test data for ten people so that you can use the database to test the dynamic pages that you are about to create. For this example, you create five entries with the same First_Name so that you can display the results of a query on multiple pages, using forward and backward navigation.

Important! ALL fields in the database must have some data. Using data that is incomplete creates programming difficulties.

Create a New Web Project to Perform a Query/Search

Now that you have created your database, you will want to create a new project file. Select **File > New** and select the template with an **X** in the **Preview** area.

Dynamic Page Icons

To complete your database project, the following Dynamic Page icons (Figure 4.1) will be visible in the Dynamic Page Manager window. These icons identify dynamic operations that you will use to update your database from a Dynamic Page.

- **Root Icon.** This is the Dynamic Page root directory. All supporting operation resources are created below this icon.
- **“IF” Statement Icon.** This is an evaluation operation. Use it to evaluate a condition against an expected value.
- **“OK” Statement Icon.** This is an action operation. Use it to initiate an action when the "IF" Statement is true.
- **“NO” Statement Icon.** This is an action operation. Use it to initiate an action when the "IF" Statement is false.
- **Statement Icon.** This is an action operation. Use it to assign a specific value to a parameter or a variable.
- **Feedback Section Icon.** This is a notification operation. Use it to display text or dynamically generated values to the Web site visitor.
- **Query Icon.** This is a update or access operation. Use it to send data to or read data from a database.
- **RecordSet Icon.** This is a sub-component of the query operation. Use it to cycle the results of the query to a feedback page.

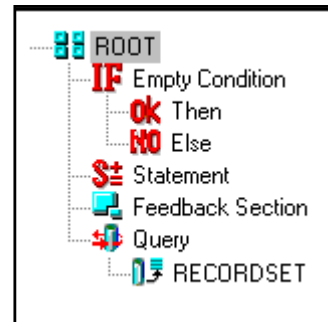


Figure 4.1 – Dynamic Page Icons

Importing the Database into the Project

Before you start to create your Dynamic Pages, you must first import the database and open it.

To import a project database

1. In the **Site Manager–Physical Layout**, right-click and select **Import Database**.



2. Navigate to the database that you created (named "Address") and click **Open**. A Database icon appears in the Physical Layout window.
3. Highlight the database, right-click and select **Properties**. The **Properties** dialog box appears. (Figure 4.2) Click on the **DataBase** tab.

Note: The Database tab provides you with a graphic representation of the database structure. A database structure is hierarchically arranged as follows: DATABASE > TABLES > FIELDS. One database can have many tables and each table will have its own fields.

You must open the database tables you want to access before the program can access the information. As you can see from Figure 4.2, you have two tables, address and table 1. Each table has its own fields.

To open the table

1. Highlight **Address** and click **Open**. After a few moments, the word "open" appears next to the Table icon.
2. Click **OK**. The necessary table is now open.

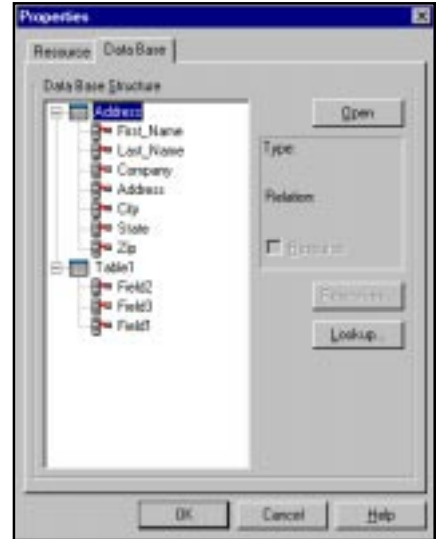


Figure 4.2 – Properties Dialog Box

Creating a Dynamic Page

Create a Dynamic Page by right-clicking in the Physical Layout window and selecting Create Dynamic Page. Double-click the page you have just created. The following window appears. (Figure 4.3)

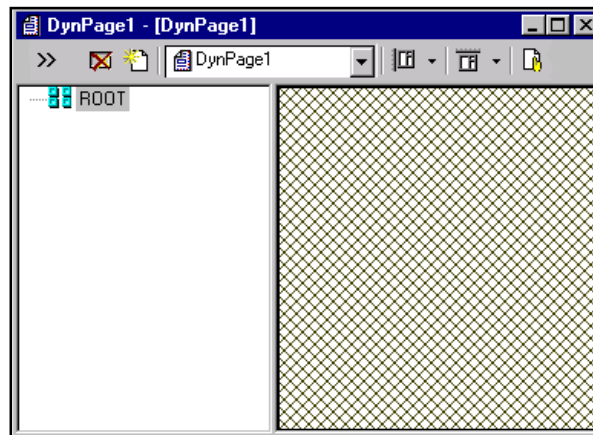


Figure 4.3 – Dynamic Page Window in Page Editor



Defining Parameters for a Dynamic Page

In the Dynamic window, you will see the **ROOT** icon. This is where you assign names and properties for parameters, variables and cookies. In this case, assign a parameter that will access the data input by users when performing the query. This data will be input on a form. Assign this parameter the query value “**P_Name**.”

Note: The prefix ‘P’ is a naming convention that identifies a data element as a parameter. Use this convention for easier data recognition, especially with complex tasks.

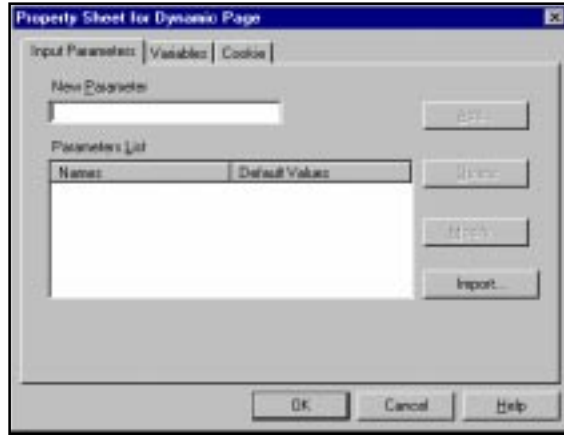


Figure 4.4 – Property Sheet for Dynamic Page Dialog Box

To define a parameter assign it a query value



1. Double-click the **ROOT** icon. The **Property Sheet for the Dynamic Page** appears. (Figure 4.4)
2. Click the **Input Parameter** tab
3. In the **New Parameter** input line, type “**P_Name**” and click **Add**.
4. Click **OK**.
5. Highlight “**P_Name**” and click **Modify**.
6. Assign “**P_Name**” the value “**null**.”

Note: You must assign the parameter a value if you intend to use the Query operation. If not, problems may result.

Working with the Query Function

Setting Up a Feedback Page

Now you must create a Feedback Page that includes a Header. Our first query output will be to identify what the query was. You want this information to appear only once per page and you also want it to appear at the top of the page.

To set up a feedback page with a header



1. Highlight the **ROOT** icon and right-click. a pop-up menu appears.
2. Select **Add Into > Feedback Section**. A **Feedback Section** icon appears.
3. Highlight the **Feedback Section** icon and double-click. Rename it “**Header**” and click **OK**.

- Place the cursor in the Page Editor window and type “The following are the results from your query of the name:” The insertion cursor remains.

