macromedia®

DRUMBEAT™ 2000 ACTIVE SERVER PAGES VERSION

Users Guide

macromedia

Trademarks

Apple Disclaimer

APPLE COMPUTER, INC. MAKES NO WARRANTIES, EITHER EXPRESS OR IMPLIED, REGARDING THE ENCLOSED COMPUTER SOFTWARE PACKAGE, ITS MERCHANTABILITY OR ITS FITNESS FOR ANY PARTICULAR PURPOSE. THE EXCLUSION OF IMPLIED WARRANTIES IS NOT PERMITTED BY SOME STATES. THE ABOVE EXCLUSION MAY NOT APPLY TO YOU. THIS WARRANTY PROVIDES YOU WITH SPECIFIC LEGAL RIGHTS. THERE MAY BE OTHER RIGHTS THAT YOU MAY HAVE WHICH VARY FROM STATE TO STATE.

Part Number ZBD10AA100

Acknowledgments

Authors: Natalie Calkins, Gayle Kidder
Contributing Writers: Stuart Harris, Deanna Bebb
Editor: Gayle Kidder
Technical Editors: Julie Thompson, Eric Ott

First Edition: August 1999

Macromedia, Inc.
600 Townsend St.
San Francisco, CA 94103

CONTENTS

INTRODUCTION

CHAPTER 1

CHAPTER 2

CHAPTER 3

CHAPTER 4

CHAPTER 5

CHAPTER 6

CHAPTER 7

CHAPTER 8

CHAPTER 9

CHAPTER 10

CHAPTER 11

CHAPTER 12

CHAPTER 13

CHAPTER 14

CHAPTER 15

CHAPTER 16

CHAPTER 17

CHAPTER 18

CHAPTER 19

CHAPTER 20

CHAPTER 21

INTRODUCTION
Getting Started

Drumbeat 2000™ is the fastest, easiest way to build Web interfaces to databases and applications that take full advantage of Active Server Pages (ASP)—and automatically work in different browsers. Without manual coding, you can build powerful intranet and Internet applications that serve customers, streamline business transactions, and help employees work more efficiently.

Drumbeat is designed for Web-site builders and application developers who need to speed the deployment of departmental-level, database-driven Web applications that access and update data in real-time. With Drumbeat, you can:

- Convert existing applications and enterprise databases into Web applications.
- Assemble ASP applications with reusable Web components.
- Create engaging Web applications quickly.
- Build sites that automatically work in different browsers
- Manage all your content from one convenient location.

Key Drumbeat features

Drumbeat contains these key features that you'll find yourself using frequently.

DataForm wizard defines pages that can search, update, insert and delete records in ODBC-compatible databases. When creating dynamic pages in Drumbeat, you see the page in the design layout exactly as your user will see it in their browser. You use simple drag and drop layout techniques to create powerful database-driven applications and easily customize the data display.

You can use the DataForm wizard to create Active Server Pages that include a full range of SmartElements without writing a line of code. Create drop-down lists, checkboxes and radio buttons, as well as edit boxes, within the wizard. A new Data Loop component lets you display multiple records on a page.

SmartElements let you can drag-and-drop Web components such as HTML and DHTML tags, Java applets, ActiveX controls, and scriptlets. As you define pages in Drumbeat, you position SmartElements on the layout, exactly where you want them to appear in your visitor's browser. SmartElements know how they can interact with each other in each browser type and function accordingly with Point-and-Click Interactions.

Point-and-click interactions create engaging, interactive, database-driven Web sites without programming or coding. For any combination of SmartElements, Drumbeat creates a popup list of the appropriate interactions among those elements. Using simple Point-and-Click interactions, you select how multiple elements interact with each other to provide multimedia effects, streaming media controllers, forms validation, and database queries. Drumbeat automatically generates the code.

SmartPages let you build sites that automatically adapt to multiple browsers. With browser-sensitive SmartPages, Drumbeat users build sites that take advantage of the most advanced browser features, and automatically adapt to support older browsers

PageSets simplify the task of creating and maintaining groups of pages that have a common layout.

Layout inheritance defines new pages or templates by changing only what is different, while maintaining a subordinate relationship to parent design templates.

Asset Center serves as a central source for collecting, managing, viewing and assigning site-wide assets, including media, queries, links, applets and other components.

One-button publishing generates sites on production or staging servers from within Drumbeat application without manually using FTP.

Open lets you import static legacy sites and turn them into active, database-driven business solutions.

Skill-based collaboration lets you create applications that people throughout the organization can contribute to as an ongoing project.

Drumbeat Extensibility supports an open environment for Web application assembly, enabling integration of a wide variety of additional functionality via standard and custom components.

SQL Query builder lets you import views and queries that have been constructed in other ODBC database applications so that you don't have to recreate your queries in Drumbeat. You can even use stored procedures that execute SQL logic on the server, and dynamic queries to execute SQL logic at run time with real-time variables.

VBScript Support enables ASP for your site, you can choose whether to use JavaScript or VBScript for server-side scripting. The Scripting Center makes writing scripts in either language easy and convenient, with a scripting tree that contains all the objects, methods, properties and events you need.

Interactions Center lets you view, filter and apply interactions to your Web applications easily from an expanded window above the Layout window. Server-side COM and ASP objects now can be included in point-and-click interactions.

Site creation and publishing wizards create a new site and set up publishing options using wizards that walk you through all the necessary choices.

Linking options let you can link within AutoTable columns, PageSets, and Recordsets.

System requirements

The following hardware and software are required to run Drumbeat:

- An Intel Pentium® 133 MHz processor (200 MHz recommended) running Microsoft Windows™ 95, 98, or NT version 4.0 or later
- 30MB of available disk space (60MB recommended)
- 32MB of RAM (64MB recommended)
- Microsoft Internet Explorer 4.0 or above (required to use the Preview and Import features; IE 4.01 is required for Personal Web Server)
- Microsoft Internet Information Server 3.0 or above for Windows NT, or Chili!Soft or other ASP interpretive software for other servers. Required to create Active Server Pages, but not to use Drumbeat.
- Personal Web Server (PWS) for Windows NT or Windows 95 to build and view database-driven applications on your own machine. (PWS is included in Windows 98.)

PWS is not necessary if you are publishing directly to IIS or an ASP-enabled web server. Personal Web Server is free from the Microsoft web site at: www.microsoft.com/windows/ie/pws/default.htm

Note: The Personal Web Server that comes with FrontPage 98 is not the correct version. You must obtain the full version from the Microsoft web site.

Installing Drumbeat

Follow these steps to install Drumbeat.

To install Drumbeat:

1 Insert the Drumbeat 2000 CD into the CD-ROM drive. The Drumbeat installation program starts automatically.

2 From the selections available, choose Install Drumbeat.

3 Follow the directions on the installation screen. This includes selecting a directory for installation. If you are a previous user of Drumbeat 2.0, it is recommended that you install Drumbeat 2000 in a separate directory (this is the default installation) and not overwrite your Drumbeat 2.0 installation. Please see the following information about Upgrading to Drumbeat 2000.

After installing Drumbeat, it is strongly recommended that you go through the exercises in the Quick Start guide to familiarize yourself with Drumbeat and its many powerful features before attempting to build your first web site.

Note: In order to properly access your databases for creating Active Server Pages in Drumbeat you need to have up-to-date database drivers (version 3.51 or higher). Drumbeat 2000 includes the correct database drivers in its installation program and automatically installs them for you if you have older drivers. This may occasionally create a conflict with some older programs that require older driver versions. See the Installation Notes for more information if you suspect conflicts with your database drivers.

Upgrading to Drumbeat 2000 from Drumbeat 2.0

The Drumbeat 2000 installation creates a separate directory from Drumbeat 2.0 and does not overwrite your previous version. You should not uninstall Drumbeat 2.0 before the installation is complete. During installation, you will be given the option to migrate your old global database to the new version of Drumbeat. It is recommended you accept this option. The global database (global.dgd) contains any custom contracts, new registered SmartElements, global styles, global queries, contracts, global JavaScript routines, and global media paths that you may have created in Drumbeat 2.0 for use in your sites. If you uninstall or overwrite your previous installation, they will be lost.

When you open sites developed in Drumbeat 2.0 in Drumbeat 2000, the files and any site-specific elements will be also be converted to the new format.

Guide to instructional media

Using Drumbeat is both a detailed reference and a step-by-step guide for new users of Drumbeat. After installing Drumbeat, see the *Drumbeat Quick Start* for a tutorial that will familiarize you with the Drumbeat workflow.

The documentation assumes you have a basic knowledge of Web page development. While you do not have to be fluent in HTML or any other Web-related language, you should have a general understanding of the logic behind generating Web pages and a basic understanding of Web processes.

Drumbeat Support Center

The Drumbeat Support Center Web site is updated regularly with the latest information on Drumbeat, plus advice from expert users, advanced topics, examples, tips, and updates. Check the site often for the latest news on Drumbeat and how to get the most out of the program at www.macromedia.com/support/drumbeat/.

CHAPTER 1
Drumbeat Basics

Welcome to Drumbeat, the easiest, fastest way to create dynamic, database-driven Web sites and to manage all your web site needs.

This chapter introduces the Drumbeat interface. Read it to familiarize yourself with the interface, terminology, and basic commands before you start creating your first Web site.

About Drumbeat files

Drumbeat files are a complete Web site resource. Drumbeat manages all the assets you use on your site—pages, references to images and other media files, links, database components, scripts, style sheets—in short, everything you use to create your site.

When you're working in Drumbeat, all your resources are at your fingertips, no matter where they reside on your computer or network. So are a wide array of point-and-click interactions and scripting resources.

When the time comes to publish, Drumbeat knows what components you need and takes care of almost all the work of getting them to the right place on your web server. You can easily integrate databases and other objects that already exist on your system wherever they reside.

Where to find your Drumbeat files

Drumbeat files have the extension EDF. By default, Drumbeat saves these files in the Drumbeat\My Sites directory. You can change the default directory for your site files or save any individual file wherever you specify.

To change the default site directory:

1 Select File > Preferences > Drumbeat.

2 Enter the path for your default sites directory, or browse to locate the desired directory.

Where to find your published files

Drumbeat accesses all other resources for your web site through the paths you provide when you add new media folders to your project. Drumbeat also takes care of posting them to the right directories on the server, as specified in your Publish Settings. You can switch between two sets of Publish Settings for any site, Staging or Production, so you can easily test sites before "going live."

To find the Publish Settings:

1 Choose Publish > Staging or Publish > Production to select the settings you want to set up.

2 Select Publish > Publish Settings.

Browser support

Drumbeat includes built-in support for multiple browsers. When you set up your site in Drumbeat, you choose the initial target browser for the site. When you want to add support for other browsers, just turn any page into a SmartPage and create the versions you want automatically. Each browser-specific page offers you only those features that are supported by that browser. Drumbeat generates scripts appropriate for that browser. When you create a SmartPage, all the redirect scripts are written for you, so that visitors to your site will be directed to the appropriate page for their browser.

The table below shows the features supported for each browser group and the type of code Drumbeat generates for each.

Browser Groups	Layout	Fonts and Styles
Generic Browser and Internet Explorer	Hidden table positioning	Font tags HTML 2.0 attributes
Netscape Navigator 3.x and greater	Hidden table positioning JavaScript	Font tags HTML 2.0 attributes
Both Internet Explorer 4 and Navigator 4.x	CSS Positioning and Netscape Layers Hidden table positioning of form elements JavaScript and DHTML	Cascading Style Sheets (CSS 1)
Internet Explorer 4.0 and greater	CSS Positioning JavaScript and DHTML	Cascading Style Sheets (CSS 2)—HTML 3.0

Hidden HTML tables are used for positioning in older browsers and are automatically generated by Drumbeat when publishing pages targeted at older browsers. They are not available as a design element in Drumbeat. (With a little experience, you will discover that CSS positioning will accomplish everything you may have used tables for, and a lot more.)

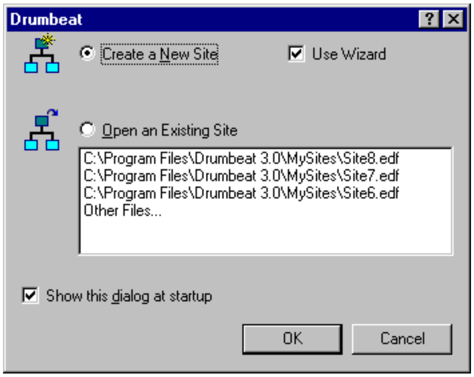
Creating a new site with the Site wizard

When you launch Drumbeat, the start-up dialog gives you the option to create a new site or open an existing site. You can also import a site through the New Site wizard, or import pages into an open Drumbeat file later. The Import feature is a powerful and convenient option if you have an existing site you'd like to bring into Drumbeat to manage.

To create a new site with the Site wizard:

1 From the Drumbeat start-up dialog, select Create a New Site.

2 Select Use Wizard and click OK.



Drumbeat

● Create a New Site ☑ Use Wizard

○ Open an Existing Site

C:\Program Files\Drumbeat 3.0\MySites\Site8.edf
C:\Program Files\Drumbeat 3.0\MySites\Site7.edf
C:\Program Files\Drumbeat 3.0\MySites\Site6.edf
Other Files...

☑ Show this dialog at startup

The New Site wizard walks you through the choices you need to make to set up your site. All the choices you make in the wizard can be changed later in Drumbeat, if necessary, so you can safely make temporary decisions if you are not sure of your needs yet.

3 Follow the instructions on screen.

Publish Settings wizard

You can choose the Publish Settings wizard from the New Site wizard or open the Publish Settings wizard within Drumbeat (Publish > Publish Settings) when you are ready to publish. The Publish Settings wizard helps you set options required for publishing a site to a staging server for testing or to a production server for release. All options you set can be changed at a later time. See Chapter 8, "Publishing."

Using Starting Points

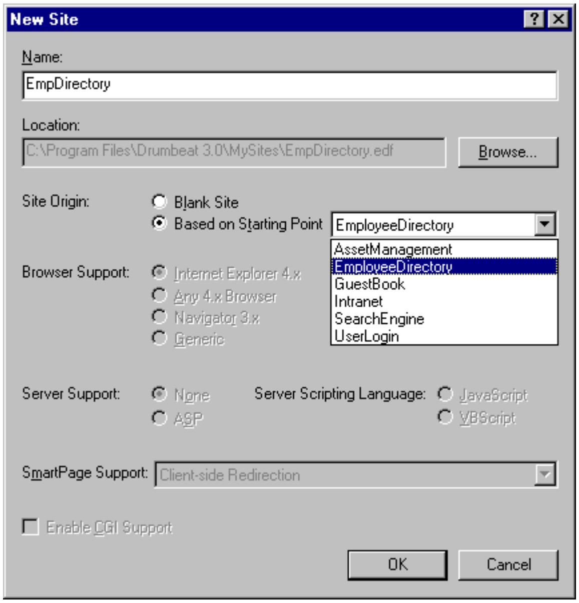
Drumbeat includes a number of Starting Point sites that you can use as a starter for your own web site. The Starting Points included in the installation are complete web site starters that include databases, custom contracts, sample pages and media. The DSNs for the databases will be set up for you during the installation process. You can also find new Starting Points on the Drumbeat Web site (macromedia.com/software/drumbeat).

You can create your own Starting Points that contain all the features and standard settings you want in your web sites. This can include Publish Settings, templates, media files and folders, styles, custom scripts, databases and any number of special elements.

To open a site based on a Starting Point:

1 Do one of the following:

- In the New Site wizard, select Drumbeat Starting Point in the Site Origin dialog, and then select the Starting Point you want to use.
- From within Drumbeat, click the New Web Site button or select File > New. In the New Site dialog, for Site Origin click the Based on Starting Point radio button and select a Starting Point from the drop-down list.



New Site

Name: EmpDirectory

Location: C:\Program Files\Drumbeat 3.0\MySites\EmpDirectory.edf

Site Origin: ○ Blank Site ● Based on Starting Point: EmployeeDirectory

AssetManagement
EmployeeDirectory
GuestBook
Intranet
SearchEngine
UserLogin

Browser Support: ● Internet Explorer 4.x ○ Any 4.x Browser ○ Navigator 3.x ○ Generic

Server Support: ● None ○ ASP

Server Scripting Language: ○ JavaScript ○ VBScript

SmartPage Support: Client-side Redirection

□ Enable CGI Support

To create a Starting Point:

1. Create a new site and customize it with all the features you want.
2. Save a Snapshot (File > Snapshot) in the directory C:\Program Files\Drumbeat 2000\StartingPoints (or the StartingPoints directory of Drumbeat where you installed it). Give it an appropriate name (that will show in the Starting Point selection list).

The next time you open a new site, your new Starting Point will appear in the Starting Points list.

Naming and renaming a site

When you create a new site, either when Drumbeat is launched or by selecting File > New in Drumbeat, you can name your site as you wish in the New Site dialog.

Once a site has been created, you must rename the site through Windows Explorer.

To rename a site:

1. Make sure the site file is not open or in use.
2. Open Windows Explorer from your Windows 95/98 or NT system.
3. Browse to the location of the file.
4. Using any standard Windows method, change the name of the file, ensuring that it keeps the .edf extension.

Saving files

Drumbeat automatically saves updates to your file each time you switch pages, publish or close the file. There is no need to manually save the file.

You can force a save function with the Snapshot feature. Site Snapshot enables you to save a copy of the site file in its current state under a different name. You can then continue to work on the site or switch to the Snapshot. This is useful if you are making a major change and are uncertain of the effects your changes will have, or if you want to retain the ability to return to the previous state should you change your mind.

Creating snapshots

Whenever you want to save a current copy of your site as a backup you can use the Snapshot feature. Snapshot provides you with an easy way to save a copy of your site in its current state. This is especially useful when you are making major alterations to a site and you want to revert to an earlier version if changes don't work out.

The Snapshot function saves the current site information under a different site name, leaving the current site open for further modification.

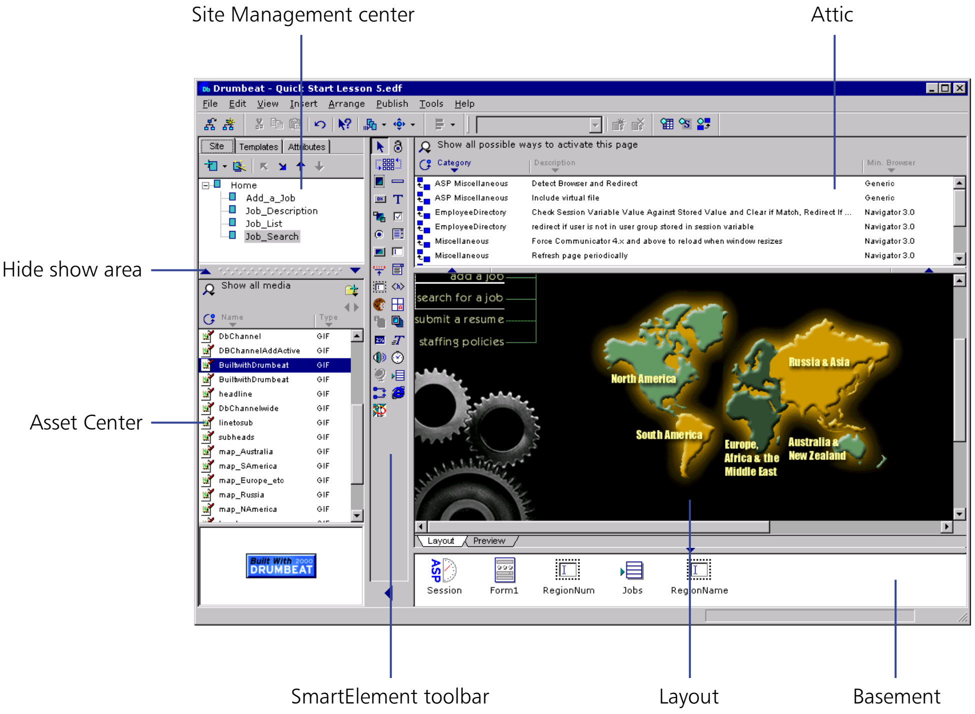
To save a Snapshot of a Drumbeat file:

1 From the File Menu, select File > Snapshot.

2 Enter a File name, and click Save.

The Drumbeat workspace

The main areas of the Drumbeat interface and the components in each are shown below:



Category	Description	Min. Browser
ASP Miscellaneous	Detect Browser and Redirect	Generic
ASP Miscellaneous	Include virtual file	Generic
EmployeeDirectory	Check Session Variable Value Against Stored Value and Clear if Match, Redirect If ...	Navigator 3.0
EmployeeDirectory	redirect if user is not in user group stored in session variable	Navigator 3.0
Miscellaneous	Force Communicator 4.x and above to reload when window resizes	Navigator 3.0
Miscellaneous	Refresh page periodically	Navigator 3.0

Name	Type
DbChannel	GIF
DBChannelAddActive	GIF
BuiltwithDrumbeat	GIF
BuiltwithDrumbeat	GIF
headline	GIF
DbChannelwide	GIF
linetosub	GIF
subheads	GIF
map_Australia	GIF
map_SAmerica	GIF
map_Europe_etc	GIF
map_Russia	GIF
map_NAmerica	GIF

Major Drumbeat interface areas and their functions.

Site Management Center	Site Manager: Displays site tree showing logical page hierarchy. Template Manager: Displays design templates and pages according to the template hierarchy. Attributes tab: Displays and allows changes to the attributes of any selected page, template or element.
Asset Center	Displays all site assets and allows querying and filtering of assets by type and special criteria.
SmartElements toolbar	Customizable toolbar displays commonly used SmartElements for easy drag-and-drop access. View all SmartElements in the SmartElement Library or by Asset Center query.
Layout	Layout: Environment in which page elements can be dragged and positioned. Preview: Allows previewing of a site within Drumbeat using a modified, built-in Internet Explore browser. (Installs only if IE is found on the system.) Previewing is also available using external browsers.
Attic: Content, Interactions & Script Center	Content Center: Displays content tables of three types: 1) Manually created content tables 2) Tables created from a CSV file 3) ODBC content tables based on a SQL query of a database. Interactions Center: Displays point-and-click interactions that apply to selected elements on the layout. Interactions are also filterable by various criteria. Script Center: Allows scripting by JavaScript or VBScript using the scripting tree or by hand.
Basement	Displays representations of non-visual elements used on the layout such as database components and hidden form elements.

Using toolbars

The Drumbeat workspace can be customized to display or hide any of the available toolbars. Toolbars can be made to float over the interface or be docked either above or below the main workspace.

To display or hide a toolbar:

Select View > Toolbars > toolbar name.

Checked toolbars will be displayed. Unchecked toolbars are hidden. Selecting a toolbar from the menu toggles its display. Hidden toolbars, such as the text formatting toolbar, will pop up as floating when the function is required.

To float a toolbar:

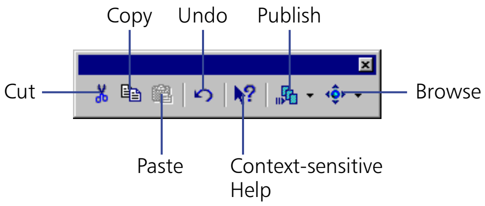
1. Click one of the gray dividers on a toolbar to select it.
2. Hold the mouse key down and drag the toolbar to the desired location on your screen.

To dock a floating toolbar:

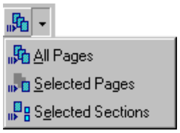
Drag the floating toolbar from the title bar to the desired location at the top or bottom of the screen and release the mouse button. .

Standard toolbar

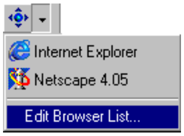
The Standard toolbar contains many commonly used functions while creating a site.



The Publish button has options to select the extent of the publish operation. The Browse button has options to choose the browser in which to preview the site.



Drop-down publish options.



Drop-down browsing options.

View toolbar

The View toolbar allows you display the Content Center, Interactions Center or Script Center in the space above the layout.

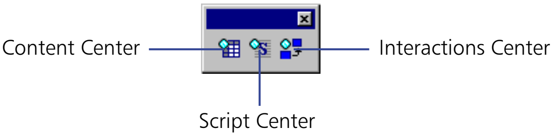


Image Map toolbar

The Image Map toolbar lets you create hot spots of different shapes. The Image Map toolbar is hidden by default until you need to use it. However, you can add it to your toolbar if you want by choosing View > Toolbars > Image Map.

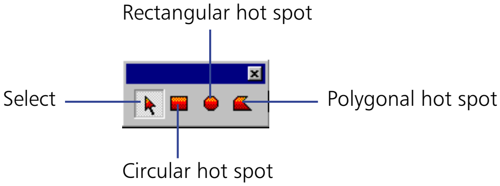
To activate the Image Map toolbar:

1 Drag the Image Map SmartElement onto the layout from the SmartElement toolbar.

2 Drag an image file from Asset Center onto the image map on the layout.

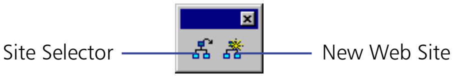
3 Click the center of the image and the Image Map toolbar will pop up.

The Image Map toobar buttons allow you to draw hot spots of different shapes that can be linked to different files or URLs for navigation.



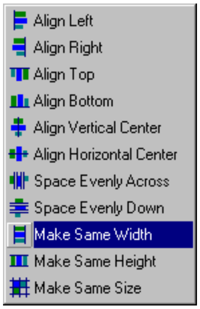
Site toolbar

The Site toolbar buttons are used to open an existing site or create a new site. The New Web Site button opens the New Site dialog.



Arrange toolbar

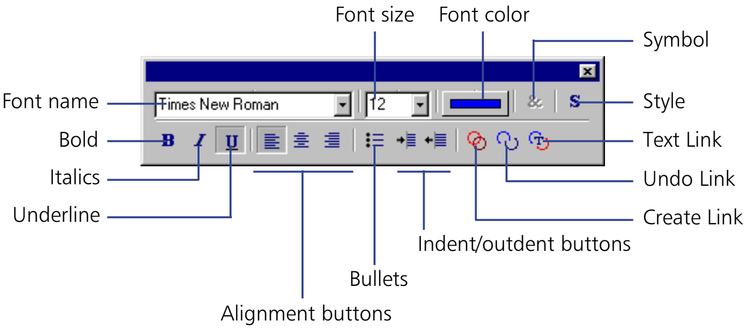
The Arrange button is used to select an option to automatically align multiple elements on the layout.



Text Formatting toolbar

This toolbar contains all of the functions for modifying the formatting of any text you place on your site within a Text SmartElement, such as a text element, Scrolling Text, or Edit Box. The Text Formatting toolbar is hidden by default and pops up when you begin inline editing of a text element. However, you can add it to your toolbar by choosing View > Toolbars > Text Formatting.

If the toolbar is hidden, it will pop into view whenever you click a Text SmartElement or text cell of a content table for editing.



Format text inline with the Text Formatting toolbar.

2

CHAPTER 2
Site Management

Web site management tasks generally include several dimensions. Creating an efficient management structure for your site includes:

- Creating a logical structure on which to base navigation.
- Managing design tasks efficiently.
- Managing resources used repeatedly in a way that will cut down the work involved in changes.
- Managing external resources such as databases that form the core content of your site.