To insert a dynamic element to display beneath the header

- Select **Dynamic Page > Insert Dynamic Element** or click **Insert Dynamic Element** on the Dynamic Page toolbar. The **Dynamic Elements** dialog box appears. (Figure 4.5)
- Select and double-click “**P_Name**.” The parameter “**P_Name**” is inserted onto the feedback page.

Note: When inserted, dynamic elements are shaded to indicate that actual values will be substituted.

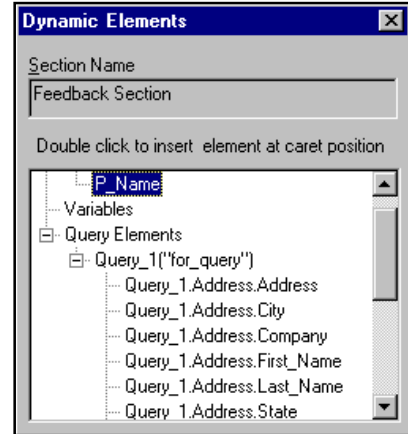


Figure 4.5 – Dynamic Elements Dialog Box

Setting Up a Query

To set up a query



- Highlight the **ROOT** icon, right-click, and select **Add Into > Query**. A **Query** icon appears.



- Double-click the **Query** icon. The **Choose Database** dialog box appears. Highlight the database you want to query.

Note: The database for this project was previously named “For_query.”

- Specify **SELECT** for Query Type and click **Edit Query**. The **Query Properties** dialog box appears. (Figure 4.6)

Note: The title bar for the Query Properties dialog box identifies the query name and the associated database.

- Click the **Select** tab and specify which fields to use. In this case, use the button **Add All**.
- Select the **Criteria** tab to specify some conditions. (Figure 4.7)

Add some conditions for the query to perform. By default, the query logic is set

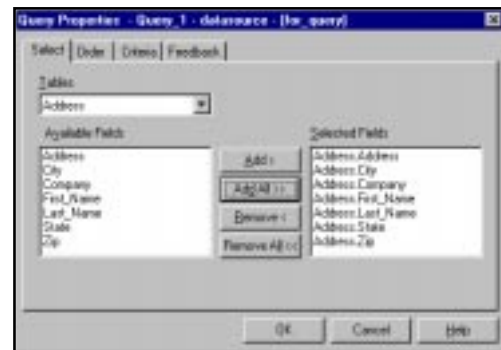


Figure 4.6 – Query Properties Dialog Box

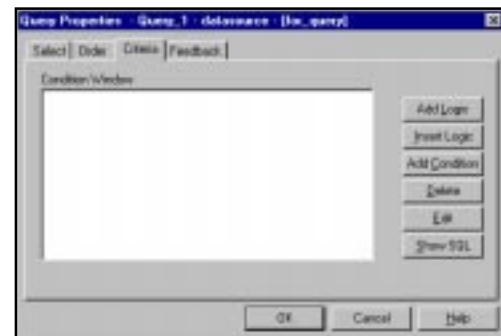


Figure 4.7 – Query Properties Dialog Box



for “AND.” In this case you want to use the function “OR.” This will allow the query to find values in either field.

To set up an OR query function

1. Click **Insert Logic**. The **Choose Logic Operation** dialog box appears.

Note: You must now specify the operation and conditions that the query will use to get the correct results.



Figure 4.8 – Add Condition Dialog Box

2. Select “**OR**” and click **OK**. The **OR** condition icon appears in the **Condition Window** area.
3. Click **Add Condition**. The **Add Condition** dialog box appears. (Figure 4.8)

Note: You are making a query for the user name. You will need to query both the First Name and Last Name. You want to match the first and last name to the parameter that the user inputs (that is, “P_Name,” equals the value that the user inputs).

4. In the left operand from the table “Address,” select the available field, “**Address.First_Name**”

Note: The field selection structure is “<table>.<field name>.”

5. For the operation, select “**CONTAINS**”

Note: This operation looks for any values in the database that match the input “P_Name.”

6. In the right operand, click the **Control** tab and select “**P_Name**” from the drop-down box.
7. Click **OK**. The **Query Properties** dialog box reappears.
8. Repeat Steps 1–7 to specify the operation and conditions for “**Address.Last_Name**.”

Note: When done, you must now set the number of records that are displayed per page.

9. Click the **Feedback** tab, click **Custom** and specify the number of records that you want viewed per page. In this case, use the value 3 because the database contains only 10 entries.

Creating a RecordSet

RecordSet is used to cycle the results of the query to the feedback page. Right now, you have set the feedback to contain three entries per page. There may be more than three entries per query, so you can create a RecordSet to remember all of the results.

To create a RecordSet



1. Highlight the **Query** icon, right-click, and select **Add Into > RecordSet**. A **RecordSet** icon appears.

Note: Now you are ready to display your dynamically produced results on a feedback page.



2. Highlight the **RecordSet** icon, right-click, and select **Add Into > Feedback Section**. A **Feedback Section** icon appears.



3. Double-click the **Feedback Section** icon. Rename it “Results” and click **OK**.

Note: Now you will add the dynamic information to the feedback page. For formatting reasons you will use tables to display information.

4. Place the cursor in the Page Editor window and select **Table > Insert Table**. The **Table Properties** dialog box appears.
5. Specify 2 rows, 5 columns, and 1 header. Click **OK**. The specified table is inserted.
6. Place the cursor in the top-left table cell and type “Name”
7. Tab to the next table cell and type “Address”
8. Repeat Steps 6–7 until the header row is complete.
9. Place the cursor in the second row first column.



10. To enter the dynamic elements, select **Dynamic Page > Insert Dynamic Element** or click **Insert Dynamic Element** on the Dynamic Page toolbar. The **Dynamic Elements** dialog box appears. (Figure 4.10)

11. Double-click “**Query_1.Address.First_Name**.”
12. Click on the table and move the cursor to the right of the shaded text.
13. Press the **SPACEBAR** once to insert a space between First and Last Name.
14. Click back on the **Dynamic Elements** dialog box and double-click “**Query_1.Address.Last_Name**.”
15. Place the cursor in the second row second column.
16. Click back on the **Dynamic Elements** dialog box and double-click “**Query_1.Address.Address**.”
17. Repeat Steps 15–16 until all dynamic elements are inserted into the remaining cells.

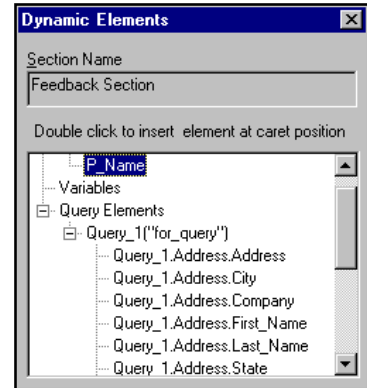


Figure 4.9 – Dynamic Elements Dialog

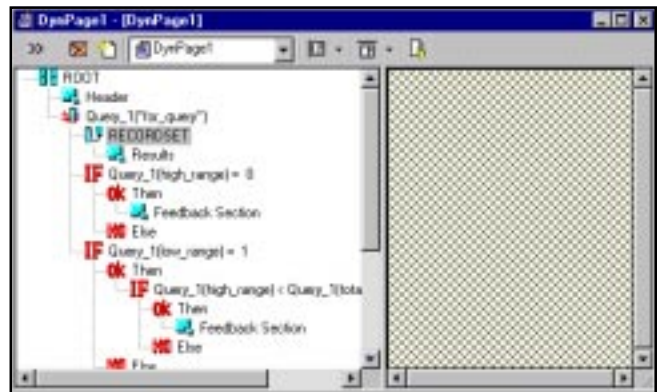


Figure 4.10 – Dynamic Page Window with Dynamic Page Manager



Creating Navigation Links For Feedback Pages

As you have only allowed three entries to be listed per page you will need to create links to the rest of the results. This requires that you evaluate the results of the query and set up a forward link (first page), forward and backward links (middle page) and backward link (last page).

Currently, any feedback specified as “custom” automatically generates the query variables “Forward_Navigation” and “Backward_Navigation” which you can dynamically place on a feedback page to generate the required link. These query variables generate the text links, “Next” and “Previous,” respectively.

In the following tutorial, these automatically generated text links are used on the forward link only feedback page and on the backward link only feedback page. Since many people like to use button links, rather than text links, the forward and backward feedback page tutorial illustrates how to assign forward and backward links to image files that are inserted onto the feedback page.

For No Result on Query

The first condition that you are going to determine is if there are “no results” for the query.

To determine if there are no results for the query



1. Highlight the **Query** icon, right-click, and select **Add Into > Condition**. The **Condition** icons IF–OK–NO appear.

Note: When complete, you will see the layout illustrated in Figure 4.10.



2. Double-click the “**IF**” statement icon. The **Set Condition** dialog box appears. (Figure 4.11)
3. Click the **Query** tab.

Note: One of the functions listed is called High_Range. This is the number of resulting entries found in the database as a result of the query. (It is an absolute value.) You will use this and Low Range to determine what feedback page should be implemented, if there are no results from the query.

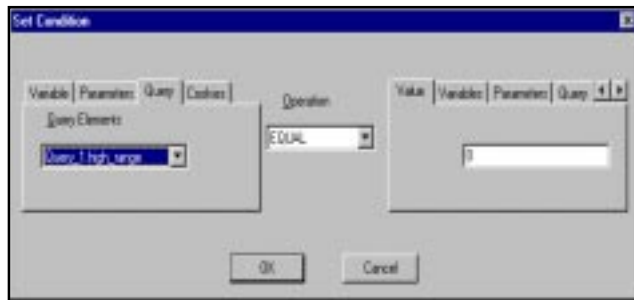


Figure 4.11 – Set Condition Dialog Box

4. For the left operand, select “**Query_1.High_Range.**”

Note: You want to determine if the “High_Range” is equal to zero (that is, no result).

5. For the operation, select **EQUAL**.
6. For the right operand, click the **Value** tab and enter the value 0.
7. Click **OK**.

Note: Notice that under the “IF” statement, there are two sub components called “OK” and “NO.” In this case, if the condition high range equals 0 is true (OK) you will want to have some feedback telling the user that there was no person found by that name.



8. Highlight “OK” statement icon, right-click, and select **Add Into > Feedback Section**. The **Feedback Section** icon appears.



9. Double-click the **Feedback Section** icon. Rename it “NO_Results” and click **OK**.

10. Place the cursor in the Page Editor window and type “There was no match for the person.”



11. Select **Dynamic Page > Insert Dynamic Element** or click **Insert Dynamic Element** on the Dynamic Page toolbar. The **Dynamic Elements** dialog box appears. (Figure 4.12)

12. Select “P_Name” and press **RETURN** once.

13. On the next line, type “To try a new query, click here.”

14. Highlight the link text “click here,” right-click and select **Insert > Link**. The **Link** dialog box appears. (Figure 4.13)

15. Click the **To Resource** tab and select “index.htm.”

16. Click **Assign URL**.

You have just completed the first part of the navigation of Query Results. Do not use the “NO” statement icon. There is no need to specify this part of the condition.

Creating Navigation Links When There Are Results

To define the rest of the conditions for navigation you will need to define a starting value. As you know your first page of results you will not need a backward link. So, for your next condition you use the IF/ELSE function.

To set up an IF/ELSE condition



1. Highlight the **Query** icon, right-click, and select **Add Into > Condition**. The **Condition** icons IF–OK–NO appear.



2. Double-click the “IF” statement icon. The **Set Condition** dialog box appears.
3. For the left operand, select “Query_1.Low_Range.”
4. For the operation, select “EQUAL.”

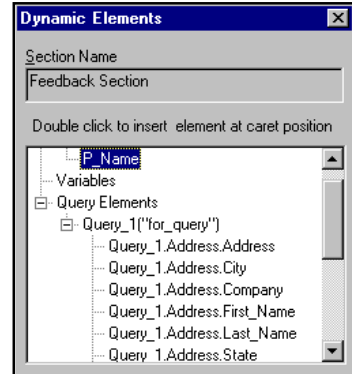


Figure 4.12 – Dynamic Elements Dialog

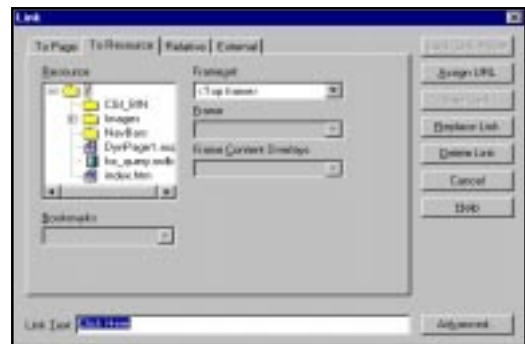


Figure 4.13 – Link Dialog Box



- For the right operand, click the **Value** tab and enter the value 1.

Setting Up A Forward Link

For the first page, you know that if the condition “low_range = 1” is true, then the first page has been identified. However, put another way, the “IF” condition icon should more completely state that “If low range equals 1 *and* the high range is less than the total number of records, then print Forward Navigation feedback only.” To complete the full evaluation the first evaluation, if “Query_1.Low_Range = 1,” leads to the “OK” action where you must set up a second evaluation, if “High_Range” is less than the total number of records. At this point, queries containing at least one page of results have been identified.

To create the Forward Navigation link for the first page



- Highlight the “OK” statement icon for the condition “**Query_1(Low_Range)=1,**” right-click and select **Add Into > Condition**. The **Condition** icons IF–OK–NO appear.



- Select the “IF” statement icon you just created and double-click. The **Set Condition** dialog box appears.
- For the right operand, click the **Query** tab and select “**Query_1.High_Range.**”
- For the operation, select “<”
- For the right operand, click the **Query** tab and select “**Query_1.Total_Record_Number.**”

Note: You will now create the feedback page if both these conditions are true (that is, the first page).



- Highlight the “OK” statement icon for the condition “**Query_1(High_Range) < Query_1(Total_Record_Number),**” right-click, and select **Add Into > Feedback Section**. A **Feedback Section** icon appears.



- Double-click the **Feedback Section** icon. Rename it “Forward_Only” and click **OK**.

- Highlight “**Forward_Only**” and place the cursor in the Page Editor window.



- Select **Dynamic Page > Insert Dynamic Element** or click **Insert Dynamic Element** on the Dynamic Page toolbar. The **Dynamic Elements** dialog box appears.

- Double-click “**Query_1.Forward_Navigation.**”

Setting Up Forward and Backward Links

To create Forward and Backward navigation, use arrow images as the navigation links. To do this, use the “NO” statement for the condition “Query_1(Low_Range)=1.” Define the condition “If ‘Low_Range’ is not = to 1 and ‘High_Range’ is < ‘Total_Record_Number’” (that is, the pages ‘in between’ the first and last page).