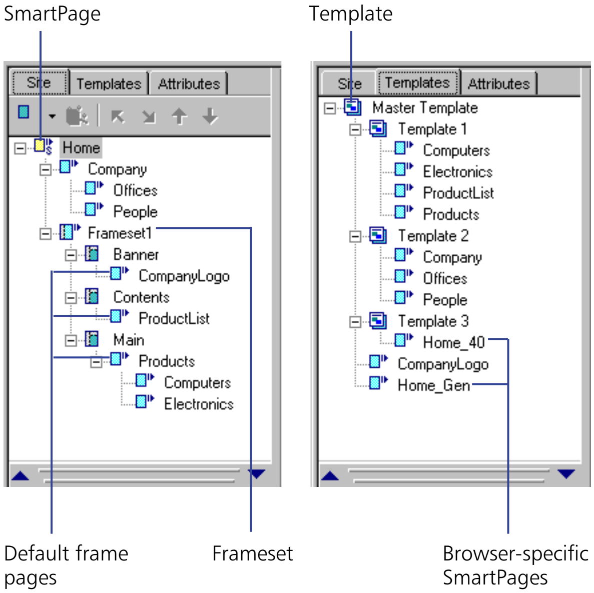
This chapter discusses the way that Drumbeat helps you manage the first two tasks: organizing your site and managing design tasks. Drumbeat takes a dual approach to site management that allows Web authors to work in the way most natural to them and to focus on the task at hand in the most efficient manner.

Understanding the site tree

Drumbeat offers two different, complementary views of your site in Site Manager and Template Manager. While many functions are identical in the two views, the purpose of each is different. Site Manager is the logical center of your site, while Template Manager is the design center.

In Site Manager the site tree displays the structure of your site according to a hierarchy you create much as you would an outline. Site Manager displays standard pages (including DataForms), PageSets, framesets and SmartPages. SmartPages are represented as a single page, since for structural purposes multiple browser versions of a page can be treated as one page. External pages that you choose to make part of your site structure for organizational purposes will also be shown in Site Manager.

Template Manager shows the design hierarchy of your site. Pages, Dataforms and PageSets are displayed the same as in Site Manager. Framesets do not display in Template Manager (since a frameset does not inherently have a design). SmartPages are shown as individual browser-specific pages, since design-wise these may be treated very differently and may be put under different templates. External pages are not shown in Template Manager.



Two views of the same site: Site Manager (left) and Template Manager (right).

While you can do most functions in either Site Manager or Template Manager, there are three cases in which you must work in Site Manager:

- To create or work with framesets.
- To create SmartPages or work with SmartPages as a group.
- To add an External Page.

There are two cases in which you must work in Template Manager:

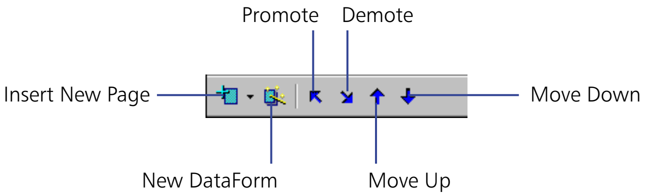
- To create or work with design templates.
- To work with individual browser-specific SmartPages.

Using Site Manager to organize your site

The primary functions of Site Manager are to visualize and organize site structure, and to create relational links (links such as previous and next page). Other Site Manager functions include adding, rearranging, and deleting pages, assigning pages to framesets, creating SmartPages for browser redirection, and managing External Pages.

Blank sites open with a single page, called Home. The Home page can be renamed, but cannot be deleted, copied or moved.

The Site Manager toolbar at the top of Site Manager provides several key functions for use in Site Manager.



Site Manager toolbar.

Site hierarchy

Pages can be arranged in collapsing and expanding hierarchies, just as in Windows Explorer. However, the site hierarchy has no meaning as far as inherited elements in Site Manager. In Site Manager you can assign elements freely to individual pages.

The one circumstance in which the site hierarchy does convey meaning is that the site hierarchy dictates the way relative links such as next and previous page will operate.

Relative linking in the site tree

Relative links between pages within the site tree can be created using power links. These links adhere to the structural logic of your site and maintain their integrity even as you move pages around.

To create a relative link to the next or previous page:

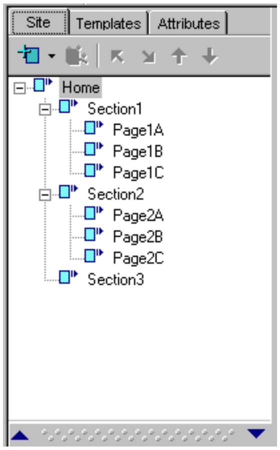
1 Do one of the following:

- Select the anchor element (image or text element). In the Attributes tab, click the Link Assign button.
- Within a text box, select the text and click the Assign Link button on the Text Formatting toolbar, or right-click and select Assign Link.

2 In the Link dialog box, select Previous Page or Next Page.

A link is created to the next page in the site tree at the same level. If you move pages around in the site structure, links will be remade to reflect the new next page or previous page, as shown in the Site Manager site tree.

Next page and previous page links are only active between pages that share a common ancestor and are within the same level in the site tree.



Here, a next page link on Page1A will go to Page1B. A next page link on Section 1 will go to Section 2. A next page link on Page 1C will be inactive, as there are no more pages within that sibling group.

Moving pages

By moving pages within Site Manager, you can reorganize the site structure and remake the relative linking system. Groups of pages in the hierarchy can be moved together. Subordinate (child) pages in the site tree will be moved along with the parent page.

To move a page, along with all subordinate pages:

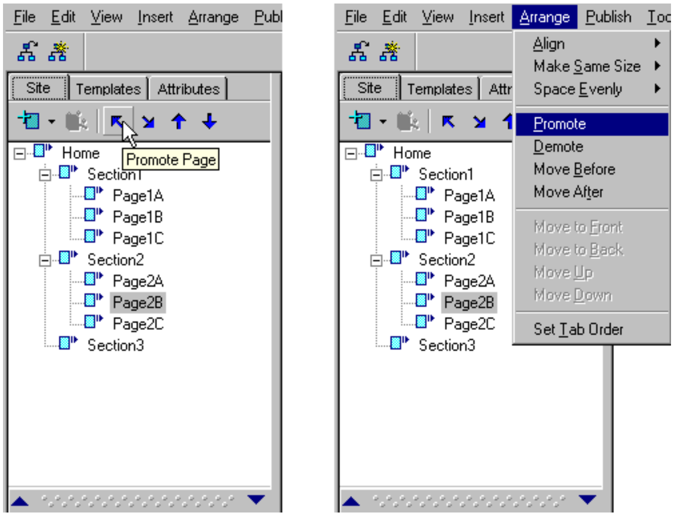
Do one of the following:

- Drag the page to the desired location.
- Select the page and click the up or down arrow on the Site Manager toolbar.
- Use Edit > Cut or Ctrl-X to cut the page and Edit > Paste or Ctrl-V to paste it to a new location.

To promote or demote a page to a new level in the site tree:

Do one of the following:

- Select the page and click the Promote Page or Demote Page arrow button.
- Choose Arrange > Promote Page or Demote Page.



In this example, Page2B will be promoted to the same level as its parent page, Section2.

Using Template Manager to streamline design

Template Manager shows the way design elements are shared between pages. Using templates helps minimize design time and facilitates rapid changes of common page elements such as navigation buttons, logos, and page backgrounds.

Blank sites open with one template, the Master Template. The Master Template cannot be deleted or renamed. New templates will always be subordinate to the Master Template. New templates can also be created subordinate to existing templates and will become part of the cascading inheritance hierarchy. Items that can be inherited from a template include all SmartElements, all associated contracts, links, styles, background colors, background images, and link styles.

The site tree in Template Manager shows pages assigned to each template, ordered alphabetically under each template.

To create a new template:

1 Select the Master Template, or any existing template under which you want the new template to appear.

2 Right-click and choose Insert > Template, or choose Insert > Template from the main menu.

A new template will assume a default name such as Template 1. You can rename the template as you wish.

To rename a template:

1 Do one of the following:

- Select a template and press F2.
- Right-click a template and choose Rename.

2 Type the new name of the page and press Enter.

Converting a page to a template

Any existing page can be converted into a template. This can save design time if you have already spent a considerable amount of time fine-tuning the layout of a single page and would like to reuse all or part of the design on other pages.

To convert a page to a design template:

1 In Template Manager, select the page you want to convert.

2 Right-click and choose Create Template From Page.

The new template will be created immediately above the original page. All of the elements on the original page will be transferred to the template. You can then delete any elements you do not want to appear as inheritable elements on the template, or cut and paste them back to the original page.

Assigning pages to templates

You can assign pages to templates so that those pages contain the same design characteristics as the template, or unassign them, to make them independently formatted. (The exception is the Master Template, whose attributes will be inherited by all subsequent pages and templates).

A page or template moved to a new parent template in Template Manager takes on all the elements and styles from the new parent template, plus any individual elements and styles that it had previously. It will lose any elements it may have inherited from the previous parent.

To assign pages and other templates to a parent template:

Drag the page or template onto the desired template in the Template Manager.

To remove all template elements from a page:

Drag the page onto the Master Template.

Only those elements on the Master Template will remain. If the Master Template is blank, no elements will be inherited.

Template attributes

Templates have attributes similar to pages. Attributes assigned to a template will be inherited by all pages under that template.

Template attributes include background image, background color, fixed background, link style (unvisited links), visited link style, active link style, and hover link style (mouse-over).

See “Page attributes” on page 39 for more explanation on how to change individual page attributes.

To view and change inherited attributes:

1. Select the page in Site Manager or Template Manager so it is active in the layout. (If it is already selected, click an empty place in the layout to deselect any currently selected elements.)
2. Click the Attributes tab or right-click on an empty spot in the layout and choose Attributes.

   Currently assigned attributes will be displayed in the Attributes tab. To change any of them, you must first cancel any inheritance from higher templates.
3. In the Attributes tab for the template, click the Inherit button.
4. Deselect the attributes you want to assign individually. Select those that you want to assign through inheritance from another template.

Meta tags and template inheritance

Meta tags can also be added to a template and inherited by subsequent pages. Meta tag information inherited from a template can be added to at the page level, but cannot be changed.

To add meta tags to a template:

1. In the Attributes tab for the template, click Meta Tags Assign.
2. In the Header Content dialog box, enter the meta tag information (in HTML format).

For more information, see “Meta tags” on page 40.

Template hierarchy and inheritance

The Master Template is the root node of the Template Manager. Regardless of any other template assignment, all pages and templates are under the control of the Master Template, which may contain nothing at all or may contain site-wide items that affect all pages in the site. The Master Template transfers all of its attributes to subsequent pages and templates at publish time. It cannot be deleted and its name cannot be changed.

In general, you will only want to place content on a design template if you want that content to appear on every subordinate page. You can also use the Master Template to place guides that will assist in placing content on pages in your site. Guides are inherited along with other content from templates (although they do not publish with the page), so they may be used to control page size and assist in precise placement of page elements.

Templates are subordinate to other templates in a design hierarchy, so that one template can be based on another, but can contain additional elements. Any template can be used as the basis for yet another template, through a descending level of layout inheritance. Elements inherited from a template will appear with a hatched pattern on the page layout to distinguish them and cannot be edited. To change their position or contents, you must make the changes on the template. Templates are designed in Layout mode, exactly as are pages.

Any time a change is made to a design template at any level, all pages and templates subordinate to that template will reflect that change when displayed in the layout or in the browser when republished. Changes to a template will cause subordinate pages and templates to be flagged for republishing.

Although inherited elements from a template layout cannot be changed, there is an exception for the document properties that include background image, background color and link colors. You can choose whether a page or template inherits any of these properties from the template by deselecting them in the Inherit Attributes dialog box.

Besides visible design elements, you can also add nonvisible SmartElements, such as interactions, audio files and recordset elements to a template. When you add a nonvisible SmartElement, the SmartElement is automatically placed in the Basement of the template. Protected objects in the Basement are dimmed to indicate that they cannot be modified. (Open or close the Basement by clicking on the arrow indicator on the horizontal splitbar.)

To disable inheritance of document-level attributes from a template:

1 Select the page in Site Manager or Template Manager and click the Attributes tab. (If the page is already in layout, click an empty spot on the layout.)

2 Click the Inheritance Assign button on the Attributes tab.

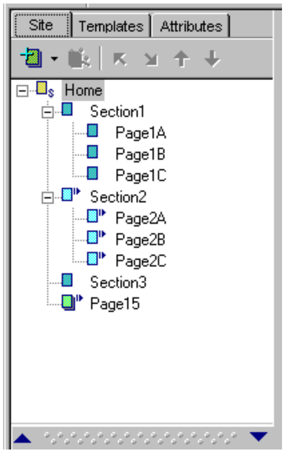
3 In the Inherited Attributes dialog box, deselect those properties that you do not want the current page or template to inherit from the parent template.

Creating Drumbeat pages and setting their properties

Pages in Drumbeat can be of several types. Pages that have different functionality are represented with different icons in Site Manager and Template Manager, as shown.

▨	Standard Pages
▨ S	SmartPages
▨	PageSets
▨	DataForm Pages
▨	External Pages (do not publish) External Page (publish)

In addition, published pages are represented with a darker icon. Unpublished pages are flagged in the site tree.



Adding pages

Pages can be added either in Site Manager or in Template Manager.

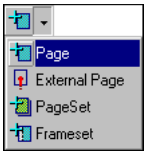
- Use Site Manager if you want to add a page at a specific point in the site tree, to keep the outline order you have established. (In Template Manager the page will appear at the bottom of the site tree, subordinate only to the Master Template; you can then drag it under any template.)
- Use Template Manager if you want to add a page to a design template so that it will instantly inherit design elements. (In Site Manager, the page will appear at the bottom of the site tree; you can then move it within the site tree as desired.)

To add a page in Site Manager:

1 Select the page under which you want the new page to appear in the site tree.

2 Do one of the following:

- Click the Insert Page button on the Site Manager toolbar or click the down-arrow on the button to select the type of page you want to insert.



- Right-click a selected page and choose Insert > Page.
- Choose Insert > Page.

To add a page in Template Manager:

1 Select the template under which you want it to appear in the design tree.

2 Right-click the selected page and choose Insert > Page, or choose Insert > Page.

Renaming pages

Pages assume initial default names (Page1, Page2, and so on) when created. You can rename pages to reflect the content and facilitate identification. Each page must have a unique name.

The page name you assign in Drumbeat will be used as the file name for the HTML file, if you are publishing with that file type. Spaces and special characters (including punctuation) should not be used in file names, as this may cause publishing problems with Netscape browsers. You can use underscores (Products_and_Services) or embedded capital letters (ProductsAndServices) to increase the legibility of page names.

The Drumbeat page name will also be used for the page title for the page unless you specify a different page title in the Web Page Properties dialog box for the page. The title is visible in the title bar at the top of the browser and is often used by search engines to catalog a page, so you may want to assign a different published title for the page. Unlike the page name, titles can contain spaces and punctuation.

To rename a page:

1. Select the page that you want to rename.
2. Click on the page name a second time or right-click the page and select Rename, or press the F2 key.
3. Type the new name of the page and press Enter.

To assign a title other than the page name used in Drumbeat:

1. Select the page in the site tree, right-click and choose Properties.
2. In the Publish tab, enter a page title in the edit box.

Page attributes

Pages have specific attributes that can be assigned individually or through the templates to which they belong.

Page attributes include the following:

- Background image used as the page background.
- Background color.
- Fixed background that determines whether a background image is fixed (nonscrolling) or scrolls with the page.
- Link Style includes font, font size and color of text links and border, if any, around links (anchor elements).
- Visited Link Style includes font, font size and color of text links and border, if any, around visited link elements.
- Active Link Style includes font, font size and color of text links and border, if any, around active link elements (as it is clicked).
- Hover Link Style includes font, font size and color of text links and border, if any, around link elements when the mouse is over it (Internet Explorer 4.x only).

To view and change page attributes:

1. Select the page in Site Manager or Template Manager so it is active in the layout. (If it is already selected, click an empty place in the layout to deselect any currently selected elements.)
2. Click the Attributes tab or right-click an empty spot in the layout and choose Attributes.

   Assigned attributes are displayed in the Attributes tab. Attributes that are inherited from a template are grayed out. To change any attribute, you must first cancel any inheritance from higher templates.
3. In the Attributes tab for the template, click the Inherit button.
4. Deselect the attributes that you want to assign individually. Select those that you want to assign through inheritance from a template.

   Once you have disabled inheritance from templates, you can assign those page attributes that are not inherited individually.

To set a background image:

1 Do one of the following:

- In the Asset Center, locate the image you want to use and Shift-drag it to the layout.
- In the Attributes tab for the page, select Bkgnd Image and click the ellipses (…) button next to the Bkgnd Content text box. In the Content dialog box, enter the path to the image or browse to locate it. To use an image from a content table, choose Static Content (for a manual content table) or Recordset (for content from an ODBC database) from the Bkgnd Content pop-up menu. In the Content dialog box, choose the row and column that contains the image reference.

See "Content Tables" on page 199 for more information about using content tables.

To select a background color:

1 Make sure Bkgnd Color is not selected in the Inherit Attributes dialog box.

2 In the Attributes tab, click the color swatch next to Bkgnd.

3 In the Color dialog box, select a color from the palette or click Define Custom Colors to select a color from the color spectrum or by RGB values.

To select link styles:

1 Make sure that the link colors you want to change are not selected in the Inherited Attributes dialog box.

2 In the Attributes tab, select a style from the pop-up menu for each type of link (Link, Visited Link, Active Link and Hover).

For information on how to create Styles, see "Using styles" on page 128.

Meta tags

Meta tags can be added to a page to convey extra information such as keywords and descriptions that may be used by search engines. Meta tag information can be inherited from a template or be added at the page level. Meta tags inherited from a template cannot be changed, but you may add new meta tags to the page.

To add meta tags to a page:

1 In the Attributes tab for the template, click the Meta Tags Assign button.

2 In the Header Content dialog box, enter the meta tag information (in HTML format).

Using external pages in your site structure

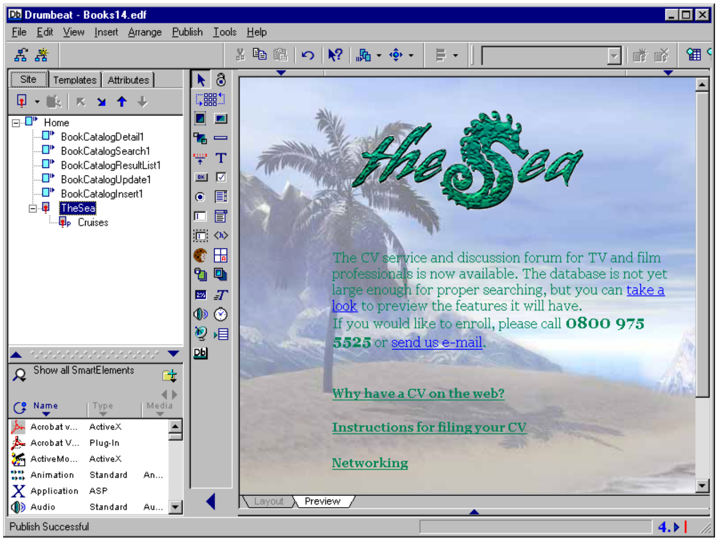
External pages are Web pages that you do not want to edit within Drumbeat, but would like to include in your site structure for navigational purposes. Bringing external pages into your Drumbeat site structure allows you to reference the page for organizational or navigational purposes, maintain the integrity of the link structure within the site, and maintain a page created or edited outside of Drumbeat without overwriting it.

Drumbeat can publish the external code intact for you during publishing. An external page that will be published by Drumbeat has a P next to the icon in the site tree.

	An external page that will not be published by Drumbeat may be either local or remote.
P	An external page that will be published by Drumbeat. (The flag next to the icon indicates that the page has not yet been published.)

External pages are only visible in Site Manager, not Template Manager. (Design is not controlled within Drumbeat). You may move the page within the site structure, and rename the page in the site tree. Renaming the page in Drumbeat will not affect the remote page.

Because external pages are not converted to the general format of Drumbeat, you may not make any modifications to them inside of Drumbeat. When the external page is selected, the page will be displayed in the Preview mode. Layout mode will be inaccessible for the page. You can edit the HTML code through any HTML editor. (See “Editing external pages” on page 44.)



The CV service and discussion forum for TV and film professionals is now available. The database is not yet large enough for proper searching, but you can take a look to preview the features it will have.
If you would like to enroll, please call **0800 975 5525** or send us e-mail.

Why have a CV on the web?

Instructions for filing your CV

Networking

Publish Successful

Select an external page and preview it in Preview mode in Drumbeat.

To add an external page:

1. In the Site Manager, select a page below which you want the page to appear in the site tree.
2. Right-click the page and choose Insert > External Page.
3. In the External Page dialog box, enter the URL of the external page or browse to the location. You may enter a complete URL or a relative URL, such as /directory/filename.html. See “Relative paths for external pages” on page 43.
4. If the site is local and you want to include the media, enter the URL or browse to the location of the media directory, and select Copy All Sub-Directories.
5. If you would like to have Drumbeat transfer the files to the publish directory when the site is published, select Publish This Page. Do not select it if the page is already published or you will publish it separately.

Note: If you do not select the Publish This Page option, no page components will be downloaded. A representation of the external page will be placed into your Site Structure for organizational and navigational purposes. You will be able to create page and relative links to it (but not from it). If you do choose this option, the page components will be downloaded and can be reused locally.

Relative paths for external pages

A relative path to an external page may have any of several acceptable formats.

Valid relative path formats include:

- filename.html (the file is in the current working directory)
- ./filename.html (same as above)
- directory/filename.html (the file is in the indicated directory, relative to the current working directory)
- ./directory/filename.html (same as above)

Editing external pages

Once an external page has been inserted into the Drumbeat site file, the code can be modified in an external editor. Since the page has not been converted to Drumbeat format, it cannot be modified within Drumbeat.

To modify the code of an external page:

1. In Site Manager, select the external page.
2. Right-click the external page and select Properties.
3. In the Web Page Properties dialog box, click Edit HTML. This will display the HTML code for the page in the HTML editor you have specified in Drumbeat Preferences (Windows Notepad is the default).
4. Edit the HTML and save the file.

   If the file is local, you will be able to view the modifications to the page in the Preview mode in Drumbeat. However, if you have chosen to have Drumbeat publish the page for you, the changes will not take effect on the remote site until you publish to the remote server.

To specify an HTML editor:

1. Choose File > Preferences > Drumbeat from the main menu.
2. In the Preferences dialog box, select the Content tab.
3. Specify your External HTML editor or browse to locate it. Then click OK.

Viewing external pages in an ASP site

Viewing external pages in an ASP site can sometimes be a problem, since the behavior of an ASP site is to always begin with the page you have designated as the home page in your Publish Settings. You can set up an external page to preview directly, without going through the default home page by selecting Preview Directly when you create the page or afterward in the Web Page Properties dialog box.

Publishing external pages

The publish engine in Drumbeat will not generate HTML for any pages that are designated as external pages. The purpose of publishing an external page is to copy the source files and any associated media to the publish destination. These files may then be used with other pages and can be viewed locally.

In order for Drumbeat to publish external pages (which may be created with other editing tools) you must select Publish This Page when the page is created, or select this option in the Web Page Properties dialog box.

To view or change the publish properties of an external page:

1 In Site Manager, select the page. Right-click and choose Properties.

2 Click the Publish tab.

If the Publish This Page option is selected, you can specify the destination directory, the published file name, and the media directory as you wish.

Importing pages and sites

An existing site can be imported into a new or existing Drumbeat file. During the import process, Drumbeat reads pages created in text editors or other tools and creates editable pages on which the graphics, text and components are positioned as closely as possible to the original. You can limit the content and scope of the files imported by limiting the number of linking levels and restricting the domains to import.

Because the import functionality is so thorough, importing large or complex sites may take some time. Start by importing a single page or limiting the number of levels to one or two to familiarize yourself with the process and the results before attempting to import an entire site.

Importing sites to Drumbeat

The Import function is available either from the Drumbeat New Site dialog box at launch or from within Drumbeat if you are working with an existing site.

To import a site into an open Drumbeat file:

1 Choose File > Import on the main menu or press Ctrl+I. (This option is only available when Site Manager is active.)

2 In the Page Name and Location box, specify the location of the site or sites to be imported. Choose local (Path) or Remote (URL). A local site is on a drive on your computer or network. A remote site is hosted in a location other than your local system or network, and is accessed from a URL. Enter the local path or the complete URL to the starting page or browse to locate and select it.

3 Choose whether to restrict the domains to import:

- Import This Domain restricts the import to pages that reside on the same domain as the root page. Any references to other domains will not be imported, but maintained as links.
- Import All Domains expands the import to include all links to other domains within the site. Pages on other domain names will be included in the import, as long as they are within the number of levels specified.
- Import Custom Domains allows you to specify exactly which domains you want to include in the import in a separate dialog box. This is useful if a company maintains several possible domains. Valid entries for the domain option include the following syntax:

```
http://www.drumbeat.com
www.drumbeat.com
drumbeat.com
iron.drumbeat.com
207.204.128.0
```

4 Limit the content to import only text, only images, only the site structure (all pages will be blank), or all the site content.

5 Choose whether to import all pages referenced in the site or restrict the number of linked pages to import.

- Choose 0 if you only want to import the current page and no linked pages.
- Choose 1 level to import that page plus any objects linked from that page.
- Choose 2 to import all pages linked from the second-level pages, and so on.

6 If you would like visual feedback on the status of your import, select View Import in Progress. For faster import, deselect this option.

7 Click Start Import to begin the import process. If you have chosen to view import in progress, the Import Progress dialog box will display information such as which page is currently being imported and which, if any, errors are encountered.

If any errors occur during importing, Drumbeat displays an error message indicating the general nature of the error. When importing is complete, click Log File to view a comprehensive error log for any errors incurred during the Import process.

8 Click Close to return to the Drumbeat interface and work with your newly imported pages.

Import site structure

Imported pages are shown as a series of sibling pages (a group of pages at the same hierarchy level) in Site Manager. Use the Page Arrangement tools from the Site Structure toolbar to move pages and organize the site as you wish.

If the imported site is a new Drumbeat site file, all pages (except those associated with a frameset) will be listed under the default Drumbeat home page. If the site import was initiated while working on an open site file, all imported pages will be listed as children of the page that was selected when the import was initiated.

Duplicate page names

If the import encounters a duplicate page name, the import function will modify the page name of all duplicate named pages after the first page. Additional pages will have a number value added sequentially to the page name and all page links will be updated to use the new page name.

Import limitations

The Drumbeat Import feature has the following limitations:

- Drumbeat Site Import cannot import the frameset structure itself. Imported framesets are shown as blank pages in Site Manager. Pages from framesets are imported, but not the frameset structure. You will have to re-create framesets and reassign the pages to the corresponding frames to match the imported site. See “Creating framesets” on page 256.
- Drumbeat Site Import does not retain attachments between scripts and objects. All scripts are imported and saved in a log file where you can use them to create custom contracts or paste them into the Script Center for assignment to an object.

  To open the script log, choose Tools > Logs > Import > Script Log.
- The Import settings for font size, color, and so on, assume your default browser settings for Internet Explorer 4.0 (if this is installed). For best results, follow the instructions below for setting fonts in your browser before importing any sites. If you do not set your browser to these settings, the font could be too large or small. Settings that are too large or too small may render the content inaccurately.

To change your Internet Explorer settings for optimal import:

1. In Internet Explorer 4.0, choose View > Internet Options.
2. Click the General tab and click the Fonts button.
3. In the Fonts dialog box, make these changes, as necessary:

- Proportional Font set to Arial.
- Fixed-width Font set to Courier New.
- Font Size set to Medium.

4. Click the Set as Default button, and then the OK button, which will return you to the Internet Options dialog box, and then click OK.

Import errors

Drumbeat creates two logs to record the import progress. Two files are created, Import.log and ImportScript.log.

To view the Import Error log:

Choose Tools > Logs > Status Log (or Ctrl+T) or Script Log (or Ctrl+R).