To create the Forward and Backward Navigation links for the middle pages



- Highlight the “NO” statement icon (should be the last item on the list), right-click, and select **Add Into > Condition**. The **Condition** icons IF–OK–NO appear.



- Select the “IF” statement icon and double-click. The **Set Condition** dialog box appears.
- For the left operand, click the **Query** tab and select “**Query_1(High_Range).**”

4. For the operation, select “<”
5. For the right operand, click the **Query** tab and select “**Query_1(Total_Record_Number)**.”
6. Highlight the “**OK**” statement icon, right-click, and select **Add Into > Feedback Page**. A **Feedback Section** icon appears.
7. Double-click the **Feedback Section**. Rename it “**Forward_Backward**” and click **OK**.
8. Highlight “**Forward_Backward**” and place the cursor in the Page Editor window.

Note: Instead of simply adding the Dynamic Element of Forward and Backward Navigation, use a left and right arrow as a link to the next or previous pages.

9. Select **Insert > Image** or click **Insert Image** on the **Insert Object** toolbar. The **Image Attributes** dialog box appears. (Figure 4.14)
10. Click the **Library** tab, use the drop-down menu, and select **Arrows**.
11. Select an arrow image pointing to the left and click **OK**.
12. Place the cursor to the right of the arrow and press **TAB**.
13. Repeat Steps 9–11 to insert an arrow image pointing to the right.
14. When done, highlight the left arrow image, right-click, and select **Insert > Link**. The **Link** dialog box appears.
15. Click the **To Resource** tab and select “**DynPage_1.asp**.”
16. Click **Advanced**. The **Advanced WEB Operations** dialog box appears.
17. Select the **Dynamic Object** tab.
18. Select **HREF** and click **Modify**. The **Modify** dialog box appears. (Figure 4.15)
19. Click the **Query** tab and using the drop-down menu, select “**Query_1.Backward_Navigation**.”
20. Click **OK** twice and click **Assign URL**.
21. Repeat Steps 14-20 for the arrow pointing right (Forward Navigation) except when using the **Modify** dialog box, choose “**Query_1.Forward_Navigation**.”

This completes the creation of Forward and Backward Navigation for the query RecordSet. The

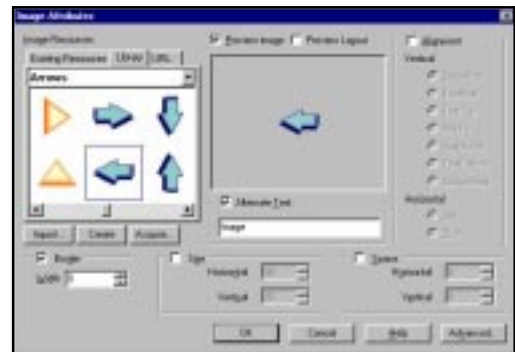


Figure 4.14 – Image Attributes Dialog Box

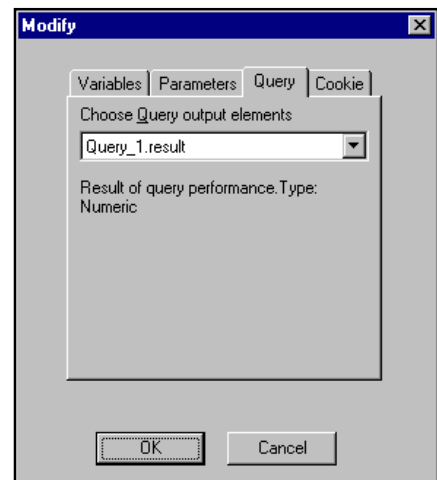


Figure 4.15 – Modify Dialog Box



Feedback Page, “Forward_Backward” appears only if “Low_Range is not = 1 and High_Range is >=Total Record Amount.”

Setting Up A Backward Link

Create a Backward Navigation link for the last page. To do this, specify the condition: “If the ‘High_Range’ is greater than or equal to the ‘Total_Record_Number,’ then a feedback page with only Backward navigation is provided.”

To create the Backward Navigation for the last page



1. Highlight the “**NO**” statement icon for the condition “**Query_1(Low_Range) = 1,**” right-click and select **Add Into > Condition**. The **Condition** icons IF–OK–NO appear.



2. Select the “**IF**” statement icon and double-click. The **Set Condition** dialog box appears.

3. For the left operand, click the **Query** tab and select “**Query_1.High_Range.**”

4. For the operation, select “**> =**”

5. For the right operand, click the **Query** tab and select “**Query_1.Total_Record_Number.**”



6. Highlight the “**OK**” statement icon for the condition that you just created, right-click, and select **Add Into > Feedback Page**. A **Feedback Section** icon appears.



7. Double-click the **Feedback Section** icon. Rename it “**Backward_Only**” and click **OK**.



8. Select **Dynamic Page > Insert Dynamic Element** or click **Insert Dynamic Element** on the Dynamic Page toolbar. The **Dynamic Elements** dialog box appears.

9. Double-click “**Query_1.Backward_Navigation.**”

Creating a Form for Use with a Dynamic Page

Create a form so that a user can input some information for the query. To do this, go to **Site Manager–Logical Layout** and double-click on the index page. Click **Show/Hide** on the Standard toolbar. You will now see a display symbol whenever a paragraph ends. You will also see, after inserting a form, two lines, that designate the form area. All form fields must be within these lines.

To create a form

1. From an open page in Page Editor, position the cursor where you want to insert the form.



2. Click **Show/Hide** on the Standard toolbar. Paragraph symbols appear.

3. Select **Insert > Form**. Two lines designating the form area appear.

4. Place the cursor inside the two lines and type “Please enter the name of the person you wish to find.”

5. Press **ENTER**.

6. Select **Insert > Form Fields > Text Line**. The **Text Line Properties** dialog box appears.

7. In the **Name** input line, type “**P_Name**”

8. Click **OK**.

9. Select **Insert > Form Field > Push Button**. The **Push Button Properties** dialog box appears.



10. To accept all defaults, click **OK**.
11. Right-click somewhere in the form and select **Forms**. The **Form Properties** dialog box appears.
12. In the **Action** section, click **Local**, select “**DynPage1.asp**” and click **OK**.

You have now completed all steps involved with making your query page. Save your project, upload it to your site, and test it.

Note: You cannot preview the query locally on your browser. Once uploaded, access the index page by specifying the appropriate URL. Enter in the name that occurs most frequently in the database and press Submit. You will see the results displayed in the table with links on the bottom.

Chapter 5.

Redirecting URLs

Introduction

Redirect can have many uses. Only two are examined here. The first is a redirect to a new page. The second is a redirect that lets you download resources (such as, programs, documents, files and so on) automatically.

Page Redirects

There are two ways to redirect to a new page.

1. Have **index.asp** automatically redirect you, or
2. Set up a link that redirects you.

When redirecting to a new page, you can redirect to an internal or external Web page.

Download Redirects

Using redirect enables visitors to download resources such as programs, documents and files. It is an efficient method for you to enable downloading. However, to do this, you must ask your Web-hosting Company to modify the permission settings on your **Download** folder.