Drumbeat tracks the following information in the Import Status Log:

Tags not Supported by Drumbeat Import In the event that a tag is not supported by the import function of Drumbeat, the following line will appear in the Import Status log:

```
<Date/Time> <Imported File Name> Tag Not Supported
<HTML TAG THAT IS NOT SUPPORTED>
```

Inaccessible Files If Drumbeat attempts to import a site that has password-protected access, a login screen will appear for entering the username and password. If the username and password are accepted, the site will be imported. Otherwise, the following line will appear in the Import log:

```
<Date/Time> <Import File Name> is not available.
```

Missing Content If content for a tag is missing, the tag is converted to the appropriate empty SmartElement on the layout and an error message will be recorded in the Import Log.

Note: This may affect the layout positioning of the other elements on the page. Content may not appear in the exact positioning that was originally created.

Import Fails or Aborts Before Finishing If the import fails or aborts before the listed number of levels is imported, the log file will provide the following entry and include a list of all successfully imported files.

```
<Date/Time> <Import File Name> <Line Number> Drumbeat was not able to complete
the import process. The following files were imported:
<File Listing... one per line>
```

Incorrect, Unknown and Unrecognizable HTML Code Drumbeat detects bad HTML code when importing a site. Bad code is code with missing end tags, extra end tags, wrong end tags, dangling end tags, misplaced end tags, and so on. All of these are logged into the ImportStatus.log file.

Sometimes, the bad HTML causes import errors like wrapping text or incorrect fonts. If you discover anomalies in your imported page(s), please check the ImportStatus.log for errors and look for entries like this:

```
4/2/98 12:06:24 PM Unknown HTML tag detected ! /B
4/2/98 12:06:24 PM Unknown HTML tag detecte /TD
4/2/98 12:06:24 PM Unknown HTML tag detected !
4/2/98 12:06:24 PM Unknown HTML tag detected ! /TD
```

Imported assets and SmartElements

The following table shows how the Drumbeat site import function handles various assets.

Text	All simple text is imported into Drumbeat. Large blocks of text are grouped into a single text element. If text is separated by another element, Drumbeat separates the text into two text elements. If the text exceeds the maximum allotted size (approximately 28K), the text is broken up into two or more text elements at the closest line break before maximum size is reached.
Images	All images on an imported page are imported into a local media folder on your system. The folder is located in C:/Program Files/ Drumbeat/My Sites/Import/SiteName (assuming C is the drive where Drumbeat is installed). All images are displayed as they would appear in a browser.
Form Elements	Standard HTML 3.0 form elements are imported. You can import multiple forms within Drumbeat.
Links	All <A HREF> tags are imported. If the link is a hypertext link, it is displayed as a link in the text element. If the link contains an image or other SmartElement, the link is displayed in the Attributes tab. Both HREF and Target attributes are imported.
Anchors	All anchor tags (<A>) are preserved in import.
Framesets	A page is added to the Drumbeat site structure for each frame tag in the imported site. However, the frameset itself is not imported. A new frameset must be created and pages assigned to its frames.
Frames	A new HTML page is added to the site structure for each frame tag within a frameset. The default page for the frame is listed under the frame name in the site structure. Any other pages that are linked into the frame are listed as a part of the regular site structure under their default HTML names.
Applets	Applet tags and the class files they refer to are imported into Drumbeat. As applets are encountered, they will be registered into the Drumbeat SmartElement Library under the Applet tab. Only the attributes of applets are preserved in the registration process.
ActiveX	ActiveX controls wrapped in object tags are imported and registered into the SmartElement Library under the ActiveX tab. The attributes are preserved in the registration process.
Scriptlets	Scriptlets are imported and registered into the SmartElement Library as they are encountered in the HTML files. When an object tag is encountered with a type of "text/x-scriptlet," the following attributes are used to configure the Scriptlet layout: Object ID, Width, Height, and Data (this will point to the HTML file containing the scriptlet definition, which is referenced as the base source for the Scriptlet in the SmartElement Library).
Object Tags	Object tags are imported for any component type wrapped in them. ActiveX, Applets, Scriptlets and Plug-ins can all be wrapped in object tags. The component type is determined by the type attribute.

Imported JavaScript

All JavaScript files encountered during import are placed in \Program Files\Drumbeat\Import\SiteName as JS files. If multiple JavaScript files are encountered during import, the file names will be appended with a consecutively increasing number.

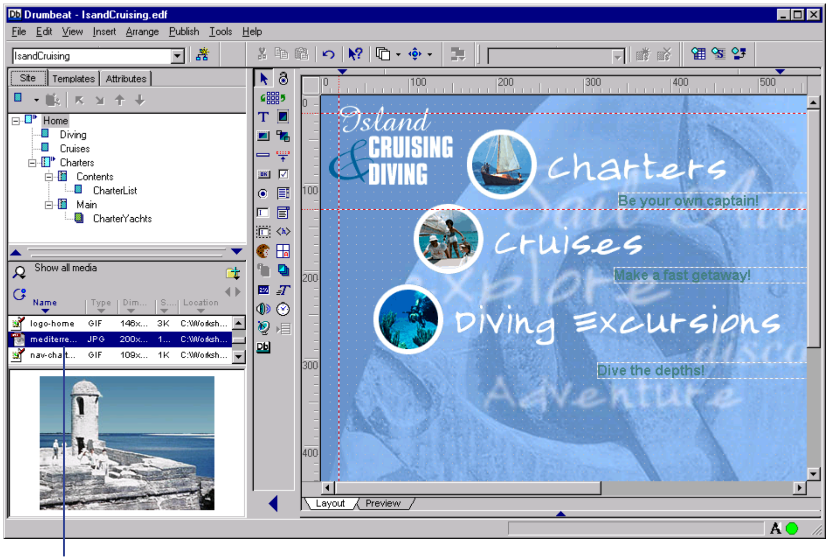
You can manually add JavaScript code or cut and paste the imported code into the Contract Properties dialog box or the Drumbeat JavaScript editor to create point-and-click interactions. The notation at the beginning of each file indicates from which page it originated.

CHAPTER 3
Asset Management

All of the assets you use in your site can be managed in Asset Center. From Asset Center you can add asset files directly to the workspace by pointing to locations on local or network drives. This comes in handy, especially in the realm of collaborative development, by keeping assets in one centralized location.

Using Asset Center

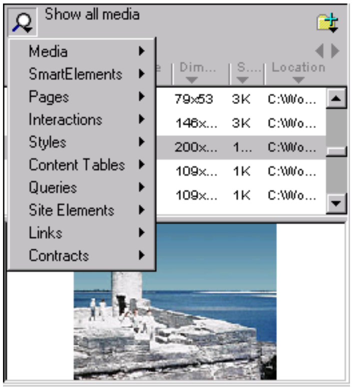
Each asset type, except pages, is added through Asset Center. However, the interface and procedure for adding each asset is very different. This chapter explains how to add and manage each type of asset.


Drumbeat - IsandCruising.edf
File Edit View Insert Arrange Publish Tools Help
IsandCruising
Site Templates Attributes
Home
Diving
Cruises
Charters
Contents
CharterList
Main
CharterYachts
Show all media
Name Type Dim... S... Location
logo-home GIF 146x... 3K C:\Worksh...
mediterre... JPG 200x... 1... C:\Worksh...
nav-chart... GIF 109x... 1K C:\Worksh...
Island CRUISING & DIVING
charters
Be your own captain!
cruises
Make a fast getaway!
Diving Excursions
Dive the depths!
Layout Preview

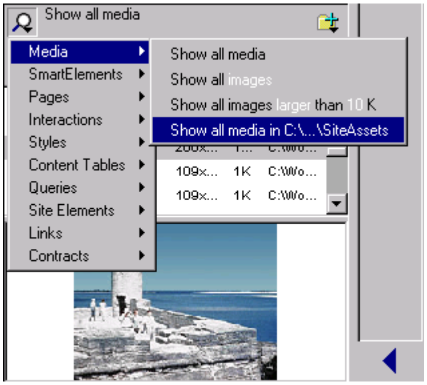
Asset Center

Asset Center is located in the lower-left corner of the Drumbeat window.

From Asset Center, you manage all the content items, including pages, that make up the site. You find assets by choosing asset queries. Asset selections are displayed on a pop-up menu opened by clicking the Locate Assets button.



Each asset has a group of queries listed on cascading menus. You select a query to refine the asset search.

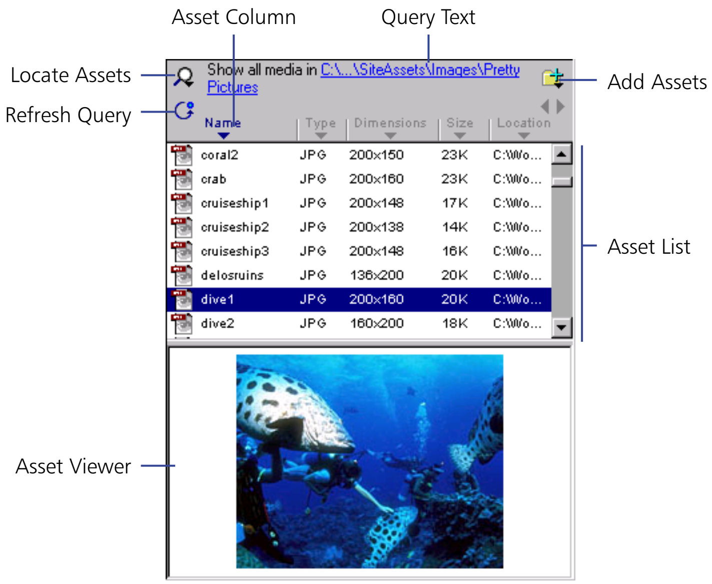


Once you have chosen an asset query, the query results are displayed in the Asset List window. Asset List is a filterable, scrollable, sortable area.

Every asset type has different columns of information. This chapter explains all the columns for each asset. All asset columns are sortable in ascending or descending order. Also, as a space-saving convenience, this same information will appear in a tooltip when you pause the pointer over an asset file.

Looking at the Asset Center

Asset Center is where you add items to pages in the layout. The layout is on the right side of the screen where you design and assemble the site. You can drag most assets, like media and SmartElements, to the layout. One exception is interactions. With interactions, you select SmartElements in the layout, then in Asset Center you assign the interaction by double-clicking on the one you want. You can assign interactions to selected SmartElements from the Interactions Center as well. See “Interactions Center” on page 397.



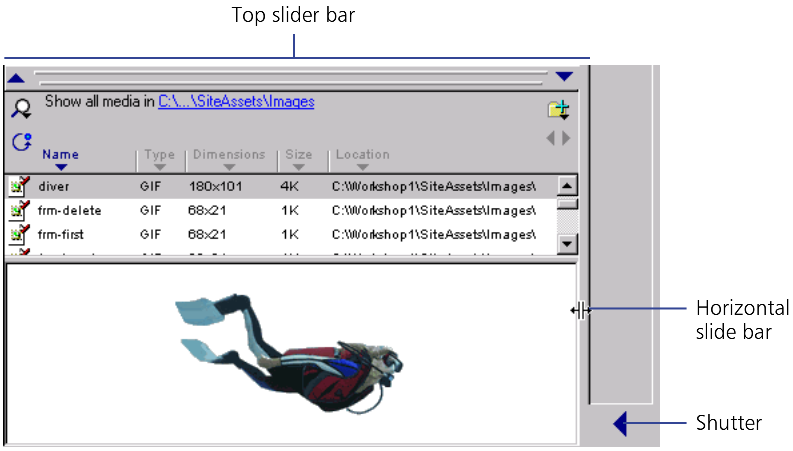
Show all media in C:\...\SiteAssets\Images\Pretty Pictures

Name	Type	Dimensions	Size	Location
coral2	JPG	200x150	23K	C:\Wo...
crab	JPG	200x160	23K	C:\Wo...
cruiseship1	JPG	200x148	17K	C:\Wo...
cruiseship2	JPG	200x138	14K	C:\Wo...
cruiseship3	JPG	200x148	16K	C:\Wo...
delosruins	JPG	136x200	20K	C:\Wo...
dive1	JPG	200x160	20K	C:\Wo...
dive2	JPG	160x200	18K	C:\Wo...

- The Locate Assets button opens a pop-up menu that displays asset queries.
- The Refresh Query button repeats the selected asset query. If you add more assets to an asset location—for example, add more images to a folder—you must refresh the query to add the new assets to the Asset Viewer.
- The Add Assets button opens a pop-up menu that displays all of the assets that you can add using Asset Center.
- Asset List is a filterable, scrollable, and sortable area that displays assets. You can drag most assets to the layout.
- Asset Viewer is a resizable area that displays thumbnails of selected graphic media files and samples of selected styles.
- Information Columns sort the assets in ascending or descending order.
- The Query Text area displays the query in plain English. Queries can be further defined using changeable values. Changeable values are underlined in blue.

Slider bars

You can enlarge the height and width of Asset Center. The horizontal slider bar separates the SmartElement toolbar and Asset Center. You can drag the bar to the left and to the right. The slider bar used to increase the height runs across the top of Asset Center. You can drag this bar up or down.



Show all media in C:\...\SiteAssets\Images

Name	Type	Dimensions	Size	Location
diver	GIF	180x101	4K	C:\Workshop1\SiteAssets\Images\
frm-delete	GIF	68x21	1K	C:\Workshop1\SiteAssets\Images\
frm-first	GIF	68x21	1K	C:\Workshop1\SiteAssets\Images\

Enlarge Asset Center with slider bars.

The arrow icons, on the top slider bar, snap the bar to the next available location. When the bar is in an intermediate vertical location both down and up arrows are present. Clicking the down arrow will snap the bar to the very bottom of the display area. This essentially hides the Asset Center area. Conversely, clicking the up arrow snaps the bar to the very top of the display area. This vertically maximizes the Asset Center area and hides the Site Management Center area.

When the bar is at the very bottom, the down arrow turns into an up arrow. Clicking either up arrows snaps the bar to its former intermediate position.

When the bar is at the very top, the up arrow turns into a down arrow. Clicking either down arrows snaps the bar to its former intermediate position.

Shutter button

The Shutter button is the button with the arrow located at the bottom of SmartElement toolbar. Clicking the Shutter button snaps the horizontal slider bar to the very leftside position, hiding Asset Center. Clicking the Shutter button a second time snaps the bar back to its former position, exposing Asset Center.

Note: You can use the keystrokes Ctrl+L or Ctrl+Space to snap the horizontal slider bar to the very leftside position, hiding Asset Center and minimizing the layout.

Managing media assets

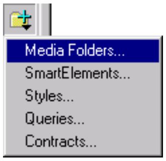
Media files that have been added to Asset Center are available for use in your workspace. From Asset Center, you can query for all the media files in the asset pool, or a subset of them, depending on what you want to list in the Asset List window. Asset List gives you information about each asset file. By clicking on the information columns, you can sort the assets in ascending or descending order. When you select an image file, you see a thumbnail of that image in Asset Viewer.

Adding media files

Media files are added to your workspace through Asset Center, with the exception of content tables which have been created from a database. When you point to an asset source folder you can choose whether to include its subfolders. When a folder includes subfolders of assets, choosing this option saves you time by automatically adding each asset within each folder in the directory structure.

To add media files to your workspace:

1 In Asset Center, click the Add Assets button and choose Media Folders.



2 In the Media Folders dialog box, click the Add button.

3 In the Add Media Folders dialog box, click the Browse button and navigate to the folder that contains the media files you want to bring into your workspace.

4 To include all the assets within the sub-directories of the media folder path, select Include Sub-Directories.

5 Choose whether to show media only for the active site or for all sites by selecting the corresponding show-media options.

6 Click OK in the Select Folder dialog box, and click OK again to close the Media Folders dialog box.

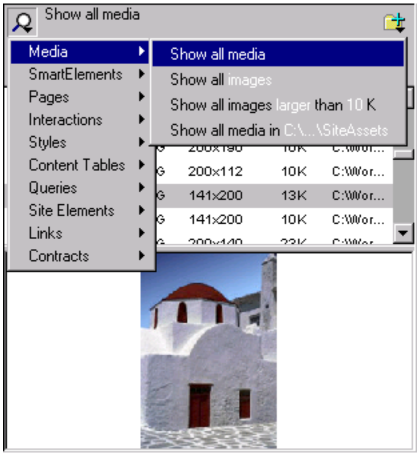
Note: If you have chosen a media query and then added more media assets, you must click the Refresh Query button to show the added assets in Asset Viewer.

Querying for media files

You can query for media assets by media type, file size, and folder location.

To query for media assets:

1. Click on the Locate Assets button, select Media, and from the pop-up menu choose a media query. Query for media assets by type, size or location.

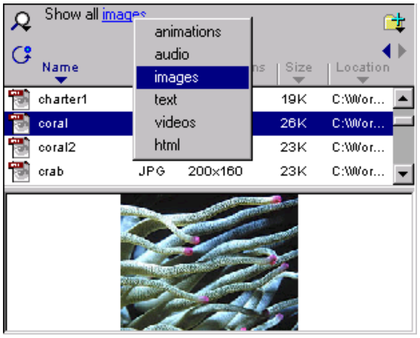


2. To find a particular media file, click on the Asset List columns to sort the assets in ascending or descending order.
3. Click inside the window and type the first letter of the file name and the first file starting with that letter will be highlighted. You can move up or down in the asset list using the arrow keys.

Refining media queries

Asset Center provides media queries that refine your search down to the type, size, and location of the media file. Each media query, except for the Show All Media query, has an option to refine the search even further. By clicking on the blue-underlined hotspot within the plain-English query sentence and choosing a parameter, you can refine the query until you have chosen the most detailed query possible.

Show All Images this query divides the media assets into media type.

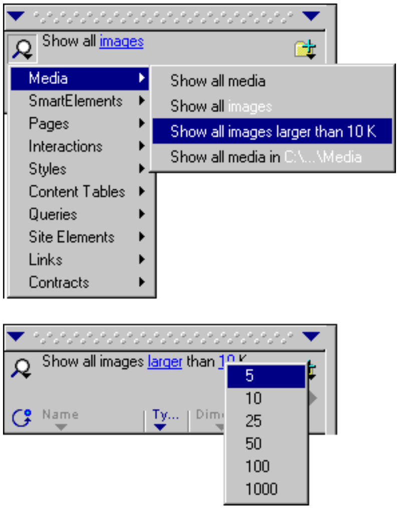


Name		ns	Size	Location
charter1			19K	C:\Wor...
coral			26K	C:\Wor...
coral2			23K	C:\Wor...
crab	JPG	200x160	23K	C:\Wor...

Query by media type.

Animations	Lists all animation files with .flc and .fli extensions. Animated .gif files do not list.
Audio	Lists all audio files with .au, .mid, .midi, .ra, .ram and .wav extensions.
Images	Lists all image files with .bmp, .eps, .gif, .jpg, .pcx, .psd, .rle and .tif extensions.
Text	Lists all text files with .rtf and .txt extensions.
Videos	Lists all video files with .asf, .avi, .dir, .mov, .mpg and .rm extensions.

Show Image Files Filtered by File Size You can filter the list of image files by file size. You can display images that are smaller or larger than a specified size.



Querying for images by size.

Show All Files in a Selected Folder This option lets you list all available media contained in the selected folder. By default, this is set to Drumbeat 2000\Media.

Changeable values for this query are the individual media path folders set in the Media Folders dialog box. This includes sub-directories when the Include Sub-Directories option is selected for a path.

Viewing media files

In the Asset List, you can open the media files in the associated editors or player by right-clicking and choosing Open or by double-clicking. Image and animation files are opened in your default browser. Text files are opened in their default text editor. Audio and video files are opened in the associated players. (This uses the file associations set up in Windows.)

At the bottom of Asset Center is a resizable area, called Asset Viewer, that displays thumbnails of graphic media files when selected in the Asset List.

Adding media to pages

Once you've located a file in Asset Center, drag it to the layout to add it to the page.

To add media to the page:

1. Within the Asset List, locate and click on the media file you want to add.
2. Drag the media file to the layout or to its media-related SmartElement on the layout. Or drag the media file to a content table cell.

   A thumbnail of the referenced media will appear on the layout or within the content table cell. If you place the media file directly on the layout, Drumbeat automatically places the media within the appropriate SmartElement container.

Managing SmartElements in Asset Center

SmartElements are components that have been registered in, or are standard in, Drumbeat. Standard and preregistered SmartElements, including the ones you register, are part of the asset pool and ready to be used in a site.

You can query for all the SmartElements in the asset pool, or a subset of them, depending on what you want to list in the Asset List window.

SmartElements that are not supported under the browser preference set up for the site or page are displayed in the Asset List with a line through them. When dealing with SmartPages and going from one browser support to another, Asset Center automatically updates the Asset List.

The information columns for SmartElements are: name, type, media/category. The media information column normally shows the media file format supported by the SmartElement: animation, audio, image, text or video. This column is blank for many SmartElement types. When filtering by functionality, this attribute is listed as Category: miscellaneous, media, text, button, database, form and table.

Adding SmartElements

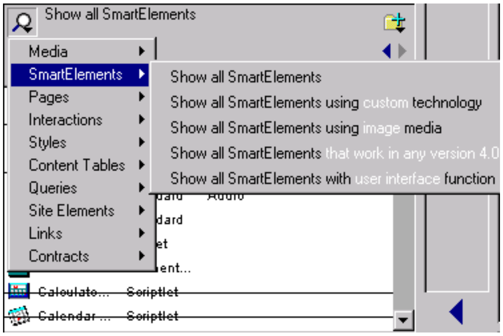
When you choose the SmartElement option from the Add Assets pop-up menu, the Element Library dialog box appears. From this dialog box you can add, delete, edit, import, and export SmartElements. See "Element Library" on page 233.

Querying for SmartElements

You can query for SmartElements by media type, component technology, category, and browser support.

To query for SmartElements:

1 Click the Locate Assets button, and select SmartElement. Choose a SmartElement query. Query by media, component technology, function, and browser support.


Show all SmartElements
Media
SmartElements
Pages
Interactions
Styles
Content Tables
Queries
Site Elements
Links
Contracts
Show all SmartElements
Show all SmartElements using custom technology
Show all SmartElements using image media
Show all SmartElements that work in any version 4.0
Show all SmartElements with user interface function
dard
et
ent...
Calculato... Scriptlet
Calendar... Scriptlet

2 To find a particular SmartElement, click the Asset List columns to sort the assets in ascending or descending order.

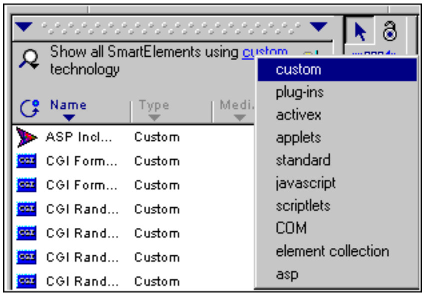
3 Click inside the window and type the first letter of the element name, and the first element starting with that letter will be highlighted.

Refining SmartElement queries

Asset Center provides SmartElement queries that refine your search down to the media type, component technology, function and browser support. By clicking the blue-underlined query phrase within the plain-English query sentence and choosing a parameter, you refine the query until you have chosen the best query for your search.

Show all SmartElements lists all SmartElements available to the site. The resulting list includes all Standard, Applets, JavaScript, Scriptlets, COM objects, ActiveX, Plug-Ins, Custom, and DTC SmartElements.

Show All SmartElements by Type lists all available SmartElements by type. The default shows all SmartElements using custom technology.

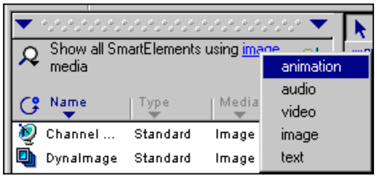


Show all SmartElements using custom technology

Name	Type	Medi.
ASP Incl...	Custom	
CGI Form...	Custom	
CGI Form...	Custom	
CGI Rand...	Custom	
CGI Rand...	Custom	
CGI Rand...	Custom	
CGI Rand...	Custom	

Querying for SmartElements using custom technology.

Show All SmartElements by Media Support list all available SmartElements by the type of media they support. The default shows all image SmartElements.

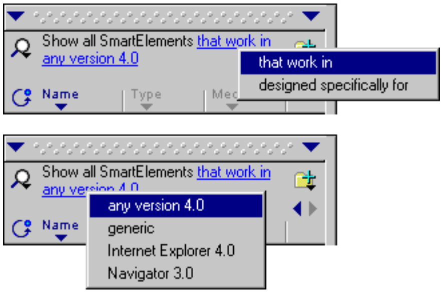


Show all SmartElements using image media

Name	Type	Media
Channel ...	Standard	Image
DynaImage	Standard	Image

Querying for image media assets.

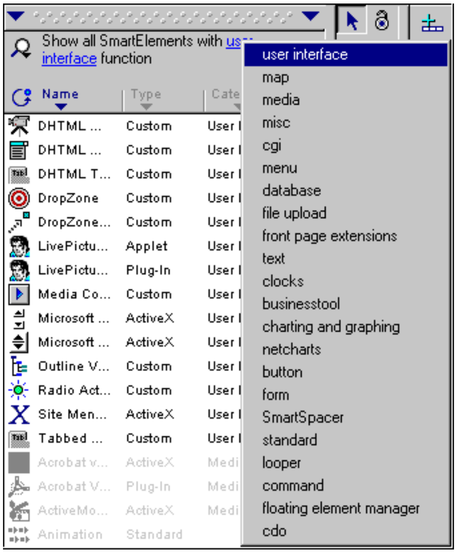
Show All SmartElements by Browser Support lists all available SmartElements by the browser options they support. The default shows all SmartElements that work in any version 4.0.



Show all SmartElements that work in any version 4.0

Querying for SmartElements by browser support.

Show All SmartElements by Functionality lists all available SmartElements by functionality. The default shows all SmartElements with navigation function.



Show all SmartElements with us interface function

Name	Type	Cate
DHTML ...	Custom	User I
DHTML ...	Custom	User I
DHTML T...	Custom	User I
DropZone	Custom	User I
DropZone...	Custom	User I
LivePictu...	Applet	User I
LivePictu...	Plug-In	User I
Media Co...	Custom	User I
Microsoft ...	ActiveX	User I
Microsoft ...	ActiveX	User I
Outline V...	Custom	User I
Radio Act...	Custom	User I
Site Men...	ActiveX	User I
Tabbed ...	Custom	User I
Acrobat v...	ActiveX	Medi
Acrobat V...	Plug-In	Medi
ActiveMo...	ActiveX	Medi
Animation	Standard	

Querying for SmartElements by functionality.

Adding SmartElements to pages

Once you've located a SmartElement, drag it to the layout to add it to the page.

To add elements to the page:

1 Within the Asset List, locate and click the SmartElement you want to add.

2 Drag the SmartElement to the layout.

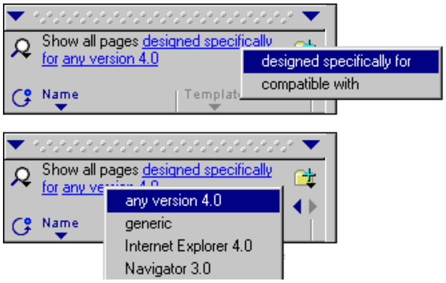
Managing pages in Asset Center

Even though pages are usually dealt with in Site Manager, you can also query for them in Asset Center. You can query for all the pages in the site or only those supported under a certain browser. The Asset List displays which template the page is under. From the right-click (context) menu, you can view the general, publish, and channel properties for each page. (The page named Home by default does not support channels and therefore does not have the Channels properties tab.) See “Developing channels for sites that support Microsoft Internet Explorer 4.0” on page 193.

Choose from the following to refine a page query:

Show All Pages lists all pages within the site. The resulting list identifies native pages, external pages, PageSets, imported pages, SmartPages, and SmartPage redirection pages.

Show All Pages by Browser Support lists all pages within the site by browser support/compatibility. The resulting list identifies native pages, external pages, pagesets, imported pages, SmartPages, and SmartPage redirection pages. The default shows all pages designed specifically for any version 4.0.