If you intend to allow visitors to download numerous resources, do not import them directly into your WebShop Project files. Your project file will become very large and unmanageable. It is recommended that you create an external folder on your hard disk (not in your Web Project folder) called “downloads.” Have your redirects linked to this folder. Once you have completed the Web site development, upload this folder to your remote Web server using an FTP program. Visit www.boomerangsoftware.com for referrals on FTP programs.

When creating a redirect, be sure to access your resources from the specified downloads folder.

Requesting ISP Information and Setting Up

To request ISP information

1. Upload the **Downloads** folder.
2. Ask the ISP Systems Administrator to open the IIS Server Manager.
3. Ask the Systems Administrator to open your Web site and highlight the **Downloads** folder.



4. Right-click and select **Properties**.
5. On the **Directory** tab, select “**None**” for permissions and “**Read**” for access permissions.

Getting Started

In this example, you will redirect **index.asp** to **Page1.htm**.

Scenario: You have recently acquired a new domain name and wish it to be redirected to your new site. Once the redirect is established you must create a link to a form where you can download resources. You then must link this form to a Dynamic Page that validates user information and proceeds to download. You may also add other dynamic elements (to send email or enter data into a database). In this case, you must send an email to both the visitor and the Webmaster. Each email contains different information.

Create a New Project

To begin your project, open a blank template

1. Select **File > New** and click on a blank template. **Site Manager–Physical Layout** appears.
2. Delete the file “index.htm.”
3. Right-click and select **Create New Dynamic Page**. The file “**DynPage1.asp**” appears.
4. Rename this file “index.asp.”

Redirecting the Page to Another Page

To redirect to another page

1. Double-click **index.asp**.
2. Right-click the **ROOT** icon and select **Add Into > Redirect**.
3. Double-click the **Redirect To** icon. The **Property Sheet for Redirect Action** dialog box appears. (Figure 5.1)
4. Click the **External** tab.
5. In the **Custom URL** box, type the URL. In this case, use <http://www.yourdomainname.com/Page1.htm>.
6. Click **OK**.

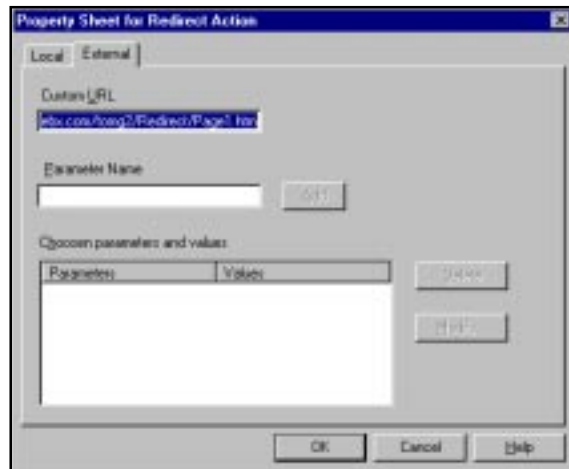


Figure 5.1 – Property Sheet for Redirect Action

You have not yet redirected your clients to a new URL. They are still in the same URL. All you have done is to make your index.asp page automatically redirect a visitor without making them perform any additional work.

Create a Link to a Download Form

To create a link to a download form

1. From **Site Manager–Physical Layout**, create three new pages: two Plain Pages called **Page1.htm** and **Page2.htm** and one Dynamic Page called **DynPage2.asp**
2. Double-click **Page1.htm**.
3. Type “Download Now.”
4. Highlight “**Download Now**,” right-click, and select **Link**. The **Link** dialog box appears.
5. Click the **To Page** tab and select **Page2.htm**.
6. Click **Assign URL**. The link is created.

Create a Download Form

1. From Site Manager, open **Page2.htm** in Page Editor, position the cursor where you want to insert the form.



2. Click **Show/Hide** on the Standard toolbar. Paragraph symbols appear.
3. Select **Insert > Form**. Two lines designating the form area appear.
4. Place the cursor inside the two lines.
5. Select **Table > Insert Table**. The **Table Properties** dialog box appears.
6. Select four rows and two columns.
7. Click **OK**.
8. Double-click on the table object. The cursor moves into a table cell.
9. Place the cursor in the top left cell and type “Name:”
10. Place the cursor in the top row second column and select **Insert > Form Fields > Text Line**. The **Text Line Properties** dialog box appears.
11. Type the name “P_Name” and delete the default Value.
12. Click **OK**.
13. In the second row first column, type “Email:”
14. In the second row second column, add a Text Line form element and name it “Email.”
15. Delete the default Value.
16. In the third row first column, type “Product:”
17. In the third row second column, add a combo box form element and name it “Select_Prod.” Select **Insert > Form Field > Combo Box**. The **Combo Box Properties** dialog box appears. (Figure 5.2)
18. Under the **Items Section**, highlight **Item 1** and click **Edit Item**. The **Item Properties** dialog box appears. (Figure 5.3)
19. In the **Item Label**, type “Prod_One” and for the **Item Value**, type “prod1.”
20. Click **OK**.



Figure 5.2 – Combo Box Properties



21. Repeat Steps 18–20 for the Item 2, use the label “Prod_Two” and the value “prod2.”
22. Click **OK**.
23. In the fourth row first column, add a **Submit** button. Select **Insert > Form Field > Push Button**.
24. Click **OK** to accept the defaults.
25. Place the cursor outside the table, but still within the two lines, and right-click. A pop-up menu appears.
26. Select **Form**. The **Form Properties** dialog box appears.
27. Select **Local** and highlight **DynPage2.asp**
28. Click **OK**.

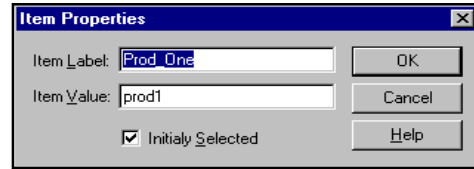





Figure 5.3 – Item Properties for Prod1

Creating a Dynamic Page

To create the Dynamic Page that serves to redirect the data entered

1. Double-click **DynPage2.asp**
-  2. Double-click the **ROOT** icon. The **Property Sheet for Dynamic Pages** appears.
3. Select the **Input Parameter** tab and add the Parameters “**P_Select_Prod,**” “**P_Name**” and “**P_Email.**”
4. Click on the **Variables** tab and add the Variables “**var_any_error,**” “**var_name,**” and “**var_email.**”
5. Click **OK**.

You will now create the validation of the form. In this case, all fields are required (Name and Email). Do not validate the combo box; they already have default values. The first validation is to test for **Name**. To do this, add a condition for the variable “**P_Name.**”

-  1. Right-click the **ROOT** icon and select **Add Into > Condition**. An “**IF**” statement icon appears.
-  2. Double-click the “**IF**” statement icon. The **Set Conditions** dialog box appears. (Figure

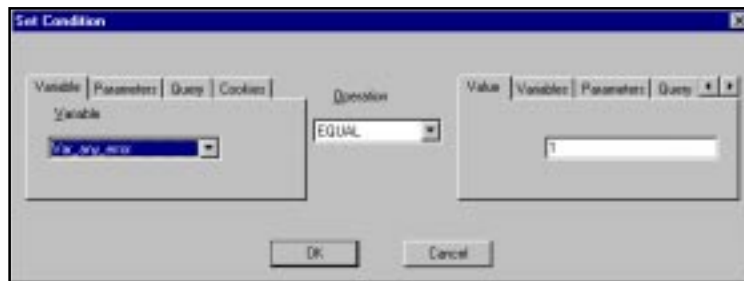


Figure 5.4 – Set Condition Dialog Box

5.4)

3. For the left operand, select the **Parameter** tab and select “P_Name.”
4. For the operation, select “EQUAL.”
5. For the right operand, select the **Value** tab and leave it blank.
6. Click **OK**.