Show all pages designed specifically for any version 4.0
designed specifically for
compatible with
Name
Templat
Show all pages designed specifically for any ve
any version 4.0
generic
Internet Explorer 4.0
Navigator 3.0
Name

Querying for pages by browser support.

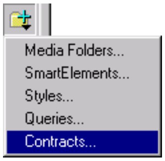
Managing contracts and interactions with Asset Center

By creating a contract, you can cause elements on your page to interact. A contract is a predefined action written in JavaScript. The contract is set up between elements by applying interactions, which can be applied either through the Asset Center or through the Interactions Center. However, we recommend using the Interactions Center, as it is easier to read the full description in the wider Attic space.

Each Interaction has its own set of parameters. When an interaction is applied to an element or elements, its parameters are available by right-clicking the interaction in the Asset Viewer and choosing Edit.

Adding contracts

Choose the Contracts option from the Add Assets pop-up menu.



Use the Contract dialog box to add, edit, delete, copy, arrange, import, and export contracts.

For more information on dealing with contracts, see “Contracts” on page 417.

Querying for interactions

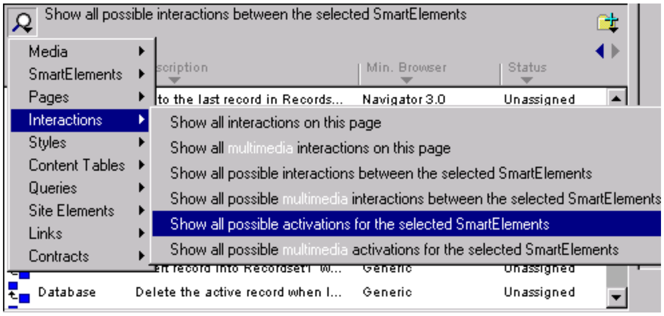
Contracts define interactions. When you select an element or elements in the layout and apply an interaction, Drumbeat sets up a contract with the element. When dealing with more than one element, Drumbeat sets up a contract between the elements, and defines the action as well as the time when it is performed.

You can query for existing interactions that have been set up on a page, or you can show all the interactions that are possible for that element using Asset Center. Interactions can also be sorted by type and browser support.

Tooltips with short descriptions of each interaction appear when you pause the pointer over an interaction in the Asset Center.

Interactions with a red line through them are disabled. If an interaction is disabled, the tooltip will give a short description of the problem and some idea of how to solve it. For more about disabled interactions, see "Disabled interactions" on page 413.

Note: Everything that you see in the Asset Center can be seen in the Interaction Center also, except for tool tip descriptions. Use the Interaction Center when you want to more easily read the text of the interaction. Use the Asset Center when you want a quick overview of the interactions you're using, or to help you diagnose disabled interactions.


Show all possible interactions between the selected SmartElements
Media
SmartElements
Pages
Interactions
Styles
Content Tables
Queries
Site Elements
Links
Contracts
scription
Min. Browser
Status
to the last record in Records...
Navigator 3.0
Unassigned
Show all interactions on this page
Show all multimedia interactions on this page
Show all possible interactions between the selected SmartElements
Show all possible multimedia interactions between the selected SmartElements
Show all possible activations for the selected SmartElements
Show all possible multimedia activations for the selected SmartElements
ert record into Recordset1 w...
Generic
Unassigned
Database
Delete the active record when l...
Generic
Unassigned

Query for interactions at the page or element level.

Show All Interactions for a Page by Type lists all assigned interactions by type for the page or template in focus. The default shows all multimedia interactions on the page.

Show All Available Interactions for a Page lists all the available interactions for a page or template. The default shows all possible multimedia ways to activate the page.

Adding interactions to SmartElements

You can add interactions to SmartElements or to pages by querying for any possible interactions in Asset Center.

To query for possible interactions for SmartElements:

1 Select the SmartElement(s) on the layout.

2 In Asset Center, select the Show All Possible Ways to Activate the Selected SmartElements query.

3 Once the query is performed, right-click the desired activation or interaction and choose Apply.

4 Each activation and interaction has its own Parameters dialog box that pops up after you choose Apply.

Managing styles

You can create new styles from Asset Center. Then, add them to SmartElements in the layout. When you click on a style in the list, a sample of text in that style appears in the Asset Viewer.

Creating styles

When you choose the Styles option from the Add Assets pop-up menu, the Styles dialog box appears. Use this dialog box to add, edit, delete, copy, arrange and import and export styles.

Querying for styles

The Show All Styles query displays all styles in the Asset List. Each style, when selected, can be previewed in the Asset Viewer window.

Applying styles

To apply styles from Asset Center:

1 Select the text or element in the layout.

2 In Asset Center, choose Styles > Show all styles.

3 From Asset Viewer, select the desired style.

4 Drag the style to a text element in the layout.

Managing content

Site content, including databases queries, content tables and site elements can also be managed and listed in Asset Center.

Managing database queries

In Asset Center, click the Add Assets button, and choose Queries from the pop-up menu.

The Query Manager allows you to create, edit, delete, export, and import queries; it manages all the queries in Drumbeat.

Show All Queries lists all the SQL queries available to the site.

Show All Queries by Type lists all available SQL queries by type.

Content table queries

Show All Content Tables lists all content tables available in the site.

Show All Content Tables by Type lists all content tables within the site by type.

Managing site elements

Site elements are SmartElements that can be referenced (set or retrieved) at the site level. Site-level elements are located on the server, with the one exception being the Cookie element. Cookies are on the client, but work with the server to pass information from page to page. Other site-level elements are part of the application server (ASP) and persist from page to page. This means that in Drumbeat a single instance of the SmartElement can be used on multiple pages. Examples of these objects include COM objects, recordsets, and the Request and Session objects.

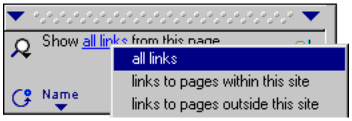
Show All Site Level Elements lists all the site elements within the site.

Show All Site Elements for a Selected Page lists all site elements within a page or template by type.

Managing links

You can check the links on any page or check links throughout the site in the Asset Center. You can choose to filter for links within or outside the site. Additionally, you can check for broken links on a page or throughout the whole site. (A check for broken links through the whole site will take some time if the site is very large.)

Show All Links From This Page lists all links from the active page in the layout.


Show all links from this page
all links
links to pages within this site
links to pages outside this site
Name

Querying for links.

Show All Broken Links to This Page lists all the broken links to the active page in the layout.

Show Broken Links From This Page lists all the broken links to the active page in the layout.

Show All Broken Links from This Site lists all the broken links from the active site.

Use the list of links in Asset Center to directly access and edit any links.

To edit a link from the Asset List:

1 Double-click a link to bring up the Link dialog box and edit the link.

Drumbeat checks all links within the page or the site, according to your selection, and displays them in the Asset Center. Double-click any entry in the list to immediately bring up the Link dialog box for the selected link and fix it.

2 Click the Refresh button in Asset Center to refresh the list when done.

Note: To avoid long delays in refreshing the list of broken links, the broken link list is not automatically updated when you fix a link. However, you can continue to fix the rest of your broken links by double-clicking the next link. Click the Refresh button in Asset Center if you want to refresh the list of broken links after you have made your edits.

CHAPTER 4
Navigation

This chapter discusses constructing and maintaining web site navigation in Drumbeat. Good site navigation through a well-thought-out linking system is one of the most important things you can provide on your web site to make it easy for users to find and access the resources they need. Keeping track of and updating a complex set of links, however, can be a daunting task.

Drumbeat provides various tools that make it easy for you to manage links. Templates can be used to hold navigation buttons, making changes and updates easy on a whole series of pages. Styles can be applied to links, just as they can to ordinary text, to centralize design changes. The linking dialog allows you to create a wide variety of link types using different protocol and custom parameters. Drumbeat also provides several SmartElements for navigational aids, including an image map construction tool and an expanding and collapsing outline structure.

Creating links

Drumbeat supports a full range of linking options. Links can be set on most elements on the layout, or from a text string within a Text SmartElement. Links may be relative or absolute; they may be to pages within the site file or to external pages or resources by URL. Support is included for other protocols, such as FTP, telnet and mailto. You can also create links to files referenced within a database. The Link dialog box handles all these types of links.

The Link dialog box

Links can be created in three different ways, depending upon the anchor element:

- Text links within a text element are created using the Assign Link button on the Text Formatting toolbar.
- Links on a SmartElement (that is, an image or any other SmartElement on the layout) are created with the Assign Link button in the Attributes tab.
- Links from an image map are created with the Image Map HotSpot toolbar.

To create or change the destination if a text link within a Text SmartElement:

1. Within the text box, select the entire linking text by placing your cursor anywhere within the link text.
2. Click the Assign Link button on the Text Formatting toolbar.
3. Assign the link in the Link dialog box or edit the link destination as desired.

To edit a text link or any SmartElement link:

1. Select the entire link text by placing your cursor anywhere within the link text.
2. Click the Link Text button on the Text Formatting toolbar.
3. In the Edit Link Text dialog box, edit the text as desired and click OK when done.

To remove a text link:

1. Select the entire link text by placing your cursor anywhere within the link text.
2. Click the Undo Link button on the Text Formatting toolbar.
3. Delete the text if desired.

To create a link from any SmartElement with a link attribute (for example, Image SmartElement):

1. Select the SmartElement and click the Attributes tab or right-click and choose Attributes.
2. In the Attributes tab, click Link Assign, or right-click on the element and choose Assign Link.
3. Assign the link in the Link dialog box.

To create a link from an image map:

1 Click the image map to activate the HotSpot toolbar. (Double-click the Image Map container if you have assigned an image to the image map but have not assigned the hot spots yet.)

2 Select a shape on the HotSpot toolbar and create a hot spot on the image map.

3 Double-click the image map hot spot (the cursor will change to a hand) to bring up the Link dialog box.

4 Assign the link in the Link dialog box.

The Link dialog box offers a number of linking options. Selecting a link option in the Link pane brings up further appropriate options in the pane on the right. The different types of links that can be created are discussed in the following sections.

Linking to pages within the site

Creating a link to a page within the site is as easy as selecting the page in the Link dialog box.

To link to a page, frameset or frame:

In the Link dialog box, select a page, frameset, or frame and click OK.

Specify a target window or anchor (bookmark) within a page for the link if desired from the Target and Anchor pop-up menus. See “Using anchors (bookmark links)” on page 76 and “Targeting links” on page 77.

Relative links within the Site Tree

Relative links to Next Page or Previous Page can be created based on the order and hierarchical relationships reflected in the Site Manager site tree.

To create a relative link:

In the Link dialog box, select Next Page or Previous Page from the Link Targets list and click OK.

Relative links are self-adjusting, meaning that they maintain their relational integrity even when new pages are added, deleted, renamed, and rearranged within the site structure. As changes are made, links will be remade to reflect the new Next Page or Previous Page, as shown in the Site Manager site tree.

Next Page and Previous Page links are active only between pages that share a common ancestor and are within the same level in the site tree.

Using anchors (bookmark links)

Anchors are tagged locations within a page to which you can create a link. When the user selects a link to an anchor, the page will automatically scroll to that point when it loads in the browser. This is useful both for long documents and for creating a hyperlinked table of contents.

Any element on a page can serve as an anchor for direct linking purposes.

To define an anchor element:

1. Select the element and click the Attributes tab or right-click and select Attributes.
2. In the Attributes tab, click the Anchor checkbox.
3. Enter a name for the element in the Name field. (This is optional, but since the name will be reflected in the Anchor pop-up menu in the Link dialog box, giving it an appropriate name will make identification easier.)

To link to an anchor within a page (create a bookmark link):

1. If the bookmark anchor does not yet exist in the target page, create it.
2. Select the text element to be linked, and in the Attributes tab, click Link Assign.
3. In the Link dialog box, select the page to which the link will be created. (If the anchor is within the same page, select the page that is currently in layout.)
4. Select an option from the Anchor pop-up menu. (The menu will include all anchors created on the page, as well as the default options, End of Page and Start of Page.)

Targeting links

By default, a linked page opens in the same window as the current page, replacing it in the browser window. If you want the page to load in a new window or into a different frame in a frameset, you can specify a target for the link. You can target Full Window to open a link from a frame into the full browser window, replacing any current frameset.

To specify a target for a link:

In the Link dialog box, click the Target pop-up menu and choose a target from the list. Possible targets include the following:

- None (Default): The new page is displayed in the current browser window.
- New Window: A new browser window is opened and the page is displayed in the new window, leaving the original window intact.
- Full Window: The new page will open in the full window, replacing any currently active frameset.
- Frame (name): The new page will load into the specified frame. When linking from a frame in a frameset, the Target pop-up menu will show the available frames.

For a working example of targeting a link to open in a new, custom window, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

Linking to a URL or other protocol

Links to URLs include web links (http) and links to other Internet protocol's (such as FTP, telnet, and mailto).

To create a link to a URL:

1. Select the element to be linked, and click Link Assign in the Attributes tab, or the Assign Link button in the Text Formatting toolbar for a text string.
2. In the Link dialog box, select URL in the Link window. A list of options will be displayed on the right.
3. Select the protocol from the options displayed (the default is http://).
4. In the URL box, enter the address. (Do not add the protocol; the appropriate protocol (prefix such as http://) will be appended to the address.)

The protocol choices are shown in the following table.

Protocol	Description
<none>	Creates an internal anchor within a text box to reference by other links
file://	Link to a local system or network file or application
ftp://	Link to an FTP server (File Transfer Protocol)
gopher://	Link to Gopher resources on the Internet
http://	HyperText Transfer Protocol (World Wide Web protocol)
https://	Proprietary security link
javascript:	Direct link to a JavaScript function
mailto:	E-mail link (brings up a pre-addressed e-mail window)
rlogin:	Remote login link
shttp:	Link to a secure server
snews://	Link to a secure news server
telnet://	Link to a Telnet connection (to a remote computer)
tn3270://	Remote access type link
wais://	Wide Area Information Server link
[custom]	Specify your own custom prefix

Relative links within the published site

To create a relative link to a file within the published directory:

1 Within the Link dialog box, select URL. For protocol, select <none>.

2 In the URL edit box, enter the relative path, for example:

../path/filename.

Mailto

A mailto link allows users to send e-mail easily by clicking on the link, which brings up a pre-addressed e-mail window.

To create a mailto link:

1 Select the text or element to be linked and open the Link dialog box.

2 Select URL in the link window, then select the mailto: option.

3 In the URL path box, enter the e-mail address.

Linking to files

You can create a link to a file or an application residing on the local machine or network with the file protocol. If the Windows file associations are correct on the user's machine, the appropriate application will be launched to view the file (such as Word to handle a DOC file or Excel to handle an XLS file). Shared applications can also be launched when using a file link.

To create a link to a file:

1 Select the text or element to be linked and open the Link dialog box.

2 Select URL in the link window, then select the file:// option.

3 In the URL path box, enter the path to the file (C:\path\filename.doc)

FTP

FTP URLs are used to point to files located on FTP servers. Many web browsers can function as FTP clients, which means casual users don't need a separate FTP client to access resources on an FTP server.

To create a link to an FTP server:

1. Select the text or element to be linked and open the Link dialog box.
2. Select URL in the link window, then select the ftp:// option.
3. In the URL path box, enter the FTP server host name.

Note: FTP URLS are best used to link directly to a file, rather than to a directory on a FTP server, since navigating FTP servers using a web browser is often much slower than using an FTP client.

An FTP link may also be established for named accounts with username/password access on an FTP server. This allows your site visitor to both upload and download files to the FTP server. The URL will look something like this:

ftp://username:password@ftp.YourCompany.com

Note: The user name and password in a URL should be included when security is not a problem.

Linking to a CGI script from a form

You can create a link to a CGI script that will be activated when a form is submitted.

To create a link to a CGI script:

1. Create a form by adding the appropriate form elements to the layout.
2. Click the Form element in the Basement to bring up the Attributes tab for the form.
3. In the Attributes tab, click the Action Assign button.
4. In the Link dialog box, select URL in the link window.
5. Using the default option of http://, enter the complete path to the CGI script in the URL box.

   Note: Make sure that the names of the form elements on the Drumbeat layout match the names used in the CGI script. You can change the name of the form elements in the Attributes tab.

Using URL parameters

URL parameters can be used to append a query string to a URL link. A query string represents name-value pairs of variables that will be appended to the Request Object collection. An example of a hyperlink with URL parameters is:

http://www.drumbeat.com/test.asp?a='test1'&b='test2'.

The part of the URL after the question mark is the query string. In this example the query string will append two variables to the Request Object collection.

To append a query string to a URL:

1 In the Link dialog box, choose the URL option.

2 In the URL edit box, enter the URL, followed by a question mark.

3 In the URL Parameters box, enter the query string you want to append to the URL.

Developers often use this technique when they wish to pass information from page to page without having to use the form submit method. One reason for using this option in Drumbeat is to position the database cursor on a specific record in the record set when a data loop is used. For an example of this usage, see the Search Engine Starting Point. The Insert and Update buttons on the Detail page of this Starting Point have URL parameters applied to position the cursor correctly for the current record. See also "Targeting links" on page 77.

Maintaining links in a content table

URLs that you want to use for links can be stored in a content table for easy editing or reuse. The content table may be either a manual content table that you create or a content table generated from a database—either a CSV file or an ODBC database (if you are using ASP server support). If you are using database content and the records have URLs associated with them, this allows you to create automatic, database-driven links for each record.

The content table cell that contains the links can contain either a full or a partial URL. The column type must be text and the text type must be unformatted.

Examples of link syntax that will work:

./products/product234.html
media/tada.midi
http://www.drumbeat.com/
ftp://209.67.164.102/XChange/DrumbeatUserGuide.exe

To create a link to a URL referenced in a content table:

1 Select the text or element to be linked and open the Link dialog box.

2 Select the content table that contains the URL in the Link Targets window.

3 Select the row and column that contains the URL.

Checking and updating links

You can check the links on any page or check links throughout the site in the Asset Center.

To query for all links:

Click the Locate Assets button in the Asset Center and from the pop-up menu choose Links > Show all links from this page or Show all links to this page.

When displaying links from the page, you can choose to filter further for links to pages within this site or links to page outside this site. (Click on the underlined text to change the query.)

To edit a link from the Asset Center list:

Double-click the link you want to edit to bring up the Link dialog box for that link.

Additionally, you can check for broken links on a page or throughout the whole site. (Checking through the whole site will take some time if the site is very large.)

To check for broken links:

1. Click the Locate Assets button in the Asset Center and from the pop-up menu choose Links > Show All Broken Links from This Page or Show All Broken Links from This Site. Drumbeat checks all links within the page or the site, according to your selection, and displays them in the Asset Center.
2. Double-click any entry in the list to immediately bring up the Link dialog box for the selected link and fix it.
3. After fixing the broken links, click the Refresh button in the Asset Center to refresh the list.

   Note: To avoid long delays in refreshing the list of broken links, the broken link list is not automatically updated when you fix a link. However, you can continue to fix the rest of your broken links by double-clicking the next link. Select the Refresh option in the Asset Center if you want to refresh the list of broken links after you have made your edits.

Link styles and effects

Because links are very often text links, they can have styles assigned to them just like ordinary text. Nontext links (image links) can also have styles assigned to them; they will reflect only those properties that are appropriate to the element (such as border color).

Link styles

To select link styles:

1. In the Attributes tab for the page or template, click the Inherit button.
2. In the Inherit Attributes window, deselect the link styles you want to assign, then click OK.
3. In the Attributes tab, select a style from the pop-up menu next to each type of link (Link, Visited Link, Active Link, and Hover Link).

If you have links set on text of various fonts and sizes on your page, you may want to assign only link colors, rather than full font styles. You can do this by setting up a custom style that contains only the color definition.

Four default styles are included for this purpose: LinkColor, VLinkColor, ALinkColor, and HLinkColor. These styles contain only color settings. You can change the color by editing the default style.

To customize link colors:

1. In the Link Style pop-up menu, select the LinkColor style.
2. Click the ellipses button next to the list to bring up the Style Properties dialog box.
3. Click the color bar and choose the color you want to apply.

Do the same for the other link styles. The style colors will be saved as the new global styles.

Note: Drumbeat 2.0 users who have migrated their global.dgd over to the new version will not have these link color styles available. You can define your own link color styles instead.

You can define your own styles for link colors if you want to create custom colors for specific sites.

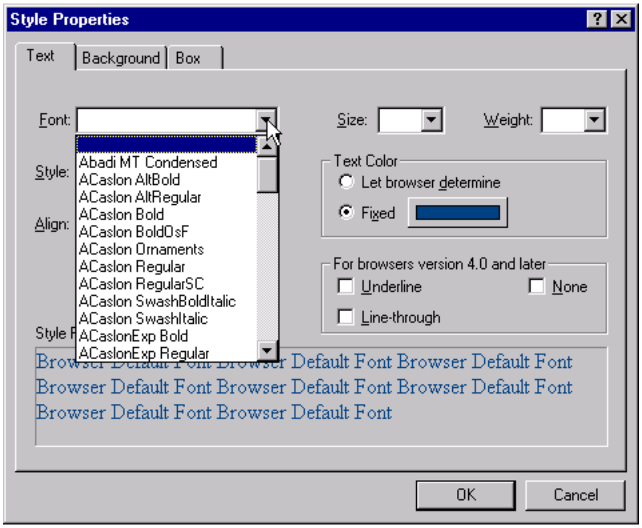
To create a style for link colors only:

1 Select Tools > Styles to create a new style.

2 In the Styles dialog box, click New.

3 In the Style Properties dialog box, select the text color for the link.

4 In the Font, Size, and Weight pop-up menus, select the blank value at the top of the list.



Choose blank values for Font, Size, and Weight to allow the link to inherit these values from other elements.

5 Save the style with an assigned name. If you want to make this a global style, select the option Make Style Available Throughout All Sites.

The new style will appear in the Style pop-up menu, where you can select it and apply it to the page. The other style properties will be inherited from the style applied to the Text SmartElement in which the link is contained.

For an explanation of how to create your own styles, see “Using styles” on page 128.

Link status text

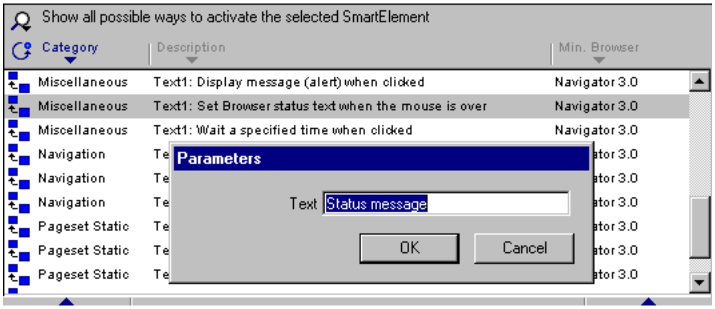
Status text is the text that appears in the status bar at the bottom of the browser screen. When the mouse passes over a link, the link destination usually appears in the status bar. You can customize this text so that, rather than displaying a URL, it displays something more meaningful. To create custom status text for a link, you must apply an interaction from the Interactions Center.

To create custom status text:

1 Right-click the element to which you want to add the status text, and from the pop-up menu, choose Possible Activations.

2 In the Interactions Center, find the activation in the Miscellaneous category that says Set Browser status text when the mouse is over.

3 In the Parameters dialog box, enter the text to display and click OK.



Show all possible ways to activate the selected SmartElement

Category	Description	Min. Browser
Miscellaneous	Text1: Display message (alert) when clicked	Navigator 3.0
Miscellaneous	Text1: Set Browser status text when the mouse is over	Navigator 3.0
Miscellaneous	Text1: Wait a specified time when clicked	Navigator 3.0
Navigation	Te	ator 3.0
Navigation	Te	ator 3.0
Navigation	Te	ator 3.0
Pageset Static	Te	ator 3.0
Pageset Static	Te	ator 3.0
Pageset Static	Te	ator 3.0

Parameters

Text: Status message

Set the status bar message with a simple activation.

To edit the status text:

1 Right-click the element that has the status text activation, and from the pop-up menu choose Assigned Interactions.

2 In the Interactions Center, double-click the status text interaction to bring up the Parameters dialog box and edit the text.

For a working example of setting browser status text, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

Note: Status text can be created only at the element level, so if you want it to apply to a text link, the link must be in its own text container. (You cannot create status text on a text string within a text box; you must apply it to the whole text box.) Status text is not limited to linked elements; it can also be applied to any element on the layout.

Using navigational SmartElements

Drumbeat provides several special-purpose SmartElements to aid in site navigation and links. These include:

- SmartElement Collections: Special collections of navigational buttons or other items that you can create and add to the SmartElement toolbar for handy drag-and-drop use.
- Image Map: A single image that contains multiple link destinations through defined hot spots.
- Outline: A collapsible and expandable outline structure with links.
- DHTML Menu: A pop-up menu implemented in Dynamic HTML that expands as soon as the mouse moves over the element.

SmartElement collections for navigation

A SmartElement collection is a reusable SmartElement created from one or more SmartElements. The collection can be added to the SmartElement toolbar for drag-and-drop use. While SmartElement collections can be made for many purposes, one of the most common and handy uses is to create a set of navigation buttons that will be used repeatedly. Once added to the SmartElement library, a SmartElement collection becomes available for use in all sites, not just the current site.

SmartElement collections retain the content, the relative positioning of the various elements used, assigned links, and interactions (provided all the participants in the interaction are included in the collection).

To create a SmartElement collection for navigation:

1. Arrange the navigation buttons on the layout of any page or template and create the appropriate links for each.
2. Select all the elements (marquee-select by drawing around all the elements or Ctrl-Click each element in turn).
3. Right-click any element and select Add to SmartElement Library.
4. In the Name dialog box, enter a name for the collection and click OK.

The SmartElement collection will appear in the Asset Center when you query for SmartElements. It can be dragged onto the layout from the Asset Center, or you can add the collection to the SmartElement toolbar for frequent use.

To add a SmartElement collection to the SmartElement toolbar:

1 Click the Element Library icon on the SmartElements toolbar.

2 In the Element Collections tab, select the collection you want to add and click Done.

All selected items in the SmartElement library will appear on the SmartElements toolbar, for drag-and-drop use. To remove any SmartElement from the toolbar, deselect it in the SmartElement library or drag the icon on the SmartElement toolbar to the SmartElement Library icon. For details on SmartElement collections, see “SmartElements” on page 213.

Image Map SmartElement

The Image Map SmartElement turns a regular image into one that can be used for navigating to several different pages or URLs by defining multiple “hot spots” on the image. A hot spot is a user-defined area of an image map that can be linked to other destinations such as pages, anchors, or URLs.

To create an Image Map SmartElement:

1 Drag and drop the Image Map SmartElement from the SmartElement toolbar to the layout.

2 Drag the image you want to use for the image map from the Asset Center or from a content table cell onto the Image Map SmartElement.

3 Click the image to bring up the Hotspot toolbar.

4 Select the shape of the hot spot you want from the HotSpot toolbar, then click and draw on the image to define the hot spot.

5 Double-click the hot spot to bring up the Link dialog box. Define the link destination and click OK.

To edit a link on an existing hot spot:

1 Double-click the image to open the image map for editing, then double-click the hot spot area you want to change.

2 In the Link dialog box, edit the link assignment as you wish, then click OK.

To delete a link on a hot spot:

1 Open the image map and double-click the hot spot to be deleted.

2 Within the Link dialog box, select the No Link option to break the assigned link and close the dialog box.

Assigned links are automatically broken when a hot spot is deleted from an image map.

To delete a hot spot:

1. Open the image map and click the hot spot to be deleted.
2. Press the Delete key on the keyboard, or from the main menu choose Edit > Delete.

For more information on using image maps, see “Image maps” on page 168.

DHTML Outline SmartElement

The Outline SmartElement is a custom SmartElement that generates a user-expandable outline in DHTML for IE 4.0 only. Each node can be linked to another page, JavaScript, or an e-mail address. The outline is populated from a Drumbeat data table, which in turn can be generated from a query to a database. This outline is suitable for corporate organizational charts, product listings, and so on.

The data used to populate an outline comes from a Drumbeat content table. The data in the data table must be of the type Unformatted Text. Up to four columns of data can be specified.

To use the Outline SmartElement:

1. In Asset Center, query for SmartElements > Show all SmartElements.
2. Locate the Outline SmartElement and drag and drop it onto the layout.
3. In the Attributes tab for the Outline element, define the content table and the row and column for each node of the outline.

- Outline Text appears in the outline in all cases (required).
- Outline Parent sets up the parent-child relationships to be used in the outline (required).
- Outline Link creates an HREF link for the element (optional).
- Outline Text Alias is used for matching numeric values from a database

Some sample content tables are provided to give you an idea of how the Outline SmartElement works and how to set up your own content table for the outline. You can create a CSV content table using any of the examples and use this for trial content.

For complete documentation on using the Outline SmartElement, see the document OutlineV10_30 in the SmartElements\Navigation directory of your Drumbeat program files, or visit the Drumbeat Element Exchange on the Drumbeat web site at exchange.drumbeat.com.

DHTML Menu SmartElement

The DHTML Menu SmartElement generates a pop-up menu implemented in Dynamic HTML that will run on all 4.0 browsers. The menu pop-up when the mouse moves over the element, providing an economical navigation element that uses a small amount of screen real estate while giving a professional look.

The menu content comes from a content table. The table must have two columns and both columns must be of the type Unformatted Text. Formatting of the text is controlled by a style applied through the Attributes tab and the link colors for the document. The first column contains the text string which appears as menu items when the menu is expanded. The second column is the link (HREF value) of the form http://www.drumbeat.com. You must include the protocol (the http:// part of the URL). (Note that you can also use the form javaScript: as in javaScript:alert('this is a test'); to display an alert box when that menu item is selected.)

To use the DHTML Menu SmartElement:

1. In the Asset Center, query for SmartElements > Show all SmartElements.
2. Locate the DHTML Menu SmartElement and drag and drop it onto the layout.
3. In the Attributes tab for the element, select Static Content from the Content pop-up menu and select content table.

Other attributes include:

- Label is the part of the menu that you see when the menu is not expanded (is collapsed).
- Style applies the selected text style to the menu.
- Height Controls the height of the menu by its content and style rather than the height specified in the Attributes tab (or by sizing the element on the layout). This can be overridden by checking the Height Override attribute. In this case the height of the element will be that specified by the height attribute. Typically you should not override the automatic height calculation of this element.

To use an image in the menu, you can enter an image link in this form:

```
<IMG SRC=myMenuImage.jpg>
```

This "text" will get published as part of the menu content and when the browser encounters this, it will load the image in the menu. You must copy the image to the publish directory, and ultimately move it to your production server, since the Drumbeat content manager is unaware of the existence of this image.

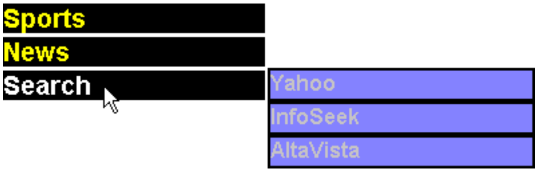
For complete documentation on using the DHTML Menu SmartElement, see the document DHTMLMenuV10_30 in the SmartElements\Menus directory of your Drumbeat program files, or visit the Drumbeat Element Exchange on the Drumbeat web site at exchange.drumbeat.com.

DHTML Cascading Menu SmartElement

The DHTML Cascading Menu SmartElement can be used in sites that target Internet Explorer 4.0. It generates a cascading pop-up menu from a series of text tiles. The menu is generated from a text file and formatted using styles. A sample menu is provided with the element.

To use the DHTML Cascading Menu SmartElement:

1. In the Asset Center, query for SmartElements > Show all SmartElements.
2. Locate the DHTML Cascading Menu SmartElement and drag and drop it onto the layout.
3. In the Attributes tab for the element, click the Outline Assign button to view and change the contents of the menu.
4. Assign styles to the different menu levels. You can assign both normal and mouseover styles. You can create as many levels in the menu structure as are practicable in your design.



Use the DHTML Cascading Menu SmartElement to create a cascading pop-up menu.

For complete documentation on using the DHTML Cascading Menu SmartElement, see the document DHTMLCascadingMenuV10_30 in the Program Files\Drumbeat\SmartElements\Menus folder, or visit the Drumbeat Element Exchange on the Drumbeat web site at exchange.drumbeat.com.

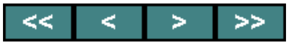
DHTML Recordset Navigation SmartElement

The DHTML Recordset Navigation SmartElement is designed to be used to navigate through recordsets in a ASP site. It replaces the autogenerated Form buttons for First, Previous, Next, and Last. The element can be used with sites that target Navigator 3.0 and later.

To use the DHTML Recordset Navigation SmartElement:

1. In the Asset Center, query for SmartElements > Show all SmartElements.
2. Locate the DHTML Recordset Navigation SmartElement and drag it onto the layout of the DataForm detail page.
3. Select the recordset in the Basement and the Recordset Navigation SmartElement on the layout. Right-click and choose Possible Interactions.
4. In the Interactions Center, locate the interaction that says Navigate Recordset when RecordsetNav is clicked and submit form. Double-click the interaction to apply it to the elements.

You can choose to have the element display words or arrows. Apply a style to the element, and use the background and box properties of the style to make the buttons look more like buttons.



DHTML Recordset Navigation buttons with a style applied that includes background and box properties.

For complete documentation on using the DHTML Recordset Navigation SmartElement, see the document DHTMLRecordsetNavigationV10_30 in the SmartElements\Menus directory of your Drumbeat program files, or visit the Drumbeat Element Exchange on the Drumbeat web site at exchange.drumbeat.com.

CHAPTER 5
Layouts

Topics covered in this chapter include using layout tools and visual aids, techniques to position elements on the layout, and using styles to share formatting attributes across multiple elements on the layout.

Using layout tools and visual aids

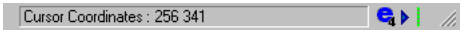
Drumbeat provides several visual aids for positioning elements on your layout. You can refer to precise cursor coordinates, or use the ruler, grid, and guides to position elements. These visual aids are visible during design, but will not appear on your published layout.

Cursor coordinates

The Drumbeat status bar displays the exact coordinates of the cursor when it is in the layout.

To display the Cursor Coordinates:

Choose View > Status Bar.



Cursor Coordinates : 256 341

Using rulers, guides, and the grid

Drumbeat has rulers, guides, and a grid for precise placement of elements on your layout. Elements can snap to accurate positions along guides and grid lines.

Rulers and the grid are displayed by default when Drumbeat is started. Guides can be added, moved, or removed when the rulers are displayed. Rulers, guides, and the grid can be toggled on and off.

Ruler

The horizontal ruler at the top of the layout and the vertical ruler at the left of the layout are measured in pixels.

To show or hide the ruler:

Choose View > Ruler > Show Ruler and Guides to show the ruler. Choose it again to display the ruler.

Guides

You can place guides on your layout from the ruler. You can also create guides based on the position of any element or group of elements on your layout. Use this feature to align multiple elements quickly, easily, and accurately.

To add a Guide to the layout:

On the horizontal or vertical ruler, click the measurement for the new guide.

To add guides around an element or a group of elements:

1 On the layout, select the element.
2 Right-click and choose Make Guides for Current Selection and select Left, Right, Top, or Bottom or All.

To move a Guide:

On the layout, select and drag the guide to the new location.

To remove a Guide:

Select the guide on the layout and press Delete or drag the guide off the layout.

To remove all Guides from the layout:

Choose View > Ruler > Remove All Guides.

To define the guide color:

1 Choose View > Ruler > Settings.
2 On the Ruler tab, click the Guide Color button.
3 In the Color palette, choose the color for all Guides on this layout.

Snapping to guides

Guides can be set to attract, or snap, elements into place along the guide as you drag elements near the guide. The range within which elements are snapped to the Guides is customizable.

To activate and customize snapping to guides:

1 Choose View > Ruler >Settings and click the Ruler tab.

2 Select snap to guides and specify the Snap Range within which an element should snap to a Guide. The snap range is measured in pixels.

Grid

A customizable grid is displayed for easy and accurate placement of elements on your layout. You can specify grid spacing, color, and snapping options.

To set the grid options:

1 Choose View > Grid > Settings.

2 Set grid spacing:

- Type or select the distance in pixels between the grid rows and columns in the Grid Width and Grid Height fields.
- Drag the sample grid to set the width and height. Shift-drag to constrain the grid to a square.

3 Click the Grid Color button and choose a color for the grid.

4 To turn off the grid, deselect Show Grid

5 To make an element snap to the grid as you drag it near a grid point, select Snap to Grid.

Using SmartSpacers

When you are creating data-driven pages, it can be difficult to know the area required on your layout for each piece of content. To avoid overlapping elements, you could assume the largest possible area for your data content and risk ruining an otherwise good layout. Or you can use SmartSpacers. A SmartSpacer is an area of dynamic space on your layout that adjusts the layout spacing based on content.

To place a SmartSpacer on your layout:

Drag the SmartSpacer SmartElement onto your layout.

The SmartSpacer is represented by horizontal dividing lines. The SmartSpacer dividers are visible on the layout, but do not appear on the published page.

The following rules apply to SmartSpacer placement:

- If an element is placed with the top of the element above a SmartSpacer, the SmartSpacer moves down and aligns just below the element.
- If an element is placed below the SmartSpacer, the element aligns just below the SmartSpacer. The SmartSpacer does not move.
- If a SmartSpacer is placed with the top of the SmartSpacer above an element, the element moves just below the SmartSpacer.
- If a SmartSpacer is placed with the top of the SmartSpacer below the top of an element, the element moves just above the SmartSpacer.

Customizing SmartSpacers

SmartSpacer size and position are customizable.

A SmartSpacer has an original position attribute that represents the topmost position on the layout that the SmartSpacer can move to.

To set the height of a SmartSpacer:

Do one of the following:

- Drag the top or bottom control on the SmartSpacer dividing lines. The extent and dimensions are displayed in the Status bar
- Select the SmartSpacer and click the Attributes tab in the Height field, type or select the height in pixels for the SmartSpacer.

To set the position of the SmartSpacer:

Do one of the following:

- Drag the SmartSpacer
- Select the SmartSpacer and click the Attributes tab in the Top field, type or select the position for the top of the SmartSpacer.

To set the SmartSpacer original position:

Select the SmartSpacer and click the Attributes tab in the Origin field, type or select the topmost position for the top of the SmartSpacer.

The SmartSpacer cannot move above the Original Position, even if all content above the SmartSpacer is removed.

Note: SmartSpacers can only be used on sites targeting 4.0 and later browsers. On pages targeting any 4.0 browser, SmartSpacers cannot be used between form elements. Also, SmartSpacers cannot be used within a data loop.

Layout techniques

Your web page is composed of SmartElements arranged in the layout. You can position, arrange, align, resize, layer, set attributes, and add effects to the elements.

Adding an element

In most cases, you will place elements on your layout from the SmartElement toolbar.

To place an element on the layout:

Do one of the following:

- From the SmartElement toolbar, drag a SmartElement to the layout.
- Click a SmartElement, and then click the position on the layout for the top-left corner of the element.
- Click a SmartElement and drag in the layout to draw the space you want the element to occupy.

Note: Text elements are self-adjusting for height. You can draw a desired width for the Text SmartElement, and the height will adjust to fit the content.

Selecting elements

Select elements to move, delete, or edit them.

To select elements on the layout:

Click an element. To select additional elements, shift-click them. You can also drag a selection box around elements to select them. (This method is also called marquee-select.)

Click any unused area of the layout to deselect an element or elements.

Deleting, cutting, copying, and pasting elements

Any element on your layout can be cut or copied to the clipboard, and pasted in this layout or in another site.

To cut, copy, and paste an element:

1 Right-click the element on the layout and choose Cut (Ctrl+X) or Copy (Ctrl+C).

2 On the layout, right-click and choose Paste (Ctrl+V).

To make a copy of an element on the layout by dragging:

Select the element and Ctrl-drag it on the layout.

To delete and element:

Select an item and press delete.

Moving elements

You can move an element by dragging it into position on the layout, by setting the absolute position in the Attributes tab, or by using the arrow keys.

To move an element by dragging:

Drag the element into position.

Refer to the coordinates displayed in the status bar, and the outline of the element on the ruler for precise positioning. To simplify precise drag positioning, turn on Snap to Guides or Snap to Grid.

To move an element using absolute position attributes:

1 Select the element and click the Attributes tab.

2 In the Left field, type or select the horizontal position for the element.

3 In the Top field, type or select the vertical position for the element.

To nudge selected elements one pixel at a time:

Use the arrow keys on the keyboard to nudge the selection one pixel at a time in the desired direction. If the grid is displayed and grid-snapping is selected, press ctrl+arrow to nudge the selection one grid space in the desired direction.

Aligning elements

It is easy to align multiple elements to a reference element or to a Guide.

In a group of selected elements, one element is the reference element. The reference element is the source of an interaction within the group, and the remaining selected elements are targets of the interaction. The reference element is also the fixed element when aligning multiple elements. The reference element remains in its position, and the other selected elements are aligned to the reference element.

By default, the reference element is the last element selected when multiple objects are selected. The reference element is indicated with a solid blue border. The other elements are indicated with a hatched border.

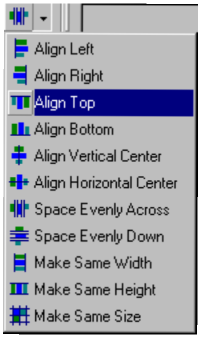
To change the reference element:

Ctrl-click the new reference element.

To align elements horizontally or vertically to a reference element:

1 Select the elements, making the element that is to remain fixed in position the reference element.

2 Click the Arrange button on the toolbar, and choose the desired Align option.



Note: If the specified alignment would result in any element exceeding the bounds of the layout, Drumbeat does not move the elements.

To align elements to a guide:

Click the left or top ruler to create a horizontal or vertical guide, turn on Show Ruler and Guides and Snap to Guides, and drag the elements into position along the Guide.

Evenly spacing elements

You can easily distribute multiple elements evenly. This is a quick, accurate way to create evenly spaced sets of radio buttons, check boxes, buttons, or other elements.

This technique can also be used to precisely align a group of elements so that they abut one another, such as a group of images that are pieces of a larger composite image.

To evenly distribute three or more elements horizontally or vertically:

1 Select the elements.

2 Click the Arrange button on the toolbar and choose one of the following options:

- Space Evenly Across distributes the elements evenly across the width of the area that the elements occupy.
- Space Evenly Down distributes the elements evenly within the height of the area that the elements occupy.

To align elements so that they exactly abut one another:

1 Place the elements on the layout so that they slightly overlap each other, in the order in which they should appear.

2 Click the Arrange button on the toolbar and choose Space Evenly Across or Space Evenly Down.

You may want to first align the elements left, right, top or bottom, depending on their intended orientation.

Scaling elements

Scaling an element enlarges or reduces an element horizontally, vertically, or horizontally and vertically. You can scale an element by dragging in relation to the size of a reference element, or by entering values in the Attributes tab.

To scale an element by dragging:

1 Select the element. If the element is resizable, selection handles are displayed around the element.

2 Do one of the following:

- Drag a center handle to scale the element horizontally or vertically.
- Drag a corner handle to scale the element both vertically and horizontally.

To scale one or more elements to the size of a reference element:

1 Select the elements to be resized and the reference element.

2 Click the Arrange button on the toolbar and choose Make Same Width, Make Same Height, or Make Same Size.

To scale an element by entering values:

1 Select the element.

2 In the Attributes tab, enter values in the Width and Height options.

Note: If the width or height of the element cannot be scaled, the corresponding attribute is not available.

Layering elements

The Left and Top attributes determine the position of an element along the x (horizontal) axis and the y (vertical) axis. the layout also supports a virtual third dimension, so you can align overlapping elements along the z axis. This is called the stacking order, or Z-order.

Elements are stacked from back to front in the order in which they are placed the layout.

To move an element in the stacking order:

Right-click the element in the layout and choose Z-Order. Select the desired option to change the stacking order of the element:

- Move to Front moves the element to the front (or top) of the stacking order.
- Move to Back moves an element to the back (or bottom) of the stacking order.
- Move Up moves an element forward one position in the stacking order.
- Move Down moves an element back one position in the stacking order.

Applying movement and transitional effects

Each element on the layout can have a variety of movement and effects associated with it, and an element can interact with other elements on the layout. Common activations for an element include movements and effects such as fly in, fly out, conceal, reveal, flip, and slide. Activations can be triggered by events like mouse click, mouse over, or page load.

For details about using Activations and Interactions, see “Interactions Center” on page 397.

The Drumbeat Interactions Center makes it easy to apply activations and interactions to elements without writing any code.

To assign an activation to an element:

1 Do one of the following:

- Right-click the element in the layout and choose Possible Activations
- Select the element and click the Interactions Center button on the toolbar.
- Choose Edit > Possible Activations.

The Interactions Center displays a list of all activations that can be applied to the selected element.

2 Double-click the activation or, in the Interactions Center, right-click an activation and choose Apply.

Certain activations have associated parameters such as speed, placement, color, or direction. For these activations, a Parameters dialog box opens.

3 In the Parameters dialog box, type or select values for the behavior of the activation, and click OK.

Editing activations and interactions

The activations and interactions for a page can be modified and re-ordered from the Interactions Center.

To view the list of activations and interactions assigned to an element:

Do one of the following:

- Right-click the element and choose Assigned Interactions (ordered).
- Select the element. In the Interactions Center, click the Locate Interactions button, and choose Show All Assigned (non-inherited) Interactions.

To view the complete list of activations and interactions for the current page:

Do one of the following:

- Right-click an unused space in the layout and choose Assigned Interactions (ordered).
- In the Interactions Center, click the Locate Interactions button and choose Show All Interactions on This Page.

To edit an interaction:

1. In the Interactions Center, double-click the interaction.
2. In the Parameters dialog box, edit the behavior parameters for the interaction.

To remove an interaction:

In the Interactions Center, right-click the interaction and choose Remove.

Interaction order

The interactions for the selected element or page are listed in the Interactions Center in the order in which they will be published. The interaction at the top of the list is the first published interaction. The order of interactions can be important for some functions, such as a series of interactions which relate to a submit button. You can modify the order.

To re-order an interaction:

In the Interactions Center, drag the interaction to a new position in the list.

For more information on ordering interactions, see “Changing the order of interaction events” on page 412.

Using styles

Each element can have a pre-defined style associated with it. You can define and save styles as assets available for use throughout your site. Most elements in your layout can have styles applied to them.

You can create your own Styles for re-use throughout your site, create global styles for use in other sites, and create custom styles for single elements. For more information, see “Using styles” on page 128.

To view a list and a preview of the available styles:

1. In the Asset Center, click the Locate Assets button. Choose Styles > Show All Styles.
2. Click a style name to preview its appearance.

   A sample of text displayed with the attributes of the selected style is displayed in the Asset View window below the Asset Center scroll list.

To apply a pre-defined Style to one or more elements:

In the layout, select elements and do one of the following:

1. In the Attributes tab, choose a style name from the Style pop-up menu.
2. Drag a style from the Asset Center onto the selection in the layout.

Note: Styles are not applicable to certain SmartElements. If the style attribute is not available for a selected element, the Style pop-up menu is not included on the Attribute tab for that element.

Using AutoTables to organize data

Drumbeat can automatically organize data from a Content Table into a tabular page element called an AutoTable. AutoTables are populated from a single Content Table; additional content cannot be added to the AutoTable. (AutoTables are different from standard HTML tables, in that they are used exclusively for organizing content, not positioning elements. For information on the Drumbeat table positioning for older browsers, see “Browser support” on page 17.)

Creating an AutoTable

AutoTables are created with the AutoTable SmartElement, and populated with data from a Content Table.

To create and populate an AutoTable:

1. Drag an AutoTable SmartElement from the SmartElements toolbar to the layout.
2. In the Attributes tab for the AutoTable, from the Content pop-up menu select Static Content for content that will come from a manual or CSV content table or Record set for content from an ODBC content table. (To use content from a Record set, you must first place the Record set SmartElement in the basement and bind it to the appropriate content table.)
3. Click the Assign Content button.
4. Choose the Content Table containing the data for the AutoTable from the Content Table pop-up menu.
5. From the Show These Columns list, select each column that you do not want included in the AutoTable, and click Remove.
6. Order the columns by selecting a column name and choose Up to move the column one position to the left in the AutoTable, or choose Down to move the column one position to the right.
7. When the Show These Columns List reflects, in order, the columns to display in the AutoTable, click OK.

AutoTable attributes

AutoTables have attributes that define the appearance and placement of the table on your layout.

To view or edit AutoTable Attributes:

1. In the layout, select the AutoTable and click the Attributes tab, or right-click the AutoTable and choose Attributes.
2. Set the following AutoTable attributes:

   Left and Top specify the position of the left and top edge respectively, of the AutoTable.

   Width displays the sum of the column widths and cell gaps in the AutoTable.

   Height displays the sum of the row heights and cell gaps in the AutoTable.

   Border sets the thickness of the outline around each cell.

   Cell Gap determines the amount of space between cells.

   Page Size is the size of the layout.

   Anchor indicates whether or not an anchor is created at the top of the AutoTable for referencing with links from other elements.

   Visible indicates whether or not the AutoTable is visible in its initial state.

   Headings indicates whether or not column headings are displayed.

   Content specifies the source of the content for the AutoTable.

   Column Alignments specifies the horizontal alignment (left, center, right) and the vertical alignment (top, center, bottom) for the contents of each column.

   Column Link specifies a link from a hot column to a URL specified in a URL column. Multiple hot columns can have links.

   Dynamic Column Link specifies links from hot columns to a specific record set on a Data Form Detail page, when the AutoTable is associated with a record set. When a Dynamic Link column is defined, the Maintain State option in the Link dialog box must be selected.

   Table Style specifies the style assigned to every cell.

   Odd Rows specifies the style assigned to cells in odd numbered rows of the AutoTable. The Odd Rows style overrides the Table Style.

   Even Rows specifies the style assigned to cells in even numbered rows of the AutoTable. The Even Rows style overrides the Table Style.

   Row 1 specifies the style applied to the first (header) row of the AutoTable. The Row 1 style overrides the Table style, the Odd Rows style, the Even Rows style, and the Column 1 style.

   Column 1 specifies the style applied to the first (left) column of the AutoTable. The Column 1 Style overrides the Table style, the Odd Rows style, and the Even Rows style.

   Name assigns a name to the AutoTable.

Moving and sizing an AutoTable

AutoTables can be moved like any SmartElement in the layout.

AutoTable size is determined by column width, row height, cell gap, and border thickness. Column width, cell gap, and border thickness can be set. Row height is determined by the height of the largest content in a cell on each row.

To position an AutoTable on your layout:

Do one of the following:

- In the layout, drag the AutoTable into position.
- In the layout, select the AutoTable and click the Attributes tab. Enter values for the left and top options.

To resize an AutoTable column:

In the layout, drag the right edge of the column to the new size.

To resize the gap between the cells of an AutoTable:

On the Attributes tab in the Cell Gap field, type or select the space, in pixels, between cells.

To resize the border around each cell, and around the AutoTable:

On the Attributes tab in the Border field, type or select the thickness, in pixels, of the cell and table border.

AutoTable cell alignment

AutoTable cells in each column can be aligned horizontally and vertically.

To define cell alignment by column:

1. From the Attributes tab, click ColumnAlignments Assign.
2. In the Column Content Alignment dialog box, select the name of the column to align.
3. Select Left, Center, or Right as the Horizontal Alignment.
4. Select Top, Center, or Bottom as the Vertical Alignment.

   The contents of all cells in the selected column will align with the specified edge or center of the cell.

AutoTable column links

Cells in one or more columns of an AutoTable can contain links to a URL identified in a column of a Content Table. Columns containing links to a URL Content Table column are called hot columns.

To create hot columns:

1 In the layout, right-click the AutoTable and choose Attributes.

2 Click Column Link Assign.

3 For Hot Column, select the AutoTable columns that are to link to the contents of a Content Table column.

4 For URL Column, select the Content Table column that contains the destination URLs.

When you publish the page containing the AutoTable, links are automatically created from each item in the hot column(s) to the specified URLs.

Dynamic column links

If the AutoTable is associated with a record set, AutoTable columns can contain links to a DataForm Detail page. These column links are called dynamic links. To associate an AutoTable with a record set, you must first place a record set element on the page and bind it to the appropriate ODBC content table. The Dynamic Link Column option will then be available in the Attributes tab.

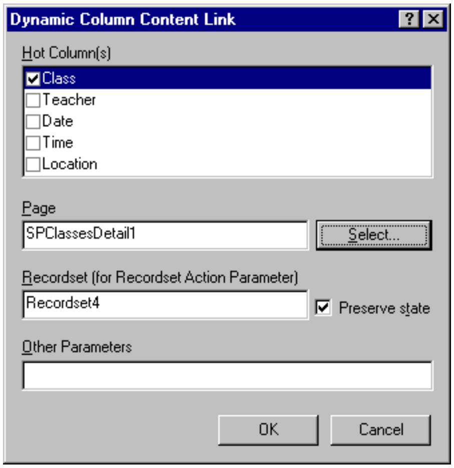
To create dynamic column Links:

1 In the layout, right-click the AutoTable and choose Attributes.

2 Click Dynamic Column Link Assign.

3 For Hot Columns, select the AutoTable columns that are to link to the contents of a Data Form Detail page.

4 Click Select Page to choose the Data Form Detail page containing the destination links.



Dynamic Column Content Link

Hot Column(s)

☑ Class
☐ Teacher
☐ Date
☐ Time
☐ Location

Page: SPClassesDetail1

Recordset (for Recordset Action Parameter): Recordset4

☑ Preserve state

Other Parameters:

CHAPTER 6
Text

Organizing and using the text components of your web site efficiently is important if you want to minimize the work involved in your site while maximizing the sharing of crucial resources. Drumbeat makes it easy to use text from a variety of different sources. External text files and text content from databases can be easily integrated. Drumbeat supports a full range of styles for text components, using both HTML attributes and cascading style sheets.

Adding text to the layout

Text for your site can come from a variety of sources in Drumbeat. You can:

- Add text directly to the page layout in in-line editing mode.
- Place text in a content table where it can be maintained for use in several places at once.
- Copy and paste text from an external file into Drumbeat.
- Use text directly from an external text file maintained in another application.
- Reference text from a database file (either CSV or ODBC).

The method you use may depend upon the amount of text you need to incorporate, where it originates, or the frequency with which it may need to be updated. The various ways of adding text to a page layout in Drumbeat as well as their respective advantages and disadvantages are shown in the following table. Each method is explained in the sections following.

Type	Method	Advantages/Disadvantages
Add text directly to the page layout with a Text SmartElement	Text element + in-line text editing mode	Advantage: The most basic way of adding text to a page. Text is entered directly into the SmartElement on the layout and can be edited easily using the in-line text editor. Disadvantage: Not efficient if the text is to be updated regularly or is to be repeated on other pages. All changes must be made within Drumbeat.
Copy text from an external file	Copy/paste to and from Clipboard	Advantage: If text has been created with a word processor or text editor, you can simply copy it into a Text SmartElement or a content table without retyping. The copied text becomes native to Drumbeat and may be edited. Disadvantage: Changes to the original external text file will not be reflected within Drumbeat or vice versa.
Reference an external text file	Asset Center or Windows Explorer	Advantage: Text can be created, edited, and maintained externally to Drumbeat and then referenced within Drumbeat. Disadvantage: The text is read-only. Changes must be made to the original file. Care must be taken to ensure that only authorized changes are made to the external text file.
Use text from a manual content table	Content table cell + text editor	Advantage: Text can be used repeatedly on other pages. If you change text within a content table cell, those changes are reflected in every occurrence of that text. Disadvantage: If you change text within a SmartElement referencing a content table cell, those changes are reflected in the content table cell and, therefore, in every occurrence of that text.
Use text from a CSV or ODBC database source	PageSets or Active Server Pages	Advantage: Text can be updated on numerous pages at once when the database changes, either by republishing PageSets or instantly on-the-fly with Active Server Pages. Disadvantage: Requires familiarity with database management, and administrative and security controls, as necessary.