Now that you have created the condition, you must specify the action that will occur if that condition is true.

In this case, you will want to use the variables that you have created to log the errors. In addition, you will want to log that “var_name” equals the value “Name” and that “var_any_error” equals 1. You assign “var_name” this value so that you can insert this dynamic element onto a feedback page later.

Remember: To know that there was an error, use “var_any_error” and assign it a value of 1. This value is arbitrary. It can be anything you want.

Assign “var_name” a value



1. Highlight the “OK” icon, right-click and select **Add Into > Statement**.



2. Double-click the **Statement** icon. The **Set Variable To** dialog box appears. (Figure 5.5)
3. Under **Choose Variable Name**, select “var_name.”
4. For the operation, select “EQUAL.”
5. For the right operand, select the **Value** tab and type “Name.”

Note: This assigns the variable with the value “Name.” Later you will use this variable in the feedback page.



Figure 5.5 – Set Variable To Dialog Box

6. Click **OK**.

Assign “var_any_error” a value



1. Highlight the “OK” icon, right-click and select **Add Into > Statement**.



2. Double-click the **Statement** icon. The **Set Variable To** dialog box appears.
3. Under **Choose Variable Name**, select “var_any_error.”
4. For the operation, select “EQUAL”
5. For the right operand, select the **Value** tab and type the number “1”

Note: Later, you will test to see if this variable has a value, and if so, it will generate a feedback page.

6. Click **OK**.

Repeat the above two procedures for the parameter “P_Email.”



Creating a Feedback Page

If a visitor filling out the form *has not* typed in a field, then the value “1” is added to “**var_any_error.**” If the visitor *has* filled out every field, then “**var_any_error**” has no value (that is, it remains blank).

To create an error feedback page, you must test the condition IF “**var_any_error = 1.**” Earlier, you specified that IF “**P_Name**” or “**P_Email**” is blank then “**var_any_error = 1.**”



1. Right-click the **ROOT** icon and select **Add Into > Condition.**



2. Double-click the “**IF**” statement icon. The **Set Condition** dialog box appears.

3. For the left operand, select the **Variables** tab and select “**var_any_error.**”

4. For the operation, select “**EQUAL.**”

5. For the right operand, select the **Values** tab and type the number “1”

6. Click **OK.**



7. Highlight “**OK**” statement icon, right-click and select **Add Into > Feedback Section.**



8. Double-click the Feedback Section icon. Rename it “**Error_Page.**”

9. Highlight “**Error_Page**” and place the cursor in the section on the right. Type “You have not entered the following required fields:” and press **ENTER.**



10. Click **Insert Dynamic Element** on the Dynamic Page toolbar. The **Dynamic Element** dialog box appears. (Figure 5.6)

11. Double-click “**var_name.**” A shadowed box appears. Place the cursor to the right of the shaded box and press **ENTER.**

Repeat this procedure for the Dynamic Element “**var_Email.**” You are now done with the feedback page.

Next condition: If the previous “**IF**” statement is not true, then there are no assumed errors. Here you will use the “**NO**” statement (from the condition “**var_any_error**” = “1”) to send emails to both the Webmaster and the visitor.

Action: Create custom emails for both parties.

Creating Emails for the Webmaster and the Visitor

For both emails



1. Right-click the “**NO**” condition icon and select **Add Into > Send Mail.** A **Send Mail** icon appears.

2. Repeat Step 1 for the second email. Another **Send Mail** icon appears.

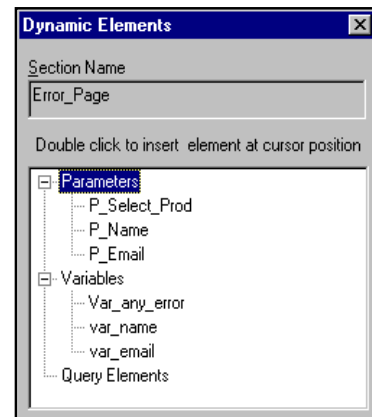


Figure 5.6 – Dynamic Elements Dialog Box

For the email to the Webmaster



1. Double-click the first **Send Mail** icon. The **Send Mail Property** dialog box appears. (Figure 5.7)
2. In the **From** section, select the **Parameters** tab and select “**P_Name.**”
3. For the **To** section, type your email address.
4. In the **Subject** section, enter a title like “Email from download customer”
5. In the **SMTP Server** section, type the email server address for your Web-hosting Company. For example, “mail.<yourISPname>.com.”
6. Click **OK**.



7. Highlight the **Send Mail** icon and place the cursor in the right-hand side.
8. Click the **Dynamic Element Insertion** icon. The **Dynamic Element** dialog box appears.
9. Double-click “**P_Name,**” place the cursor to the right of the shaded box and press the **SPACEBAR** once. Type “has purchased the following product”
10. Click on the **Dynamic Element** box and double-click “**P_Select_Prod.**”

For the email to the Visitor



1. Double-click the second **Send Mail** icon. The **Send Mail Property** dialog box appears.
2. In the **From** section, type your email address.
3. In the **To** section, select the **Parameters** tab and select “**P_Name.**”
4. In the **Subject** section, type the email title “Auto Response from Downloading”
5. In the **SMTP Server** section, type the email server address for your Web-hosting Company. For example, “mail.<yourISPname>.com”
6. Click **OK**.



7. Highlight the **Send Mail** icon and place the cursor in the right-hand side.
8. Type “Thank you for downloading our”
9. Click on the **Dynamic Element** box and double-click “**P_Select_Prod.**”





You are to do the download redirection last because it should always be done at the end. After its execution, the script cannot perform any other actions. The download redirection should look for the product that was selected in the combo box. Earlier, you assigned these products the values prod1 and prod2, respectively.







Figure 5.7 – Send Mail Property Dialog Box for Webmaster Email



Redirect for Product 1

-  1. Select the “**NO**” icon for “**var_any_error = 1,**” right-click, and select **Add Into > Condition**.
-  2. Double-click the “**IF**” condition icon. The **Set Condition** dialog box appears.
3. For the left operand, select the parameter “**P_Select_Prod.**”
4. For the operation, select “**EQUAL.**”
5. For the right operand, select the **Values** tab and type “**Prod1**”
6. Click **OK**
-  7. Select the “**OK**” icon for that condition, right-click, and select **Add Into > Redirect To**.
-  8. Double-click the **Redirect To** icon. The **Property Sheet for Redirect Action** appears.
9. Click on the **External** tab.
10. In the **Custom URL** box, type “<http://www.yourdomainname.com/downloads/prod1.exe>”
Note: Prod1.exe is a fictional product or resource.
11. Click **OK**.