Adding text directly to the layout

Text can be added directly to the layout with the Text SmartElement and the in-line text editing mode. This method is best for short sections of text that you do not foresee the need to reuse in other areas of your site and that will not be changed often.

To add text directly:

1 Do one of the following:

- Select the Text SmartElement on the SmartElement toolbar and drag and drop it onto the layout or click the layout to drop it in the desired location.
- Select the Text SmartElement, then click the layout and hold the mouse down as you draw a box of the desired width.

2 Click within the text container to activate the in-line text editing mode. The Text Formatting toolbar will appear floating on the screen. (You can add it to the main toolbar display by selecting View > Toolbars > Text Formatting. To dock the toolbar, hold down the mouse key on the title bar and drag the toolbar to the top or bottom of the Drumbeat screen.)

3 Type the desired text. Use the Text Formatting toolbar to apply font styles and sizes to selected text within the text box. Use the Hotlink button on the Formatting toolbar to create links for selected text.

4 When you finish your entry, click outside the text container or close the Text Formatting toolbar to turn off in-line text editing mode. The Formatting toolbar will also disappear (unless you choose to add it to your permanent toolbar).

The text can then be edited and reformatted at any time by clicking within the text box to activate the editing mode again.

Copying text from an external source

Text can be copied from an external text file and pasted directly into a Text SmartElement on the layout or into a content table cell. The copied text becomes native to Drumbeat, where it can be edited and formatted using the in-line text editing mode, if desired.

Note: If the text you wish to copy into the layout contains HTML code that you want to preserve, place the HTML Passthrough element on your layout and drop the text on the HTML Passthrough. HTML is not preserved within the Text SmartElement. For more information on HTML Passthrough, see “SmartElements” on page 213.

To copy and paste text from an external file:

1 In the external application, select the text to copy.

2 Copy the text to the Windows clipboard by selecting Edit > Copy or pressing Ctrl-C.

3 Do one of the following:

- Within Drumbeat, select a Text SmartElement on the layout and click within it to invoke the in-line editing mode.
- Double-click a content table cell to invoke the text editor.

4 Paste the text from the clipboard into the text element or the Text Editor using Edit > Paste or Ctrl-V.

To drag and drop external text:

1 Open Drumbeat and the external text application in side-by-side windows on your screen.

2 Within the external application, select the desired text.

3 While holding down the Ctrl key, drag the selected text directly to the layout, to a Text SmartElement, or to a content table cell (the cell type should be Unspecified).

When you copy text from another application, the original text formatting is preserved as far as possible. You may reformat the text in the Drumbeat text editor if necessary.

Using external text files in other formats

In addition to allowing you to add text directly to the page, Drumbeat also allows you to reference an external text file in RTF (Rich Text Format) or a plain text file (TXT or ASC).

Drumbeat will reference the external text file each time the page is published. If the file has changed, the new content will be inserted. Each time your site is published, the most current version of the file will be published to the server. This is particularly useful if the text is updated on a regular basis and is used in other ways external to the web application.

Since the referenced text within Drumbeat is read-only, you must make all changes to the original text file. If you click on an external-referenced file to edit it, the associated editing application (Notepad or Word, for example) will open the file for editing.

When you use content in other text formats, Drumbeat preserves that formatting as far as possible. If you find that formatting changes are necessary, edit the file in the original application.

A convenient way to add external files such as text to your web pages is to use the Asset Center. When you query for text files in Asset Center, the files appear in the Asset Viewer and can be sorted by location, name, type, or file size.

Before querying the Asset Center for text files, you must add the directory that contains the files you want to use, or copy the file to an existing media folder for the site in Drumbeat. For more information on setting up media folders and using the Asset Center, see "Using Asset Center" on page 54.

To query the Asset Center for text:

1. In the Asset Center, click the Locate Assets button.
2. Select Media > Show All Images.
3. Click the Changeable value (the underlined word images), and select Text from the pop-up menu.

You can also sort for text files within a specific folder.

To use an external text file from the Asset Center:

1. Select the file in the Asset Center list.
2. While holding down the mouse key, drag the file directly to the layout, to a Text SmartElement on the layout, or to a content table cell.

To add an RTF or TXT file to a content table cell using Windows Explorer:

1. In the Column dialog box, set the Column type to Media File and the Media Type to Document.
2. Double-click the cell and in the Media Properties dialog box, browse to locate the file.

To add an RTF or TXT file to directly to the layout:

1. Place a Text SmartElement on the layout, and either click the Attributes tab or right-click and choose Attributes.
2. In the Attributes tab, select File from the Content pop-up menu.
3. In the Media Properties dialog box, browse to locate the file.

To drag and drop an RTF or TXT file into a content table:

1. In the Column dialog box, set the Column type in the content table to Unspecified.
2. Do one of the following:

- Open Windows Explorer in side-by-side windows with Drumbeat. Locate the file in Windows Explorer and drag it into the content table cell.
- Drag the file onto the Text SmartElement on the layout.

Note: When you paste or drag external text files into a content table cell, you must either set the cell Type to Unspecified or set the cell type to Media File and Media Type to Document or Unspecified.

Using text from a content table

Text can be placed in a content table cell and then used on your page by referencing the content table. The advantage to this method of adding text is that the same text can be used on many different pages. When you change the text in the content table, the changes will ripple through the site wherever that text has been used. Content tables can be populated with content manually, or from CSV or ODBC database files (see “Using Text from CSV or OBDC database sources” on page 120).

The following instructions pertain to using text from manual content tables. For instructions on creating manual content tables, see “Manual content tables” on page 201.

Once you have created a content table to hold site assets, such as text, that you want to use on your site, you can define text columns and enter text into the cells.

To define a text column in a content table:

1 Double-click the column header or right-click the column header and choose Properties.

2 In the Type pop-up menu, select Text.

3 In the Text Type pop-up menu, select Formatted if you want the text to retain any applied formatting or Unformatted if you do not want the text to be formatted.

Note: Text that references a URL or file path must be of text type Unformatted.

To enter text directly into a cell:

1 In the Content Center, click the cell in the content table and type directly into it. (If you need to insert a tab, use CTRL+Tab. Pressing the Tab key shifts focus to the next cell.)

2 When you finish entering text, press Enter.

The in-line editing mode in a content table cell is suitable only for a very small amount of text that you do not want to format. For larger amounts of text, use the text editor. The default text type when entering text directly into a cell is Formatted.

To enter Text into a content table cell using the text editor:

1 In the Content Center, double-click the chosen cell or right-click the cell and choose Insert from the pop-up menu.

2 Enter the text (either directly or by cutting and pasting from another source). Use the Text Formatting toolbar on the text editor to format the selection as desired, then click OK.

The Text Editor can be reinvoked for editing the text.

To edit text in a content table:

Double-click the cell or click any text SmartElement on the layout that contains the text.

Note: If you change the referenced text on any page, those changes are saved in the content table cell and will therefore be reflected everywhere that cell is referenced on your site.

Once text has been placed in a content table, it can be used repeatedly in multiple Text SmartElements on the layout.

Using Text from CSV or OBDC database sources

Text from an external CSV (comma-separated values) file or an ODBC data source can be used in your web site to provide automatically updateable content from a database. To use text from database sources, you must first create a CSV or ODBC content table in Drumbeat.

CSV content tables are created from CSV files exported from another application, such as a spreadsheet or a database application. For information on creating CSV content tables, see "To create a content table from a CSV file" on page 202 and "Static PageSets with manual or CSV content tables" on page 276.

Once you have created a content table from a CSV file, you can use text from a content table cell on the layout.

To use text from a CSV content table:

1 Place a Text SmartElement from the SmartElements toolbar on the layout.

2 With the Text SmartElement selected, click the Attributes tab or right-click and choose Attributes.

3 In the Content pop-up menu, select Static Content. The Content dialog box will appear automatically. (You can bring it up manually to make changes by clicking the ellipses button next to the Content pop-up menu.)

4 In the Content dialog box, choose the content table and select the appropriate row and column where the text resides.

Note: If a CSV file on which you based a content table has changed, the changes will not be reflected in the content table, unless the CSV file is referenced by DSN and the content table has then been created as an ODBC content table. For information on creating a DSN for a CSV file, see "Static PageSets with manual or CSV content tables" on page 276.

ODBC content tables are created from SQL queries to a database. For information on creating this kind of content table, see "Creating an ODBC content table" on page 298.

You can use text in an ODBC content table anywhere on your pages, regardless of whether you choose to use the DataForm wizard to create DataForm pages. However, in order to use this feature you must have ASP server support enabled in Drumbeat Preferences and be publishing to an ASP-compatible server.

To use text from an ODBC content table in a layout:

1 Drop a Recordset SmartElement from the SmartElement toolbar onto the page. The Basement will open and the recordset element will appear in the Basement.

2 With the recordset selected, click the Attributes tab. Bind the content of the recordset to the appropriate content table by choosing the content table from the pop-up menu.

3 Place a Text SmartElement on the layout. With the Text SmartElement selected, click the Attributes tab, or right-click and choose Attributes.

4 Click the Attributes tab, and in the Content pop-up menu choose Recordset. The Content dialog box will appear automatically. (You can bring it up manually to make changes by clicking the ellipses button next to the Content pop-up menu.)

5 In the Content dialog box, select the recordset from the pop-up menu and select the column from which the data will come.

To make the content of the Text element change to reflect different records in the database, you will need to set up interactions to navigate records, or use a data loop to place more than one record on a page. For more information on working with ODBC database-driven pages, see Chapters 17 and 18.

Using the Text SmartElement

The Text SmartElement on the SmartElement toolbar is the container for all the different kinds of text. Whether you are typing text directly into the layout, or bringing it in from an external file, all text will reside within one or more Text or AutoTable SmartElements on the layout. You can use a variety of attributes to tailor the presentation for different browser support, use a full range of styles, and set up interactions between a Text SmartElement and other elements.

Text SmartElement attributes

Attribute	Description
Left	Specifies the leftmost position (in pixels) of the text box on the page layout. Along with the Top attribute, this defines the absolute positioning of the element.
Top	Specifies the topmost position (in pixels) of the text box on the layout. Along with the Left attribute, this defines the absolute positioning of the element.
Width	Displays the width (in pixels) of the element. The width may be changed directly in the Attributes tab by entering a new value or by dragging by the handles on the layout.
Height	Displays the height (in pixels) of the element. This is calculated automatically based on the specified width and font size and is a noneditable attribute.
Anchor	When selected, creates an anchor name that allows you to link to the element from a hot link. The element will then be available in the Anchor pop-up menu in the Link Selection dialog box.
Visible	Specifies initial visibility of the element (which can be changed with interactions and scripting).
Preserve Whitespace	When selected, this box ensures that spaces within the text element are treated as characters and will be preserved in the layout. Text is treated as it is with the <PRE> tag in HTML and will be rendered in the default monospace type. (Styles may be applied; the font will not change, but the font weight, style, and color will take on the style property.)
Content (Assign)	Used to specify content for the element from an external file, a content table, a recordset, or a server expression.
Style	Defines a style for the element, which can be one of several predefined styles that ship with Drumbeat, a special style created for the site, or a custom style for this element only. Styles include font face, size, weight, color and decoration, background color or image, and border attributes.
Name	Displays the name assigned to the element. You can use the default names or rename elements to facilitate identification and scripting.

To view the attributes of a Text SmartElement:

Select the text element on the layout and click the Attributes tab, or right-click and choose Attributes.

Adding, positioning, sizing, and renaming Text SmartElements

To add a Text SmartElement to your page:

1 Select the Text SmartElement on the SmartElements toolbar.

2 Do one of the following:

- Drag and drop the Text SmartElement to the layout while holding down the right mouse key.
- Click anywhere on the layout to drop the element.
- Click anywhere on the layout and hold down the right mouse key as you draw a box of a desired size.

Note: A text element will be automatically placed on the page as a container when any external text file is dragged to the layout.

A text box has a default width of 250 pixels (unless you choose the marquee-draw method to place the box initially). The height defaults to accommodate one line of text. The actual height of the text box will adjust to fit the content when the content is assigned. The width of the element can be adjusted using the two sizing handles on the text box, or by entering a width value in the Attributes tab for the element. (The height value is read-only, as it auto-adjusts to accommodate the text.)

To position a text SmartElement on the page:

Do one of the following:

- Select the element and drag and drop it to the desired position.
- Marquee-select a group of elements by holding down the right mouse key and drawing around them on the layout with the Selection tool. Then use the mouse to drag and drop the elements as a group.
- Cut or copy the element(s) and paste them to another page or template layout.
- Nudge the element with the arrow keys.
- Enter top and left positioning attributes in the Attributes tab.

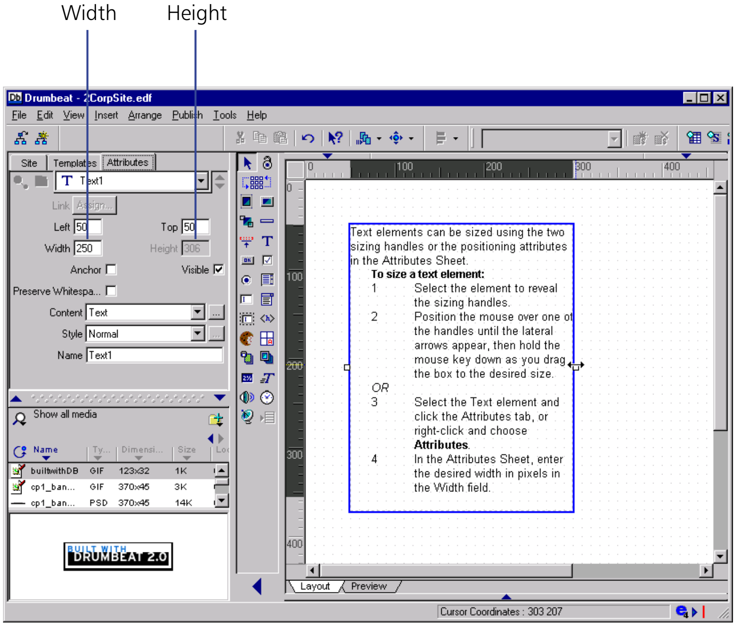
To size a Text SmartElement:

1 Select the element to reveal the sizing handles.

2 Do one of the following:

- Position the pointer over one of the handles until the lateral arrows appear, and then drag the box to the desired size.
- Select the text element and click the Attributes tab, or right-click and choose Attributes.

3 In the Attributes tab, enter the desired width in pixels in the Width field.



Text1

Left 50 Top 50

Width 250 Height 306

Anchor □ Visible ☑

Preserve Whitespa... □

Content Text

Style Normal

Name Text1

Name	Ty...	Dimensi...	Size
builtwithDB	GIF	123x32	1K
cp1_ban...	GIF	370x45	3K
cp1_ban...	PSD	370x45	14K

BUILT WITH DRUMBEAT 2.0

Text elements can be sized using the two sizing handles or the positioning attributes in the Attributes Sheet.

To size a text element:

1 Select the element to reveal the sizing handles.

2 Position the mouse over one of the handles until the lateral arrows appear, then hold the mouse key down as you drag the box to the desired size.

OR

3 Select the Text element and click the Attributes tab, or right-click and choose **Attributes**.

4 In the Attributes Sheet, enter the desired width in pixels in the Width field.

Cursor Coordinates : 303 207

Resize a text box by dragging the handles, or enter a width in the Attributes tab. The height of the text box automatically adjusts to fit the content.

Note: If you resize the text element horizontally, the new height will be calculated automatically. The height is shown in the Attributes tab as a read-only value, as it auto-adjusts to accommodate the content.

When you add a text element to the layout, it is assigned a default name, such as Text1 (with incremental numbers for new elements). You can rename it to something more meaningful to facilitate identification and for scripting purposes.

To rename a text element:

1. Select the element and click the Attributes tab, or right-click and choose Attributes from the pop-up menu.
2. Enter a new name in the Name field in the Attributes tab.

Editing and formatting text

Text can be edited in Drumbeat using either in-line text editing or the text editor.

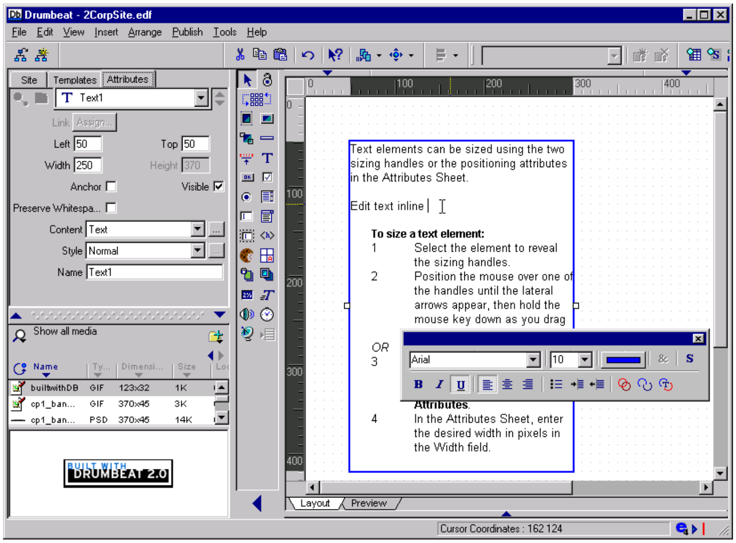
In-line text editing

You can use the on-line text editing mode to add and edit original text directly within a Text SmartElement on the layout or in a content table cell.

To activate the in-line text editing mode:

1 Do one of the following:

- Double-click the text box on the layout you want to edit.
- Select the text box and press Enter.
- Select the text box and press F2. (The cursor is inserted at the beginning of the text.)



Text1

Left 50 Top 50
Width 250 Height 370
Content Text
Style Normal
Name Text1

Name	Ty...	Dimensi...	Size
builtwithDB	GIF	123x32	1K
cp1_ban...	GIF	370x45	3K
cp1_ban...	PSD	370x45	14K

Text elements can be sized using the two sizing handles or the positioning attributes in the Attributes Sheet.

Edit text inline |

To size a text element:

1 Select the element to reveal the sizing handles.

2 Position the mouse over one of the handles until the lateral arrows appear, then hold the mouse key down as you drag

OR

3

Attributes

4 In the Attributes Sheet, enter the desired width in pixels in the Width field.

Cursor Coordinates : 162 124

Click in the text box to activate in-line editing mode and bring up the Text Formatting toolbar.

2 The text cursor will appear within the box so you can add, delete, and edit text. The Text Formatting toolbar will appear floating on the screen (unless you have previously chosen to add this to your toolbar display). You can move the formatting toolbar out of your way if necessary by dragging it.

3 To deactivate the in-line text editing mode, do one of the following:

- Click anywhere outside the edit box.
- Close the Text Formatting toolbar (click the Close button on the toolbar).
- Press Escape. (Changes will not be saved.)

Using the text editor

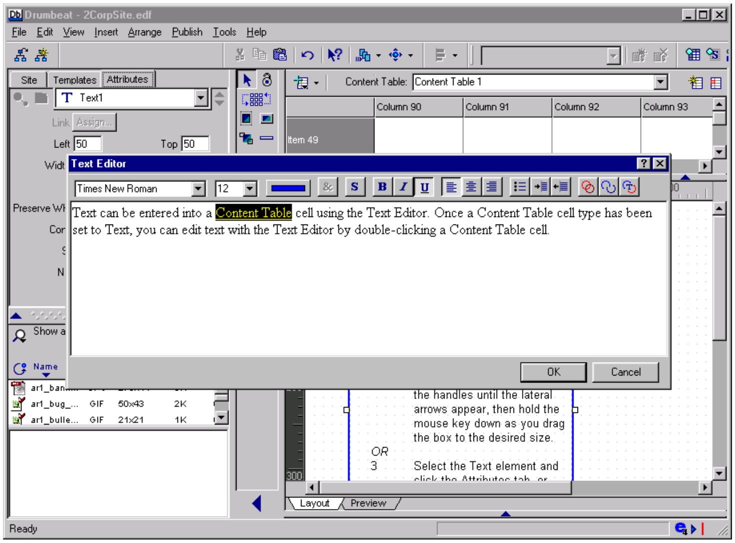
Text can be entered into a content table cell using the text editor. Once a content table cell type has been set to Text, you can edit text with the text editor by double-clicking a content table cell.

To activate the text editor:

Do one of the following:

- Double-click within the content table cell that contains the text you want to edit.
- Right-click the content table cell and select Insert (or Open, if there is existing text).

If the column type has not previously been set to Text, a warning will pop up advising you that you need to do this before entering text. The Column dialog box will then appear to allow you to set the column type, after which the text editor appears.



Text Editor

Times New Roman 12

Text can be entered into a Content Table cell using the Text Editor. Once a Content Table cell type has been set to Text, you can edit text with the Text Editor by double-clicking a Content Table cell.

Content Table: Content Table 1

Left 50 Top 50

Drumbeat - 2CorpSite.edf
File Edit View Insert Arrange Publish Tools Help
Site Templates Attributes
Text1
Link Assign...
Widt
Preserve Wh
Cor
N
Show a
Name
Column 90
Column 91
Column 92
Column 93
Item 49
OK
Cancel
ar1_ban...
ar1_bug_... GIF 50x43 2K
ar1_bulle... GIF 21x21 1K
the handles until the lateral arrows appear, then hold the mouse key down as you drag the box to the desired size.
OR
3 Select the Text element and
300
Layout
Preview
Ready

Click directly in a content table cell whose column type has been set to Text to begin entering or editing text with the text editor.

Using either the in-line editing mode or the text editor, you can add new text to the Text SmartElement, insert text within the existing block of text, or delete text as you would in a word processor or other text editor.

You can cut, copy, and paste text within the edit box using the standard Windows methods on the Edit menu.

The Text Editing toolbar within the text editor is identical to the floating Text Formatting toolbar. You can set font face, style, size, color, etc., on selected text. You can also create a hypertext link from any section of text by selecting it and clicking the Assign Link button. Edit link text by selecting it (by placing your cursor anywhere within the link text) and clicking the Link Text button.

Formatting that is applied in the text editor in in-line text editing mode is considered content-level formatting. Styles will either override or merge with container-level formatting applied to the SmartElement. For more information, see “Style inheritance” on page 129.

Using styles

Sites created in Drumbeat take advantage of CSS (cascading style sheets) and Dynamic HTML Styles. A style includes various text attributes (e.g., font face, size and color), as well as other CSS-defined style properties such as background color, box margin, and border width and color.

A powerful advantage of using styles is that you can quickly change the look and feel of your site by modifying styles that apply to multiple elements. The style changes will take effect wherever that style is applied throughout your site.

Styles can be created to apply to a single site, or they can be made global styles, to be available to all sites—so that you can use a corporate style, for instance, on any site you create.

The published HTML depends on the target browser settings you have selected. Certain Style attributes are available only when 4.0 browsers are targeted (either through Site Preferences or browser-specific SmartPages). For Generic browsers and Netscape 3.0 settings, font tags are used, rather than styles, and the attributes are limited to those supported by HTML.

Each new site that you create always starts out with the Normal Style set as the Persistent Style. You can set a new Persistent Style by choosing a different Style from the Attributes tab of any SmartElement. Once a new style is created and applied to any SmartElement through its Attributes tab, that style becomes the current Persistent Style throughout the site until a new Persistent Style is activated. The Persistent Style applies only to text elements, not other SmartElements to which styles may be applied.

Style inheritance

Styles can be inherited from templates and higher level elements. When an element on a template is associated with a style, all individual pages and other templates within the site inherit the SmartElement complete with its style association or customized style settings.

There are two levels at which style formatting can be applied:

- Container-level formatting is applied to the entire container (the whole text box or other SmartElement).
- Content-level formatting is applied to a segment within the container (that is, a text string within a text box).

Content-level formatting will always override container-level formatting. An exception is link styles. Links cannot be formatted at the content level. Links within a text container take the style that is defined as the Link Style in the Attributes tab for the page or the template to which the page is subordinate. Styles applied to the container will be merged with the link style, as described in "Setting link styles" on page 136.

All other styles assigned to an element can be overridden by content-level formatting applied with the Text Formatting toolbar while editing a text element (see "Removing and overriding an applied style" on page 133). Any style properties not assigned at the content level will be inherited from the higher level style.

Styles can also be assigned to nontext elements. However, not all style properties will apply to those elements. Only the attributes appropriate for that particular SmartElement will be applied. For example, text styles will not apply to an image element, even though these attributes are defined in a style associated with that SmartElement. However, applying the style allows other common properties, such as border colors on image links, to be applied to all elements.

Text style properties

Property	Description
Font	Specifies the font (or font face) to be used, from any of the available fonts resident on your system. (The font must also be available on a user's computer in order to be seen as intended.)
Font size	Font size expressed as an absolute point size.
Font style	Normal, italic or oblique
Font weight	Density of font expressed as a value from 100 to 900. The usual normal font weight is 400, while boldface is normally 700.
Font color	Font color (or foreground color), as chosen from the default Windows palette, from the color spectrum, or by RGB value.
Text decoration	Underline, line-through, or none (available for 4.0 browsers and later only)
Text alignment	Left, center, right or none
Background	Background color or background image, applied to the box (or container) (4.0 browsers and above only)
Box properties	Properties that apply to the box (or container) for the element, including margin, border, border color, and border style (4.0 browsers and later only)

Font face and size

The default font style in Drumbeat is Times New Roman 12 point, which is the default setting installed in most browsers. However, you may change the font and point size freely for any fonts installed on your system.

Font sizes are absolute and specified in points in Drumbeat. As an aid in converting HTML default sizes to points, common browser default point sizes for HTML tags are shown in the following table.

HTML tag	Default font size for most browsers
<H1>	24 point bold
<H2>	18 point bold
<H3>	14 point bold
<H4>	12 point bold
<H5>	10 point bold
<H6>	8 point normal
<P>	12 point normal
<BIG>	13 point font
<SMALL>	9 point font
<PRE>	Courier New 10 point

Applying styles

There are several ways to format and apply styles to text in Drumbeat:

- Select a style from the Asset Center and drag and drop it onto a selected text element or onto several elements at once (see “Removing and overriding an applied style” on page 133).
- Apply a style to a text element through the style selector in the Attributes tab (see “Removing and overriding an applied style” on page 133).
- Define a custom style for a text element through the style selector in the Attributes tab (see “Custom styles” on page 136).
- Use the Text Formatting toolbar to format selected text within the text editor or in-line text editing mode (see “Text editing within Drumbeat” on page 139).

You can also define your own styles for site-wide use with the style builder. These styles will then be available in the style selector in the Attributes tab or in the Asset Center.

To apply a style using the Asset Center:

1. Select the element or elements to which you want to apply a style.
2. In the Asset Center, click the Locate Assets button and select Styles > Show All Text Styles.
3. Select a style in the Asset Center list. A sample of the text style will appear in the Asset View window. You can view the available styles by moving down or up the list with the arrow keys.
4. Drag and drop the style you want onto the selected text element or elements on the layout.

The new style will be applied to all text within the SmartElement. New text added to the element will also inherit the style.

Note: You can apply a style to several elements at once by selecting multiple elements on the layout and dragging the style from the Asset Center to any one element in the group.

To apply a style using the Attributes tab:

1. Select the text element to which you would like to assign the new style and click the Attributes tab, or right-click and choose Attributes.
2. In the Attributes tab, select a style from the Style pop-up menu.