Redirect for Product 2

-  1. Select the “**NO**” statement icon for “**var_any_error = 1,**” right-click, and select **Add Into > Condition**.
-  2. Double-click the “**IF**” condition icon. The **Set Condition** dialog box appears.
3. For the left operand, select the parameter “**P_Select_Prod.**”
4. For the operation, select “**EQUAL.**”
5. For the right operand, select the **Values** tab and type “**Prod2**”
6. Click **OK**.
-  7. Select the “**OK**” statement icon for that condition, right-click, and select **Add Into > Redirect To**.
-  8. Double-click the **Redirect To** icon. The **Property Sheet for Redirect Action** appears.
9. Click on the **External** tab.
10. In the **Custom URL** box, type <http://www.yourdomainname.com/downloads/prod2.exe>
Note: Prod1.exe is a fictional product or resource.
11. Click **OK**.

You have now completed your Page Redirect and Download Redirect.

Chapter 6.

Cookies and Requesting Information

Introduction

Cookies are small files located on the client's computer. These files contain persistent state information, such as user information. The server, where the Web site resides, can then request this information from the client browser to perform a task. Two tasks are required when using cookies. First, they must be sent and written to the client computer. Second they must be retrieved/requested by the server from the client computer. Each cookie should have a name and value (key). Complex cookies, like cookie dictionaries, have more than one key associated with a page. Keys can have any of the following attributes:

- **Domain**—This makes sure that when the cookie is requested it will only go back to its own domain.
- **Expiration Date**—This is a date that the cookie will expire and self-remove itself from the client computer
- **Path**—This will make the cookie return itself only if requested from the page that issued it.
- **Secure**—This feature will require a secure connection over the Internet before it is sent.

It is important to note that cookies should only be used for non-sensitive or non-confidential information.

Note: To achieve these tasks, the program uses Active Server Pages (ASP) technology. ASP is the standard promoted by Microsoft and it operates on Windows NT Servers running Microsoft Internet Information Servers (IIS), Windows NT Workstation running Peer Web Server, or for Windows 95 running Personal Web Server. Recently other companies have developed ASP technology running on UNIX and Solaris operating systems. Refer to www.boomerangsoftware.com for more details.

This example will show how you can take some user information and send the user a cookie. Then access the cookie information and produce a dynamic feedback page.

Note: Do not take information from the client. You may send them a cookie, which you will later recall.



Creating Your Project

Select **File > New** and select a blank template. For the following project, create two Dynamic Pages.

Creating a Dynamic Page

To create a Dynamic Page

1. Go to **Site Manager–Physical Layout**, right-click, and select **Create Dynamic Page**. The file **DynPage1.asp** appears.
2. Repeat Step 1. The file **DynPage2.asp** appears.

Creating a Form for use with a Dynamic Page

To create a form on the index page

1. From **Site Manager–Physical Layout**, double-click “**index.htm**.” The Index page opens. Position the cursor where you want to insert the form.



2. Click **Show/Hide** on the Standard toolbar. Paragraph symbols appear.
3. Select **Insert > Form**. Two lines designating the form area appear.
4. Place the cursor inside the two lines. All form fields must be within these two lines.
5. Place the cursor inside the two lines and type “Full Name”
6. Select **Insert > Form Fields > Text Line**. The **Text Line Properties** dialog box appears.
7. For **Name**, type “Full_Name” and click **OK**.

After you have completed the above steps, add a Submit button in the same page and in the same form.

To add a Submit button

1. Place the cursor inside the form area.
2. Select **Insert > Form Field > Push Button**. The **Push Button Properties** dialog box appears.
3. Click **OK** to accept Push Button defaults. The Push Button is inserted.
4. Right-click somewhere in the form (in the white space) and select **Forms**. The **Form Properties** dialog box appears.
5. In the **Action** section, click on **Local** and select “**DynPage1.asp**.”
6. Click **OK**. The Submit button is added.

You have now created the form which will send the client computer a cookie.



Creating a Cookie

In the following, DynPage1.asp will take the client information “**Full_Name**” and assign a cookie.

To create a cookie

1. Go to **Site Manager–Physical Layout** and double-click **DynPage1.asp**. The selected page opens.
2. Double-click the **ROOT** icon. The **Property Sheet for the Dynamic Page** appears.
3. Click the **Cookie** tab. The **Cookies** fields appears.
4. For **Cookie Name**, type “**First_Cookie**” and specify an **Expiration Date**.
5. For **New Key**, type “**Name**” and click **Add**.
6. Click the **Variable** tab.
7. For **New Variable**, type the following variables and click **Add**: “**v_name**” and “**v_full_name.**” The new variables appear in the variable listing.
8. Click the **Input Parameters** tab.
9. For **New Parameter**, type “**Full_Name.**” The new parameter appears in the parameter listing.

Creating a Field Validation Test

In this section, create a test to make sure that the field was filled out. To do this, follow the steps below:

1. Highlight the **ROOT** icon, right-click, and select **Add Into > Condition**. An “**IF**” condition icon appears.
2. Double-click the “**IF**” statement icon. The **Set Condition** dialog box appears.
3. Select the **Parameters** tab.
4. In the left operand, select “**Full_Name**” from the drop-down box.
5. For the operation, select “**EQUAL.**”
6. For the right operand, select the **Value** tab and enter nothing (that is, leave it blank).
7. Click **OK**.

Note: What is being stated is that if the parameter “Full_Name” has no value (left blank) when submitted, then note this in the variable “var_full_name.” Now assign the variable “var_full_name” a value if the above statement is true.
8. Highlight the “**OK**” statement icon for that condition. Right-click and select **Add Into > Statement**.
9. Double-click the **Statement** icon. The **Set Variable To** dialog box appears.
10. For **Choose Variable Name**, select “**var_full_name.**”
11. For the operation, select “**Concatenate.**”



Note: Concatenate means “to join to.”

12. For the right operand, select the **Value** tab and type “Full Name”
13. Click **OK**.

Creating a Error Page

Now create the Feedback Page that will be used to display an Error Message.



1. Highlight the **ROOT** icon, right-click, and select **Add Into > Condition**.



2. Double-click the “**IF**” statement icon. The **Set Condition** dialog box appears.
3. For the left operand, select the **Variable** tab and select the variable “**var_full_name.**”
4. For the operation, select “**NOT EQUAL.**”
5. For the right operand, select the **Value** tab and leave the value blank.

This states that IF “**var_full_name**” contains any information (from the validation), then produce an error message (“OK”). If the condition is not true (“NO” or “ELSE”), then assume that all the fields were filled and consequently produce a success page.

For the condition, you specified above, create the error page:



1. Click the “**OK**” statement icon, right-click, and select **Add Into > Feedback Section**.



2. Highlight the **Feedback Section** icon, double-click, and rename it “Error.” Click **OK**.
3. Place the cursor on the right screen and type “You have not entered the following fields:” and press **ENTER**.
4. Click **Insert Dynamic Element** on the Dynamic Page toolbar. The **Dynamic Elements** dialog box appears.
5. Double-click “**var_full_name.**” This parameter appears in Page Editor.
6. Close the dialog box.

Writing a Cookie

For the condition, “**var_full_name.**” use the “**NO**” section to write the cookie. Place this there because if “**var_full_name**” is empty (blank) then you assume that the field has been filled. Convert the value of the parameter to a variable “**v_name.**” This is because the cookie can only be assigned a value from a variable. Follow the steps below to write the cookie:



1. Right-click the “**NO**” statement icon for “**var_full_name**” and select **Add Into > Statement**.



2. Double-click the **Statement** icon. The **Set Variable To** dialog box appears.
3. For the left operand, select “**v_name.**”
4. For the operation, select “**=**”
5. For the right operand, select the **Parameter** tab and select “**Full_Name**”
6. Click **OK**.



7. Right-click the “**NO**” statement icon for “**var_full_name**” and select **Add Into > Cookie**.



8. Double-click the **Write Cookie** icon. The **Write Cookie** dialog box appears.
9. For **Choose Key**, select “**Name**.”
Note: This is the Cookie key that was added to the Cookies Properties.
10. For the **Choose Variable** drop-down box, select “**v_name**.”
11. Click **OK**.

Creating a Feedback Page



1. Click the “**OK**” statement icon, right-click, and select **Add Into > Feedback Section**.



2. Highlight the **Feedback Section** icon, double-click, and rename it to “**Success**.” Click **OK**.
3. Place the cursor in the Page Editor window.
4. Type “A cookie has been written to your computer. To read a cookie click here.”
5. Highlight “**click here**,” right-click, and select **Insert > Link**. The **Link** dialog box appears.
6. Click the **To Resources** tab and select **DynPage2.asp**.
7. Click **Assign URL**.

Reading and Requesting Cookies

Now we will set the request that the browser/client send the cookie information back to the server before it sends the Web page. In this case, create a customized Welcome page. Address the person’s name in a statement that welcomes him/her to your Web site.

1. From **Site Manager–Physical Layout**, double-click **DynPage2.asp**.



2. Double-click the **ROOT** icon. The **Property Sheet for Dynamic Page** appears.
3. Click the **Cookie** tab.
4. For **Cookie Name**, type “**First_Cookie**” and specify an expiration date.
5. Click the **Variables** tab.
6. For **New Variable**, type “**var_name**.”
7. Click **OK**.



8. Highlight the **ROOT** icon, right-click, and select **Add Into > Statement**.



9. Double-click the **Statement** icon. The **Set Variable To** dialog box appears.
10. For the left operand, select “**var_name**.”
11. For the operation, select “**=**”
12. For the right operand, select the **Value** tab and specify a name.

Note: This is the key from the cookie properties.

13. When done, click **OK**.



14. Highlight the **ROOT** icon, right-click, and select **Add Into > Request**. The **Request From User** dialog box appears.
15. For the right operand, select the **Cookie** tab and select the variable “**var_name**.”



16. Click **OK**.



17. Highlight the **ROOT** icon, right-click, and select **Add Into > Feedback Section**.



18. Highlight the **Feedback Section** icon, double-click, and rename it to “Hello.” Click **OK**.

19. Click **Insert Dynamic Element** on the Dynamic Page toolbar. The **Dynamic Elements** dialog box appears.

20. Select and double-click “**var_name**.” This parameter appears in Page Editor.

21. Close the dialog box. The blinking cursor appears in the Page Editor window.

22. Continue typing “Welcome back to our Web site.” The Welcome page is complete.

Save your project, upload it to your site and test it.

Note: You cannot preview locally. Once uploaded, access the index page by typing the appropriate URL. Enter data into the form fields and press Submit.

Requesting Information from the Server

This program enables you to request information from the server. You can specify what information you want to get and then have it called from the server. One such command is “HTTP_USER_AGENT.” This command will detect what version browser a user is using. This may be useful to provide a Web site that uses the latest technology and a basic site for visitors that have lower version browsers.

To detect the browser version

1. Create a new project.
2. From **Site Manager–Physical Layout**, right-click and select **Create a Dynamic Page**. DynPage1.asp appears.
3. Double-click “**index.htm**.” The Index page opens.
4. Type “Click on the link to detect what version browser you are using.”
5. Select the text entered and select **Insert > Link** to make a link to DynPage1.asp. The Link dialog box appears. Complete the fields as necessary.
6. Navigate back to **Site Manager–Physical Layout** and double-click “**DynPage1.asp**.”



7. Double-click the **ROOT** icon. The **Property Sheet for Dynamic Page** appears.

8. Click on the **Variables** tab.

9. For **New Variable**, type “v_browser” and click **Add**.



10. Highlight the **ROOT** icon, right-click, and select **Add Into > Statement**.



11. Double-click the **Statement** icon. The **Set Variable To** dialog box appears.

12. For the left operand, select “v_browser.”

13. For the operation select “=”

14. For the right operand select the **Value** tab and type “HTTP_USER_AGENT”



15. Highlight the **ROOT** icon, right-click, and select **Add Into > Feedback Section**.



16. Highlight the **Feedback Section** icon and position the cursor in the Page Editor window.
17. Type “You are currently using the following browser:”
18. Click **Insert Dynamic Element** on the Dynamic Page toolbar. The **Dynamic Elements** dialog box appears.
19. Select and double-click “**v_browser.**” This parameter appears in Page Editor.
20. Close the dialog box.

You can now upload the site to your server and test it.

Index

A

- Add a submit button 3-4
- Add data entries to an open data 3-4
- Assign a variable a value 5-5
- Assign "var_name" a value 5-5

C

- Cookie Attributes
 - Domain 16-9, 6-1
 - Expiration Date 16-9, 6-1
 - Path 16-9, 6-1
 - Secure 16-9, 6-1
- Create a Database 3-1
- Create a Link to a Download Form 5-3
- Creating a Dynamic Page 2-3, 3-2, 2-3, 5-4
- Creating a Dynamic Page with Send Mail 2-4
- Creating an Error Page 6-4
- Creating a Feedback Page 2-6, 3-7, 5-6, 6-5
- Creating a Field Validation Test 2-5, 3-6, 6-3
- Creating a Form 2-3, 3-3, 2-12, 6-2
- Creating a RecordSet 2-6
- Creating a Send Mail function 2-6
- Creating a Success Page 2-6, 3-8
- Creating an Error Page 2-6, 3-7
- Creating Dynamic Page Elements 3-5
- Creating Dynamic Text 2-7
- Creating Emails for a Visitor 5-6
- Creating Emails for the Webmaster 5-6
- Creating Navigation Links 2-9
- Creating navigation links for a results page 2-8
- Creating the Cookie 6-3

D

- define a parameter assign it a query value 2-4
- Defining Parameters for a Dynamic Page 3-5, 2-4
- Defining Parameters on a Dynamic Page 2-4
- Defining Variables for a Dynamic Page 3-5
- Defining Variables on a Dynamic Page 2-5
- Download Redirects 5-1

I

- Importing a Database into a Project 3-4, 2-2

L

- Link a form to a dynamic page 3-4

N

- No Result on Query 2-8

P

- Page Redirects 5-1

R

- Reading Cookies 6-5
- Redirecting a page to another page 5-2
- Remove data entries from an open data 3-4
- Requesting Cookies 6-5
- Requesting Information from the Server 6-6

S

- Send Mail folder 2-1
- set up an IF/ELSE condition 2-9
- Setting Up A Backwards Link 2-12
- Setting Up a Feedback Header 2-4
- Setting Up a Forward Link 2-10
- Setting Up a Query 2-5
- Setting up an OR query function 2-6
- Setting Up Forward and Backward Links 2-10
- Setting Up the SendMail Function 2-6

U

- Updating Your Database Using Dynamic Pages 3-2

V

- View data entries for an open data 3-4

W

- Working with the Query Function 2-4
- Writing the Cookie 6-4
- Writing to a Database 3-6