The style will be applied to the entire text element. New text entered in the text element will also inherit the style. The style assignment will replace any previously applied style for the text element.

Removing and overriding an applied style

After assigning a style to a Text SmartElement, you can remove that style by assigning a new style.

To assign a new style to a Text SmartElement:

1. Select the text element on the layout and click the Attributes tab, or right-click and choose Attributes.
2. In the Attributes tab, select a new style from the Style pop-up menu, or select Custom and define a custom style for the selection (see “Custom styles” on page 136).

You can override a style applied to a text element and modify one or more properties for a segment of text in that container by applying styles with the Text Formatting toolbar.

To override content-level style properties applied to a Text SmartElement:

1. Select the text that you want to modify.
2. Using the Text Formatting toolbar, change the font, font size, color, or other properties as you wish.

To remove content-level style formatting and return text to the element-level style:

1. In the in-line editing mode, select the text that you want to modify (or select all the text to remove all content-level styles).

S

2. On the Text Formatting toolbar, click the Style button.

The selected text will return to the style assigned to the Text SmartElement. Any customized formatting will be removed.

Note: If you remove formatting from a text element that comes from a content table cell, those changes are reflected in the content table, as well as all other text elements that reference that content.

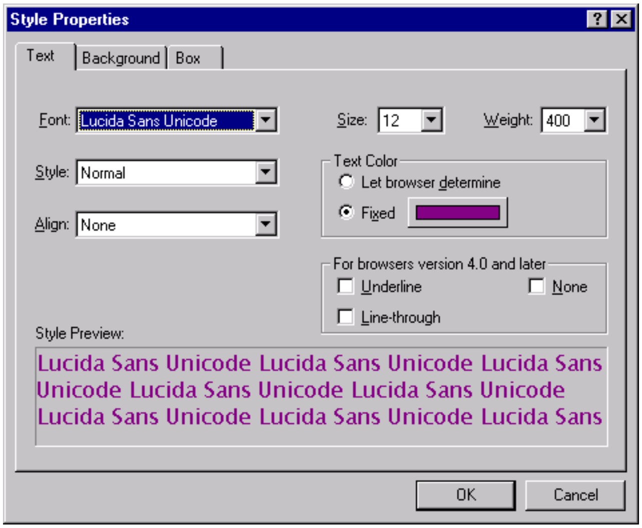
Creating new styles

To create a new style:

1 From the Tools menu, select Tools > Styles or click the Add Assets button and choose Styles.

2 In the Styles dialog box, click New to begin creating a new style. (The style will be based on the style that is currently selected in the Styles dialog box. To save time, select an existing style similar to the one you want to create before clicking New.

3 In the Style Properties dialog box, define your new style. In the Text tab, change the text attributes of the new style to suit your needs. An example of the new style will appear at the bottom of the dialog box in the Style Preview area. Set Box and Background properties in those tabs if desired. When you're done, click OK.



Font: Lucida Sans Unicode

Size: 12

Weight: 400

Style: Normal

Align: None

Text Color
○ Let browser determine
● Fixed

For browsers version 4.0 and later
□ Underline
□ None
□ Line-through

Style Preview:
Lucida Sans Unicode Lucida Sans Unicode Lucida Sans Unicode Lucida Sans Unicode Lucida Sans Unicode Lucida Sans Unicode Lucida Sans Unicode Lucida Sans

Define the properties of your new style.

4 In the Style: Save As dialog box, enter a name for the new style.

5 To add this style to your global style list, select Make Style Available Throughout All Sites. If the style is relevant only to the current site, keep the default selection, Make Style Available for This Site Only.

6 Click OK to close the dialog box and save the style with the assigned style name. Create another style or click OK again to close the Styles dialog box.

New styles appear in the Styles pop-up menu in the Attributes tab, where they will remain available for assignment to any appropriate SmartElement.

To modify an existing style:

1 From the Tools menu, select Tools > Styles.

2 In the Styles dialog box, select the Style you want to change and click Modify.

3 In the Style Properties dialog box, modify the style properties as you wish. When you're done, click OK.

Exporting and importing styles

You can exchange styles between Drumbeat site files.

To export a style:

In the Style builder, select the style or styles and click Export. Name the exported style file. It will be saved with the .css extension.

To import a style:

In the Style builder, click Import. Navigate to the folder that contains the exported style file (.css), select it, and then click OK.

Changing style scope (global or local)

You can change the scope of style from global to local or vice versa. You may want to change the scope of a style when you are sharing Drumbeat site files with another developer. If you want to re-use in other sites a style that was originally saved as a local style, you can change the style to global.

Since only local styles are saved with the site file, you can change global styles you have created to local so that they will be saved with the site file and will be transferred along with the site to another developer.

To change the scope of a style:

1 Do one of the following:

- Click the Locate Assets button and choose Styles > Show All Styles. Right-click the style and choose Modify.
- Select an element on the layout that has the style you want to change. In the Attributes tab, click the ellipses button next to the Style box.

2 Click the Scope tab and change select Global or Local. Click OK.

Note: The Scope tab does not appear if you select a style in the Style builder. You must use one of the two methods above to change the scope of a style.

Custom styles

If you want to create a style for a single element but do not anticipate reusing the style elsewhere, you can create a Custom Style for the element. When you use this method, the SmartElement has its own set of custom style properties, which will either override or merge with styles set on a template or other higher element (see "Style inheritance" on page 129).

To apply a custom style to a single element:

1. Select the text element to which you want to apply the style and click the Attributes tab or right-click and choose Attributes.
2. In the Attributes tab, select Custom from the Styles pop-up menu.
3. In the Style Properties dialog box, modify the style attributes as desired.
4. Click OK to close the Style Properties dialog box and apply the custom style settings to the selected SmartElement.

Setting link styles

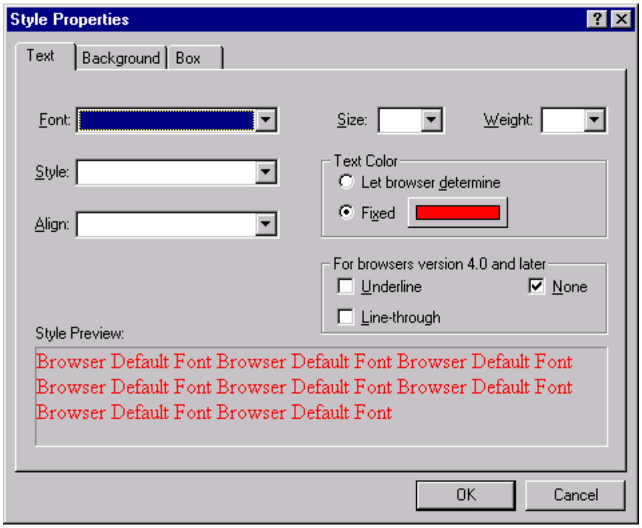
Link Styles are a special case in the style inheritance scheme, since the styles are set at the page or template level through the Attributes tab of the page or template. Link Styles cannot be applied at the content level but only at the document or container level. Often a combination of the two may be desired.

If you anticipate that links will be created using different fonts and sizes on your pages, you can set minimal Link Styles on the Master Template or any subordinate template or page, specifying only those properties you want to be uniform, such as text color and text decoration. Other style properties applied with styles at the container level will then be merged with the Link Style.

You can customize the global styles LinkColor, VLinkColor, ALinkColor, and HLinkColor, which are provided with Drumbeat for this purpose. These styles contain only color and font decoration properties, which may be changed in the Styles dialog box.

To customize link colors:

1. In the Attributes tab for the page or template, choose the LinkColor style from the Link Style pop-up menu.
2. Click the ellipses button next to the list to open the Style Properties dialog box.
3. In the text tab, click the color bar and choose the color you want to apply.



Font:
Size:
Weight:
Style:
Align:

Text Color
○ Let browser determine
● Fixed 🟥

For browsers version 4.0 and later
☐ Underline ☑ None
☐ Line-through

Style Preview:
Browser Default Font Browser Default Font Browser Default Font Browser Default Font Browser Default Font Browser Default Font Browser Default Font Browser Default Font

Choose a custom color for links. The Link Styles have blank settings for other font properties, which will be inherited from other elements.

Link underlining is turned off by default in Link Styles. You can turn underlining on or off for links with the font decoration property. If you define your own link styles, you can turn off underlining for the style if you wish.

To turn off underlining on links:

In the Text tab in the Style Properties dialog box, select None for the font decoration.

You can also create your own minimal styles that include only color settings.

To create a style for link colors only:

1. Select Tools > Styles to create a new style.
2. In the Styles dialog box, click New.
3. In the Style Properties dialog box, for the Font, Size and Weight, choose the blank entry at the top of the pop-up menu. Choose a color and specify any other properties you want for all links.

Only those properties for which you have chosen actual values will be inherited by subordinate pages. Container-level styles specifying other properties, such as font face, size, and weight (any blank properties), will be merged with those in the Link Style.

The same type of merging of styles is used in linked fields within AutoTables. Table, row, and column styles will merge with the Link Style. (See "Using AutoTables to organize data" on page 107.)

A text element whose only applied style has blank font and size properties will display in the browser with the browser defaults. (It will display in the Drumbeat layout as the Normal font, which is the Drumbeat default.)

You can also use minimal styles if you want the browser to override all font styles. This can be useful if you want specialized style sheets or browser settings defined by users to override author styles.

To allow the browser to determine all styles in the site or on a page:

1. Select the Master Template or any subordinate template or page on which you want to set browser override, and click the Attributes tab.
2. In the Attributes tab, for Link Style, select Custom and click the ellipses button to the right of the list.
3. In the Style Properties dialog box, for all properties in the Text tab, choose the blank entry at the top of the pop-up menu. (For Align, leave at None.) For Text Color, select Let Browser Determine.

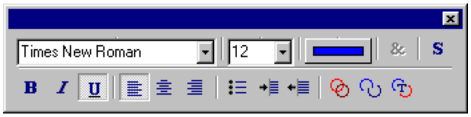
For a working example of link styles and link inheritance, and how to accommodate large font settings with styles, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

Text editing within Drumbeat

Text placed directly in a Text SmartElement or a content table in Drumbeat can be edited with the text editor. Small portions of text can also be edited and formatted in-line.

Using the Text Formatting toolbar

Text styles can be applied directly to sections of text with the Text Formatting toolbar, either in in-line editing mode or using the text editor. This method of formatting is not as powerful as using styles, as the range of attributes available is limited to conventional HTML attributes and any formatting is applied only to the specific text and cannot be reused elsewhere. However, it is useful for short sections of text.



Format inline text with the Text Formatting toolbar.

Note: Formatting text using the Text Formatting toolbar overrides any styles applied to the text at the element, page, template, or site levels.

Assigning text attributes with the Text Formatting toolbar

To set the font:

1 In the in-line text editing mode, select the text or line for which you want to change font type.

2 Select a font from the Font pop-up menu on the Text Formatting toolbar.

To set the point size:

1 In the in-line text editing mode, select the text or line for which you want to change the point size.

2 Select a point size from the Font Size pop-up menu on the Text Formatting toolbar.

To change text color:

1 In the in-line text editing mode, select the text or line for which you want to change text color.

2 Click the Text Color button (the black bar) on the Text Formatting toolbar.

3 In the Color dialog box, select a color from the default palette or click Define Custom Colors to choose a custom color from the color spectrum or by RGB values.

To apply bold, italics, or underline:

1 In the in-line text editing mode, select the text you want to underline or make bold or italic.

2 Click the Bold, Italic, or Underline button on the Text Formatting toolbar.

Text is by default aligned left. You can change text alignment to center or right, or reapply left justification.

To align a block of text:

1 In the in-line text editing mode, select the block of text you want to align.

2 Click the Align Left, Align Center, or Align Right button on the Text Formatting toolbar.

Creating bulleted lists

Bulleted lists can be created by applying list formats to the text with the Text Formatting toolbar.

To create a bulleted list:

1 In the in-line text editing mode, highlight the text to be bulleted.

2 Click the Bulleted List button on the Text Formatting toolbar.

Drumbeat automatically sets a hanging indent and adds the bullets in front of each paragraph in the selected text; it also adds bullets to each new paragraph that you type in the selected text.

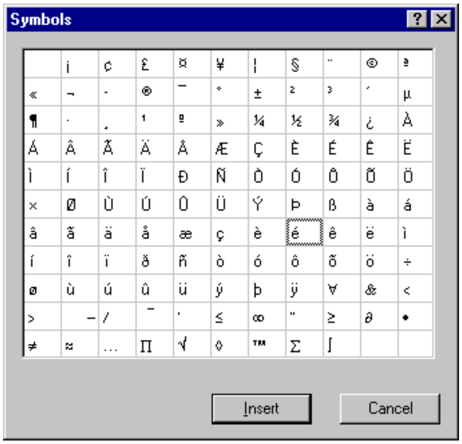
Inserting special characters

You can insert special characters and symbols into text, such as accented characters, foreign language or international characters, currency symbols, and fractions.

To insert a special character or symbol:

1 Click the Symbol button on the Text Formatting toolbar.

2 Select the symbol you want to insert and click Insert.



	¡	¢	£	¤	¥	¦	§	¨	©	ª
«	¬	-	®	¯	°	±	²	³	´	µ
¶	·	¸	¹	º	»	¼	½	¾	¿	À
Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë
Ì	Í	Î	Ï	Ð	Ñ	Ò	Ó	Ô	Õ	Ö
×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß	à	á
â	ã	ä	å	æ	ç	è	é	ê	ë	ì
í	î	ï	ð	ñ	ò	ó	ô	õ	ö	÷
ø	ù	ú	û	ü	ý	þ	ÿ	∀	&	<
>	–	/	‾	'	≤	∞	"	≥	∂	•
≠	≈	…	Π	√	◊	™	Σ	∫		

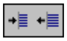
Symbols and special characters you can insert.

Indenting text

You can move the indentation point of text left or right using either the Text Formatting toolbar buttons or keyboard shortcuts.

To create an indent:

1 Select the text to be indented, or place the cursor anywhere in the line to be indented.



2 Click the Indent Right button or Indent Left button In the Text Formatting toolbar. The Indent button moves the line of text five spaces in the selected direction with each click. You can indent the line of text as much as the space within the text element will allow.

Linking and unlinking text

A hypertext link can be created on a text string from within the in-line text editor using the Link Text toolbar.

To create a link:

1 In the in-line text editing mode, select the text to be linked and click the Assign Link button.

2 In the Link dialog box, choose the page, URL, or other type of link you want to establish.

The hyperlink can be to a page, a relational page (Home, Previous, Next, etc.), a specific URL, or a file. You can edit a previously created link in the Link dialog box, by selecting a different page in the tree or entering a new URL. See “The Link dialog box” on page 74.

To edit a link destination:

1 In the in-line text editing mode, select the hyperlinked text by clicking anywhere on the link text.

2 Click the Assign Link button, which opens the Link dialog box, where you can change the link location and other settings.

To edit the link text:

1 Select the link text and click the Text Link button.

2 Edit the link text in the text editor.

To remove a text link:

In the in-line text editing mode, select the hyperlinked text and click the Undo Link button.

For more information about linking, see “Creating links” on page 73.

Deleting and undoing

To delete a Text SmartElement or a group of SmartElements from the layout:

1 Select the element or elements.

2 From the Edit menu, select Edit > Delete, or right-click and choose Cut from the pop-up menu, or press Delete.

Any links associated with the text element will be broken.

To delete selected text within a Text element in the text editor or in-line text editing mode:

1 Select the text you wish to remove.

2 From the Edit menu, choose Edit > Delete, or press Delete.

The Undo function allows you to undo recent actions. When it is activated, the most recently performed action will be undone. If a Text SmartElement, or specific text within an element, is deleted accidentally, the Undo command can be used to restore the deletion.

To undo an action:

From the Edit menu, select Edit > Undo, or click the Undo button.

The Undo command provides 20 levels of undo. This means that Drumbeat can undo up to the last 20 actions performed, in reverse sequential order.

Using Find and Replace

The Find and Replace functions in Drumbeat allow you to search for text strings, words, or phrases and replace the string with new text. You can search individual text elements, a page, or multiple pages in the site.

Note: You cannot find or replace text within text elements linked to an external text source (e.g., RTF or TXT files). A find or replace can be conducted only on text generated within Drumbeat or pasted or dropped into Drumbeat from an external source.

Finding text

With the Find feature, you can quickly locate a word or phrase within a specific SmartElement, on a single page, or on multiple pages. This may be useful if, for instance, you want to copy text from one page to another but can not remember the location of the text.

You can tailor the search to match or ignore the case (upper or lower) of the text. You can also restrict the search to whole words rather than a text string. This can increase the speed of your search.

While using the Find feature, you cannot edit text directly. You must first close the Find and Replace dialog box and then edit the text within the text element.

To find text within an element or multiple elements, or within any element on a page or template:

1 Select the Text SmartElement on the layout, or select the text elements that you would like to search on the layout by Ctrl-clicking each element, or by using the marquee-select technique. To find text within any of the elements on a page or template, click on the background (any empty area) of the page or template on the layout or select the page within the Site Manager or Template Manager (you can select multiple templates in Template Manager).

2 From the Edit menu, choose Edit > Find, or press Ctrl+F.

3 In the Find tab of the Find and Replace dialog box, type the text to search for in the Find What: Edit box.

4 In the Scope pop-up menu, choose the scope of the search. (The scope will be automatically set to Element if an element was initially selected.)

5 Select search options as desired by checking Match Case or Find Whole Words Only.

6 Click Find Next or press Enter to begin the search.

Drumbeat begins the search with the first word within the element and continues until all the text within the selected element or elements is searched. Each successive element is automatically searched in turn until all the selected elements have been searched.

The first occurrence of the text is displayed and highlighted within the context window of the dialog box. Surrounding text is also displayed, allowing you to see how the text is used within context. If the surrounding text occupies more space than available within the context window, a scroll bar appears, allowing you to scroll up and down within the text. The name of the element containing the search item and the name of its attribute are displayed in a status area at the bottom of the dialog box.

To continue the search:

Click Find Next, or press the Enter key again.

If Drumbeat cannot find any occurrence of the text, the following message appears in the status area at the bottom of the dialog box:

Search Completed. The search item was not found.

When no additional occurrences of the search item are found, the following message appears:

Search completed.

Note: If the page being searched contains text referenced to an external file, Drumbeat will not search the external file. However, it will search the path of the external file and find pathname and file name matches.

When you stop the search and close the Find and Replace dialog box, the last element containing found text remains highlighted and selected.

Replacing text

In addition to finding text, you can also replace it automatically. You can make global changes throughout your site using Find and Replace, or you can confirm each replacement separately.

The Replace feature locates matching text based on the same rules as the Find feature (you can select Match Case and Find Whole Words Only). In the case of Match Case, Drumbeat does not replicate the case of the existing text with the replaced text, but replaces it with the new text exactly as entered. Individual usages, such as initial word capitalizations, will have to be replaced separately.

Note: Before using Find and Replace to do an extensive replace operation, take a Snapshot to back up your site just to be safe.

To replace text in an element, in multiple elements, or on a page or template:

1. Select the text element on the layout, select the desired elements on the layout by Ctrl-clicking each element, or by using the marquee-select technique. To replace text on a page or template, click on the background (any empty area) of the page or template on the layout or select the page within Site Manager or Template Manager.
2. From the Edit menu, choose Edit > Replace, or press Ctrl+H. The Find and Replace dialog box appears with the Replace tab selected.
3. In the Replace tab, type the text you want to find in the Find What: Edit box.
4. In the Replace With dialog box, type the text you want to substitute for the found text.
5. In the Scope pop-up menu, choose the scope of the search. (The scope will be automatically set to Element if an element was selected.)
6. If you wish, select one or more of the options (Match Case or Find Whole Word Only).
7. Click Find Next, Replace, or Replace All to begin the search operation.

The appropriate choice, Element or Page, will be selected in the Scope pop-up menu. When Drumbeat finds the first occurrence of the specified text, you may click Replace to change the text, or Find Next to continue the search, without altering the selected occurrence. By clicking Replace All, you can immediately change all occurrences of the specified text without confirming each one.

As with searches, Drumbeat searches and replaces text within the selected element or elements, beginning with the first character of the element and continuing until the whole element and each successive element have been searched.

The name of the element containing the search item and the name of its attribute are displayed in the status area at the bottom of the dialog box. As text is replaced, the new text immediately appears both in the dialog context window and in the text element. You can monitor the changes in the text element by moving the Find and Replace dialog box out of the way.

If Drumbeat cannot find any occurrence of the search item, the following message appears in the status area at the bottom of the dialog box:

Search completed. The search item was not found.

When no additional occurrences of the search item are found, the following message appears:

Search completed.

In both of the above situations, the dialog box remains open, allowing you to enter new search and replace criteria or to close the dialog box if you wish. If you close the Find and Replace dialog box, the last element containing found text remains highlighted and selected.

If you have chosen the Replace All option, Drumbeat automatically changes each occurrence without confirmation. Text will not appear within the dialog context window, nor will the elements be highlighted as changes are made within them. When all the changes have been made, the following message appears in the status area at the bottom of the dialog box:

Search completed. # item(s) replaced.

When you have finished replacing text, close the Find and Replace dialog box by clicking Close or by closing the window. All changes made up to that point remain in effect.

Note: If the page being searched contains text referenced to an external file, Drumbeat will not search the external file. However, it will search the path of external files and find matches with the pathname and file names.

Using the spell checker

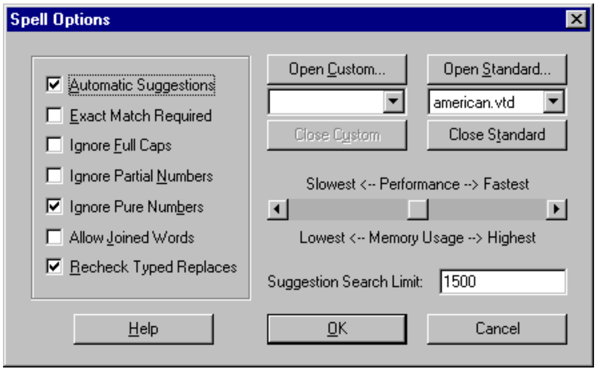
Drumbeat 2000 includes a spell checker that allows you to perform spell checks on the text used in your site. Spell checks are performed on a page basis. The spell checker searches each element on the page and returns results, with suggestions for correction.

To perform a spell check on a page:

1 Select the page on which you want to perform a spell check.

2 From the main menu, select Tools > Spell Check Page or press F7.

The Spell Checker can be customized for your requirements. You can include a custom dictionary and choose options to ignore, as shown in the following figure.



☑ Automatic Suggestions
□ Exact Match Required
□ Ignore Full Caps
□ Ignore Partial Numbers
☑ Ignore Pure Numbers
□ Allow Joined Words
☑ Recheck Typed Replaces

american.vtd

Suggestion Search Limit: 1500

Spell-checking options

CHAPTER 7
Images

Images are an important part of any good web site, because they can make your site entertaining, educational, and interesting. Use Drumbeat Image SmartElements to create static and dynamic graphics without writing HTML code.

Managing image files

Image files are managed in the Asset Center. Drumbeat creates a reference to the image file for each image used in a Drumbeat site. An image does not have to be moved or copied from its original location for use in a Drumbeat site; a reference to the file location is created in the Asset Center.

Drumbeat supports all of the most common image file formats: BMP (Bitmap), GIF (Graphic Interchange Format), and JPG (Joint Photographic Experts Group)

To view the images available for use in a Drumbeat site:

1. From the Asset Center, click the Locate Assets button.
2. Choose Media > Show All Images. The available Images are listed in the Asset Center scroll list.
3. To preview an image, select the image file name in the Asset Center. A thumbnail preview of the Image is displayed in the preview area of the Asset Center, below the scroll list.

Managing image folders

The Asset Center lists only image files that reside in folders that it recognizes as media folders. The Select Media Folders dialog box is used to manage the list of media files for this site.

To add a media folder to the Asset Center:

1. From the Asset Center, click the Add Assets button, and choose Media Folders.
2. In the Select Media Folders dialog box, click Add.
3. In the Add Media Folder dialog box, type the path and name of the folder, or click Browse to select the folder.
4. Do one of the following:

- Select Add Folder To This Site Only to add media from the selected folder to the current site.
- Select Add Folder to All Sites to add the media from the selected folder to all Drumbeat sites.

5. Select Include Sub-directories to add the selected folder and any subfolders.
6. Click OK.

To remove a media folder from the Asset Center:

1. From the Asset Center, click the Add Assets button, and choose Media Folders.
2. In the Select Media Folders dialog box, select the folder to remove, and click Delete.

To modify the settings or location of a media folder:

1. From the Asset Center, click the Add Assets button, and choose Media Folders.
2. In the Select Media Folders dialog box, select the folder to modify, and click Edit.
3. In the Add Media Folder dialog box, edit the folder definition, and click OK.
4. To confirm the media folder selections, in the Select Media Folders dialog box, click Done.

Updating media paths

Because Drumbeat relies on references to image files, Drumbeat cannot publish an image if the image file is moved from the specified media folder. If you move referenced image files, you must update the media path for all images that have already been placed on the layout, so the paths to the media will be correct.

To update a media file path:

1 From the main menu, choose Tools > Media File Paths.

2 In the Media File Paths dialog box, select the media path that has become obsolete.

3 Click Browse, and select the folder that now contains the media.

4 Click Update Path, and then click Close to exit the dialog box.

Placing images

You can place images on your layout by dragging a SmartElement container onto the layout and adding an image reference, or by placing an image directly from the Asset Center or the clipboard onto the layout, or by adding a reference to an image contained in a content table cell to the layout.

The Image SmartElements

Image SmartElements are containers for images. Each type of image SmartElement has specific behavior associated with it.

There are several types of image SmartElements:

Image	The Image SmartElement is a container for a single static image.
Image Map	The Image Map SmartElement is a container for a single image to which multiple hotspots for hyperlinks are added.
Image Button	The Image Button SmartElement is a container for the multiple images corresponding to the multiple states of a rollover image button.
DynaImage	The DynaImage SmartElement is a container for multiple dynamic images that are controlled through interactions or scripts.
Image Checkbox	The Image Checkbox SmartElement is a container for multiple dynamic images creating a graphical element with standard checkbox functionality.
Image Radio Button	The Image Radio Button SmartElement is a container for multiple dynamic images comprising a graphical button with standard Radio Button functionality.
Slide Show	The Slide Show SmartElement is a container for multiple images displayed one at a time as a slide show.
Scrolling Image	The Scrolling Image SmartElement is a container for a single image that moves in a specified direction and speed across the SmartElement container.

Using an Image SmartElement

All image SmartElements are placed using these basic steps. For specific instructions on using each type of image SmartElement, see the sections below.

To place an Image using a SmartElement:

1 Do one of the following:

- From the SmartElement toolbar, select the SmartElement and click the layout to create a container of default size.
- Select the SmartElement, and drag a container size on the layout.

2 From the layout, select the container.

3 Do one of the following:

- From the Attributes tab Content field, click the Assign Content button and in the corresponding Content dialog box or Media Properties dialog box, select the image(s) for this SmartElement.
- From the Asset Center, drag and drop the image(s) for this SmartElement.

See the following sections for details about adding content to each type of image SmartElement.

Placing images from the Asset Center

Images can be placed directly on the layout from the Asset Center.

To place an image from the Asset Center:

1 From the Asset Center, select the image file name.

2 Drag the image file name or thumbnail preview image onto the layout.

A standard Image SmartElement container is created for the image, if one does not already exist.

- If an Image is dropped onto a single-image SmartElement, the Image becomes the content for the SmartElement, replacing any existing content.
- If an Image is dropped onto a multi-image SmartElement container, the Image is added to the content for the SmartElement.
- If an Image is dropped directly on the layout, it becomes a standard single-image element. You cannot change the Image type of a dropped image to a multi-image or image button. Create a multi-image or Image Button element using the corresponding SmartElement.

Placing images from an external source

Images can be cut or copied from another page, site, or graphics application and pasted onto your layout.

To add an image from an external source:

1. Open the source page, site, or file.
2. Cut or copy the image.
3. Open your Drumbeat site.
4. Do one of the following:

- Right-click an unused space on the layout and choose Paste.
- From the main menu, choose Edit > Paste.

Using images from content tables

Images from a content table cell can be referenced to appear on your layout.

Note: Place an image from a content table when you want to use multiple occurrences of an identical image in your site, and when you want changes to the image to ripple across all occurrences of the image.

To open a content table:

1. Do one of the following:

- From the main menu, choose View > Content Center.
- From the toolbar, click the Content Center button.

2. From the content table pop-up menu, choose the content table.

For details about creating and populating content tables, see “Content Tables” on page 199.

To add an image to a content table, you must define the column for the image as an image column.

To define a column to contain images:

1. Do one of the following:

- In the Content Center, double-click the column header.
- Right-click the column head and choose Properties.

2. From the Type pop-up menu in the Column dialog box, choose Media File.
3. From the Media Type pop-up menu, choose Image.

To create multiple-image cells in this column, select Allow Multiple Content. The column header is amended with an icon indicating that the column is a media column and a plus sign (+) if multiple media can be placed in each cell. Multiple content image cells can be used to create rollover image buttons (see “Rollover image buttons” on page 161).

To place an image in a content table:

1. Do one of the following:

- From the Asset Center, drag the file name or thumbnail into the content table cell.
- Cut or copy an image to the clipboard, select the content table cell, right-click, and choose Paste.

Note: Any changes made on the layout to an image reference from a content table are also made to the image in the content table cell. The changes are therefore reflected across every occurrence of the image reference.

Referencing images by URL

An image may be referenced by URL in a content table and included on your page in this manner.

To use an image referenced by URL in a manual or CSV content table:

1. Set the properties of the content table column to Type:Text and choose the Text Type: Unformatted.
2. Enter the full URL or the relative path from the root publish directory in the cell.
3. Place an Image SmartElement on the layout. In the Attributes tab, for content, select Static Content. In the Content dialog box, select the content table and the row and column that contains the image URL.

This method may also be used when referencing images associated with a database record and included in a database from an ODBC content table. The database column must contain the path to the image—either the full path or the relative path from the root publish directory.

If you use the DataForm wizard to generate the page, the wizard will use a standard text box element to contain the URL. You need to replace the text box with an Image SmartElement and then bind the content to the recordset.

To use an image referenced by URL on a DataForm page:

1. Drop an Image SmartElement on the page.
2. In the Attributes tab, for content, select Recordset.
3. In the Content dialog box, choose the content table and the appropriate column that includes the image path.

Defining image attributes

Each type of image has a set of attributes associated with it. The following attributes are common to most images:

Link	The Link attribute defines any hyperlink from the selected Image to another page.
Left	The Left attribute defines the position in pixels for the left edge of the element.
Top	The Top attribute defines the position in pixels for the top edge of the element.
Auto Size	The Auto Size attribute specifies whether or not the element is fixed at its original size.
Width	The Width attribute defines the width in pixels for the element. Width cannot be changed if Auto Size is selected.
Height	The Height attribute defines the height in pixels for the element. Height cannot be changed if Auto Size is selected.
Border	The Border attribute defines the thickness in pixels of the border around the image.
Anchor	The Anchor attribute specifies whether or not an anchor tag is created for referencing the image from other links.
Visible	The Visible attribute specifies whether or not a dynamic image is visible in its initial state.
Content	The Content attribute defines the image file(s) that make up this image element.
Style	The Style attribute defines the pre-defined or custom style assigned to this element.
Name	The Name attribute defines the name assigned to this element.
Alt Text	The Alt Text attribute defines the text string that appears in place of the Image. The alternate text is displayed in place of the Image any time the Image cannot be displayed, and is displayed by most browsers as hover text for the Image.

For details about attributes that are specific to each Image type, see the corresponding section below.

To define attributes for an Image:

1 Do one of the following:

- From the layout, double-click the image.
- Right-click the Image and choose Attributes.
- From the Attributes tab, choose the image name from the pop-up menu.

2 From the Attributes tab, specify the content, placement, size, style, and other attributes for the image.

The attributes on the Attributes tab are specific for the type of image element selected.

For a working example of setting alternate text, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

Moving and sizing images

To move an image:

1 Do one of the following:

- From the layout, drag the image into position.
- From the Attributes tab, type or select, in pixels, the left and top positions for the left and top edges of the image.

To size an image:

1 From the layout, select the image.

2 From the Attributes tab, deselect Auto Size.

3 Do one of the following:

- From the layout, drag the top, bottom, left, right, or corner handle to resize the image.
- Select the image.

4 From the Attributes tab, deselect Auto Size. Type or select the width and height for the image.

To return an image to its original dimensions:

1 Select the image.

2 From the Attributes tab, select Auto Size.

Rollover image buttons

Drumbeat simplifies the process for creating rollover image buttons with SmartElements that display a set of multiple graphics that rollover to represent the multiple states of a button.

There are three SmartElements that create buttons with rollover effects:

- Image Button
- Image Radio Button
- Image Checkbox

The image files for rollover images are referenced from a multiple-image cell in a content table, or specified as a file list on the Attributes tab for the SmartElement.

Creating a rollover image

All rollover images are created following these steps. For details about each specific type of rollover Image, see the corresponding section below.

To create a rollover Image Button:

1 Drag the Image Button, Image Radio Button, or Image Checkbox SmartElement onto your layout.

2 Do one of the following:

- From the Content Center or Attributes tab, add content from a multiple-image content table cell into the SmartElement container.
- In the Attributes tab, assign a file list of multiple images as content for the SmartElement.
- Drag and drop the images you want to use individually from the Asset Center to the Image Button on the layout.

Adding multiple images to a content table

To add multiple images to a content table, define a column as a multiple-image column and add the image files to a cell.

To define a column to contain multiple Images:

1 Do one of the following:

- In the Content Center, double-click the column header.
- Right-click the column head and choose Properties.

2 From the Type pop-up menu in the Column dialog box, choose Media File.

3 From the Media Type pop-up menu, choose Image.

4 Select Allow Multiple Content, and click OK.

The column header is amended with an icon indicating that the column is a media column and a plus sign (+) indicating that multiple media can be placed in each cell.

To place multiple images in a content table:

1 Do one of the following:

- From the Asset Center, drag each file name or thumbnail into the content table cell.
- For each image, cut or copy the image to the clipboard, select the content table cell, right-click, and choose Paste.

A triangle is displayed in the upper right corner of each cell that contains multiple media.

Assigning multiple images from a content table

To assign images from a multi-image content table cell to a rollover SmartElement:

1 From the Content Center, select the content table containing the images.

2 Do one of the following:

- Drag and drop the contents of the cell from the Content Center to the SmartElement container on your layout.
- Select the SmartElement container.

3 From the Attributes tab, choose Static Content from the Content pop-up menu.

4 Click the Assign Content button.

5 In the Content dialog box, choose the content table, Row, and Column names from the pop-up menus for the cell containing the multiple images, and click OK.

The first image in the cell is displayed in the SmartElement on the layout.

See "Ordering multiple images" on page 164 for details about defining the order of multiple media.

Using multiple images from a content table cell to create a rollover image button facilitates maintenance if you use multiple instances of that button on your site. Any changes to the contents in the content table will be instantly reflected wherever that content is referenced in your site.

Assigning a multiple image file list

You can assign a list of files to an image button through the Attributes tab.

To assign a file list to a multiple-image SmartElement:

1 Select the SmartElement container.

2 From the Attributes tab, choose File List from the Content pop-up menu.

3 Click the Assign Content button.

4 In the Multiple Media Properties dialog box, click Select Files.

5 In the Open dialog box, select one or more image files for the SmartElement, and click Open.

Note: In the Open dialog box, Ctrl+click to select multiple files from a single folder.

6 Repeat the Select Files steps to choose image files from another folder.

The image paths and file names are listed in the Multiple Media Properties dialog box, in the default order.

Ordering multiple images

The order of multiple image content determines which image is displayed during each state of interaction with the rollover button:

- The first image in the content order is the image displayed when the element is in its initial off state. This is the default state on page load, and when the mouse is not over the rollover button.
- The second image in the content order is the image displayed during the hover state—when the mouse is over the rollover button.

By default, the images are ordered in the reverse order of selection—the first item in the multiple-image order is the last item that was added to the multiple-media contents.

To define the order of images for a rollover image button:

1 Do one of the following:

- From the Content Center, right-click the cell containing the images, and choose Show Content Order.
- From the layout, right-click the Image Button SmartElement, and choose Show Content Order.

2 From the Content Order dialog box, select a file name in the media list.

3 Do one of the following:

- Click Up to move the selected file up one position in the media list.
- Click Down to move the selected file down one position in the media list.

4 When the order is correct, click OK.

Note: Refer to the Content Preview area for a thumbnail image of the selected file.

Order can also be defined during creation of the rollover SmartElement in the Multiple Media Properties dialog box.

Creating a rollover image button

Rollover Image Buttons make effective, interesting links to other pages, sites, or bookmarks.

To create a rollover image button:

1 Drag the Image Button SmartElement onto the layout.

2 Do one of the following:

- From the Content Center or Attributes tab, add the contents of a multiple-media content table cell to the SmartElement container.
- From the Attributes tab, create a file list of the image files.
- Drag and drop images individually from the Asset Center to the image button on the layout.

See "Assigning multiple images from a content table" on page 163 and "Assigning a multiple image file list" on page 163 for details about adding content to a rollover image.

3 Define the order for the multiple images.

See "Ordering multiple images" on page 164 for details about defining the order of multiple images.

4 From the layout, select the rollover Image Button and assign a link from the button to another page, site, or anchor. See "Creating links" on page 73.

Creating rollover image radio buttons

Image Radio Buttons function like a set of standard radio buttons, with custom images in place of the usual radio button. Image Radio Buttons, like standard radio buttons, are mutually exclusive within a defined group—that is. only one Image Radio Button in a group can be selected. Selection of another Image Radio Button in the group clears any previous selection.

For image radio buttons, the order determines the button behavior as follows:

- The first image in the multiple-media order is the Off state.
- The second image in the order is the Rollover state during mouse over.
- The third image in the order is the Selected state. The Selected state remains in effect until another button in the group is selected.

To create a set of image radio buttons:

1 Drag the Image Radio Button SmartElement onto your layout.

Note: If the Image Radio Button SmartElement is not on your SmartElement toolbar, add the tool. From the main menu, choose Tools > Element Library, and on the Standard tab, select Image Radio Button.

2 Do one of the following:

- From the Content Center or Attributes tab, add the contents of a multiple-media content table cell to the Image Radio Button container on the layout.
- From the Attributes tab, create a file list of the image files.

See “Assigning multiple images from a content table” on page 163 and “Assigning a multiple image file list” on page 163 for details about adding content to a rollover image.

3 Define the order for the multiple images.

See “Ordering multiple images” on page 164 for details about defining the order of multiple images.

4 From the layout, select the Image Radio Button.

5 From the Attributes tab, type a name for this Radio Button and a group name for the group of Radio Buttons that this button belongs to.

6 Repeat all of the above steps to create at least one other Image Radio Button with the same group name but a different button name. Image Radio Buttons must belong to a group of multiple buttons.

Note that the graphical states of the radio button are, in order: 0, 1, 2. The default state is 0. When a button in state 1 is clicked, that button is set to state 2, and all other buttons in the group are set to state 0. If a button in state 2 is clicked, there is no change. One button must always be in state 2.

Creating a rollover image checkbox

An Image Checkbox functions like a standard checkbox, with custom images in place of the usual checkbox.

For an Image Checkbox, the order determines the button behavior as follows:

- The first image in the multiple-media order is the Off state.
- The second image in the order is the Rollover state during mouse over.
- The third image in the order is the Selected (Checked) state. A second click deselects the Image Checkbox, returning it to the first image.

To create an image checkbox:

1 Drag the Image Checkbox SmartElement onto your layout.

   Note: If the Image Checkbox SmartElement is not on your SmartElement toolbar, add the tool. From the main menu, choose Tools > Element Library, and on the Standard tab, select Image Checkbox.

2 From the Content Center or Attributes tab, add the contents of a multiple-media content table cell to the Image Checkbox container on the layout or, from the Attributes tab, create a file list of the image files.

   See “Assigning multiple images from a content table” on page 163 and “Assigning a multiple image file list” on page 163 for details about adding content to a rollover image.

3 Define the order for the multiple images.

See “Ordering multiple images” on page 164 for details about defining the order of multiple images.

4 From the layout, select the Image Checkbox.

5 From the Attributes tab, type a name for this Image Checkbox.

Note that the graphical states of the checkbox are, in order: 0, 1, 2. The default state is 0. When a checkbox in state 1 is clicked, it is set to state 2. When a checkbox in state 2 is clicked, it is set to state 0.

Composite images

Composite images are images composed of several smaller images. Breaking a large image up into smaller pieces makes it possible to assign different interactions to parts of an image (such as separate tool tips, status bar, messages or interactions with other images).

Creating and aligning composite images

The Arrange Elements button makes it easy to precisely align images so that they abut one another to make a larger composite image.

To align images so that they exactly abut one another:

1 Place the elements on the layout so that they slightly overlap one another, in the relative order in which they should appear.

2 From the drop-down menu on the Arrange button on the toolbar (or from the Arrange menu), choose Space Evenly Across or Space Evenly Down.

You may want to first align the elements left, right, top, or bottom, depending on their intended orientation.

For a working example of using a composite image to create separate rollover effects with other images, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

Image maps

An Image Map is an image that has been customized to include hotspots—defined areas with links to other pages, sites, or anchors.

To create an image map:

1 Drop an Image Map SmartElement onto the layout.

2 From the layout, select the Image Map container.

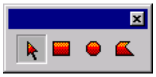
3 From the Asset Center, Attributes tab, or a content table, select an image file for the Image Map.

The image is displayed in the Image Map container.

4 From the Attributes tab, assign the Image Map element a name.

To add hotspots to the image map:

1 From the layout, double-click the image. The Image Map toolbar opens.



2 Select the tool that most closely approximates the shape of your first hotspot.

- Rectangle: To create a rectangular hotspot.
- Circle: To create a circular hotspot.
- Polygon: To create an irregularly shaped, multi-sided hotspot.

3 On the image, drag out the shape of the hotspot.

- For rectangular hotspots, click the top left corner of the hotspot and drag to the bottom right corner.
- For circular hotspots, click the center of the hotspot and drag outward.
- For polygonal hotspots, click a starting point, then click each vertex in sequence to create the polygon.

The hotspot outline appears superimposed over the image.

4 Repeat these steps for each hotspot on the image.

To customize the size and shape of a hotspot:

1 With the image selected, click to select a hotspot.

2 Drag a selection handle to resize the hotspot.

To move a hotspot:

1 With the image selected, click to select the hotspot.

2 Move the cursor to an area within the bounds of the hotspot. The cursor changes to a hand shape.

3 Drag the hotspot to a new position on the image.

Assigning links to hotspots

Create a link for each hotspot on your Image Map. The link will be activated when a user clicks anywhere in the hotspot.

To assign links to the hotspots:

1. With the Image Map selected, double-click a hotspot.
2. In the Link dialog box, select the page, site, or anchor to which this hotspot links.

DynaImages

A DynaImage is a dynamic multi-image element that is controlled through scripts or interactions.

Use a DynaImage element to create interactions between other elements on the page and each individual image of the DynaImage.

To create a DynaImage:

1. Drop a DynaImage SmartElement onto the layout.
2. Add multiple images as content to the DynaImage container.

   See "Adding multiple images to a content table" on page 162, "Assigning multiple images from a content table" on page 163, and "Assigning a multiple image file list" on page 163 for instructions on populating a SmartElement container with multiple media.
3. Customize the content order.

   See "Ordering multiple images" on page 164 for instructions on defining the order of multiple media.
4. Assign an interaction or script to the DynaImage.

   See "Multiple complex activations" on page 403 for details about creating multi-element interactions.
5. To assign an Interaction that displays a specific image of the DynaImage, Choose an Interaction, for example: Show the next image in DynaImage when DynaImage is clicked.

Image movements

Scrolling images

Drumbeat includes a SmartElement to create an image that scrolls across the SmartElement container horizontally or vertically, with customizable speed and direction.

Horizontal and vertical Scrolling Images are created with the Scrolling Image SmartElement. You can customize the scrolling rate, delay, and direction for a Scrolling Image element.

To create a scrolling image:

1 Drop the Scrolling Image SmartElement onto your layout.

2 Assign an image file as content to the SmartElement container.

See “Using an Image SmartElement” on page 155 for instructions on populating an Image SmartElement container.

To define scrolling attributes:

1 From the layout, double-click the Scrolling Image.

2 From the Direction pop-up menu on the Attributes tab, choose the scrolling direction:

- Down: Scrolls from top to bottom within the SmartElement container.
- Left: Scrolls from right to left within the SmartElement container.
- Right: Scrolls from left to right within the SmartElement container.
- Up: Scrolls from bottom to top within the SmartElement container.

3 In the Pixels/Step field, type or select the scrolling rate for the image.

4 In the Delay field, type or select the scrolling delay time, in seconds.

5 Make sure AutoStart is selected if you want the scrolling action to start immediately.

Slide shows

A Slide Show element in Drumbeat displays a set of images one at a time in the defined area of the Slide Show element. Images in the Slide Show are displayed in order, with random transition effects.

To create a Slide Show:

1. Drop the Slide Show SmartElement onto the layout.
2. Add multiple images as content to the Slide Show container.

   See "Adding multiple images to a content table" on page 162, "Assigning multiple images from a content table" on page 163, and "Assigning a multiple image file list" on page 163 for instructions on populating a SmartElement container with multiple media.
3. Customize the content order. The Slide Show images will appear in order from first to last.

   See "Ordering multiple images" on page 164 for instructions on defining the order of multiple-media.
4. Double-click the Slide Show element.
5. From the Attributes tab, type or select a delay, in seconds, to specify the time between image transitions.

Linking and activating images

Images can be assigned links to other pages, sites, or anchors. Image elements can be assigned a variety of activations and interactions.

Linking images

A link can be assigned from an image to another page, site, or anchor. Image links function like a link from any other type of element.

To assign a link to an Image:

1. Do one of the following:

- From the layout, right-click the Image element and choose Assign Link.
- Select the image and from the Attributes tab, click the Link Assign button.

2. In the Link dialog box, choose the page, site, or anchor link to assign to this image.

   For details about creating and working with Links, see "Creating links" on page 73.

Activating images

Activations and interactions are applied to images just as to any other element on your layout.

To assign an Activation to an Image:

1 Do one of the following:

- From the layout, right-click the image and choose Possible Activations.
- From the Asset Center or Interactions Center, click the Locate Assets button and select Show All Possible Ways to Activate the Selected SmartElement.

2 From the Interactions Center, select an activation.

To assign an Interaction between an Image and one or more additional element(s):

1 From the layout, select the elements.

2 Do one of the following:

- Right-click one of the selected elements and choose Possible Interactions.
- From the Asset Center or Interactions Center, click the Locate Assets button and select Show All Possible Interactions Between the Selected SmartElements.

3 From the Interactions Center, select an interaction.

For details about activations and interactions, see "Interactions Center" on page 397.

Background images

An image can be used as a background for any page or template in your Drumbeat site. Any background image that is smaller than the page layout area is automatically tiled. A background image that is larger than the page layout area is displayed at its original size.

Adding a background image

To add a background image to a page or template:

1 From the Templates tab, select the page or template.

2 Do one of the following:

- From the Asset Center or a content table, select an image and Shift-drag the image onto an unused area of the layout.
- From the Templates tab, select the page or template.

3 From the Attributes tab, select Bkgnd Image.

Note: If the Bkgnd Image attribute is not available, the attribute is inherited from the ruling Template. Click the Inherit button, and in the Inherit Attributes dialog box, deselect Bkgnd Image and Bkgnd Content.

4 Do one of the following:

- From the Bkgnd Content drop-down list, choose File to assign a background image from a file in the Asset Center.
- Choose Static Content to assign a background image from a content table cell.

5 Click the Assign Bkgnd Content button, and from the Media Properties dialog box, select an image file.

Specifying a background color

The default background color of the master template and all subordinate pages is white. To change the color of a template or page, specify a background. If a background image is specified, the background color is used any time the background image cannot be displayed.

To add a background color:

1 From the Templates tab, select the page or template.

2 From the Attributes tab, select Bkgnd Color.

3 From the Color palette, choose a standard or custom color.

Note: If the Bkgnd attribute is not available, the attribute is inherited from the ruling template. Click the Inherit button, and in the Inherit Attributes dialog box, deselect Bkgnd.

Background inheritance

A background image or a background color on a template is, by default, inherited by subordinate pages.

To remove an inherited background color from a page or template:

1 In the Templates tab, select the page or template.

2 In the Attributes tab, click the Inherit button.

3 In the Inherit Attributes dialog box, deselect Bkgnd.

To remove an inherited background image from a page or template:

1 In the Templates tab, select the page or template.

2 In the Attributes tab, click the Inherit button.

3 Do one of the following:

- In the Inherit Attributes dialog box, deselect BkgndImage if the page is not to have a background image.
- Deselect BkgndContent if the page is to have a different background image than the template.

CHAPTER 8
Publishing

Drumbeat offers a number of options for publishing your site. You can publish to a local server or to a remote server, use any regular web server, or publish to a database server with ASP support. Drumbeat can publish native Drumbeat pages and external pages created with other applications.

The Drumbeat Publish feature handles all the details of publishing your web site to a server, including transferring images and other linked files. (One exception is files referenced from within databases, over which Drumbeat has no control—these must be transferred manually.)

Drumbeat can help you determine publish settings with the Publishing wizard (Publish > Publish Settings Wizard and available from the New Site wizard). This chapter explains the way Drumbeat handles publishing and how to choose the publishing options for your site.

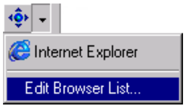
Previewing pages in a web browser

While you are developing and testing your pages, you can preview pages directly with any browser on your system. If you have Microsoft Internet Explorer 4.0 installed on your computer, you can preview directly in Drumbeat by clicking the Preview tab at the bottom of the layout. Previewing a page also lets you test links and interactive elements. At installation, Drumbeat automatically detects installed browsers and adds them to your browser list (if you select this option). You can add new browsers yourself or edit the list and quickly select the browser you want from the list to preview your site. The default browser is the one that will be used when you click the Browse button on the toolbar.

Previewing a page initiates the publishing process. Drumbeat publishes all pages, selected pages, or selected sections according to the selection in the Options tab of the Publish Settings dialog box. See “Setting publishing scope” on page 184.

To set up the browser list:

1 Choose Publish > Browse With > Edit Browser List or, from the Browse button pop-up menu, choose Edit Browser List.



2 Click Add to add a new browser to the list.

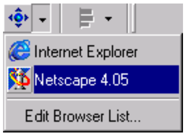
3 In the Browser dialog box, enter the name of the browser and the path to its application file, or click Browse to locate it.

4 If you want this browser to be your default browser, check Use this browser when the Browse Page button is selected. Click OK.

To view the published site in a browser:

Do one of the following:

- Click the Browse button to preview in the default browser.



- Select a browser from the Browse pop-up menu.
- Choose Publish > Browse With and select a browser from the list.

When you choose a browser to use, that browser becomes the default selection on the Browse button, until you choose a different browser. The appearance of the button changes according to your most recent selection.

Changing browser and server-application support

When you create a new site in Drumbeat, you are required to choose the browser and server-application support for the site. The server-application support you select for your site affects some other publishing options. You can change both browser support and server-application support for a site at any time. See also "Browser support" on page 17

To change browser and server-application support:

1 Select File > Preferences > Site.

2 Choose a category of browsers from the Browser Support option.

3 Choose the desired server-application support option:

- None publishes sites for a standard web server with no database extension.
- ASP: publishes Active Server Pages. Your server must also be ASP-enabled and you must have Personal Web Server installed on your system to preview files on your own machine. (You do not need Personal Web Server if you are publishing directly to the server.)

Note: If you choose Server-side Redirection for SmartPages on a non-ASP site, you will not be able to view SmartPages locally. To view SmartPages with server-side redirection on a non-ASP site, you must configure your CGI publish settings correctly, including the correct path to Perl on the server. You must publish to the CGI server and view your pages there.

Setting Drumbeat preferences

Several Drumbeat preferences affect the publishing process. These preferences include the default location for new Drumbeat site files, automatic reminders to re-publish when global changes are made, the location of an external HTML editor, and how images are displayed in a cell of a content table.

If you are sharing Drumbeat files with other developers who are working on the same sites, consider having everyone change their default directory and media directories to a shared network drive. If everyone accesses site resources in a centralized location, it will be unnecessary for different users to update media paths to reflect locations on their local computer.

Note: If you edit a page's source code outside of Drumbeat, turn off publishing of the page you have changed to prevent the code being overwritten when the page is republished. Turn off publishing for a page in the Publishing tab of the Web Page Properties dialog box. Drumbeat does not include an HTML source editor.

To set Drumbeat preferences:

1. Choose File > Preferences > Drumbeat.
2. Click the General tab and specify options as desired, including the default directory for new sites. If you installed Drumbeat to the default location, this will be C:\Program Files\Drumbeat 2000\MySites.
3. Click the Content tab and specify the maximum number of rows displayed as the result of a SQL query. Restricting the display to a limited number will make working with large external databases at design time faster and easier.
4. Enter the path to your HTML editor in the External HTML Editor option or click Browse to locate it.
5. To display an image when a cell in a content table contains one, select Display Thumbnails. Otherwise, the pathname of the image appears in the cell.
6. If you display thumbnails, select Stretch to Fit to resize the image to the cell. Otherwise the image remains at its native size (in which case you may see only a part of it in the table cell).

Publish settings

When you open a new site in Drumbeat, you choose the basic Site Preferences that control the way Drumbeat will publish your site. These include the following:

- Browser support that specifies the target browser for the site (which can be extended with SmartPages).
- Server support that specifies using a no database server (ASP) or not.
- Client-side or server-side scripting for browser redirection with SmartPages.
- For ASP support, choose JavaScript or VBScript for the server scripting language.

Note: Choosing client-side or server-side redirection creates the logic for redirection for browser-specific pages on either the server or the browser. The benefit of server-side redirection is that you don't have to have a return trip to the server to display the browser-specific page. ASP sites automatically use server-side redirection. If you choose server-side redirection on a non-ASP site, you need to configure CGI settings for your server and you will only be able to view your site on the server.

All Publish Settings chosen at the time you open a new site can be changed later, and you may choose options at any time before or after you publish your site.

Staging and production settings

You can keep two complete, independent sets of publishing options for staging and production servers. Staging settings can be used to publish a site to an internal server for review, approval, and testing of all scripts before posting to a public server.

Options for staging and production servers are identical. Only one set is active at any one time. When you first open a new site file, staging settings are selected by default.

To select staging or production settings:

Choose Publish > Staging or Publish > Production (or Staging). The checkmark next to the selection indicates which Publish Setting is active.

To view Publish Settings:

On the main menu, select Publish > Publish Settings.

The title bar of the Publish dialog box indicates whether the settings are applicable to staging or production server.

Setting the local publishing directories

The local publishing destination for a non-ASP site is, by default, set to the Publish directory in the Drumbeat application directory. If you installed Drumbeat to the default location, this will be C:\Program Files\Drumbeat 2000\Publish. You may use this directory for local viewing and previewing of your site, or you may choose another directory.

The default publishing directory for an ASP site is C:\inetpub\wwwroot. If you are using ASP server support, you will need to create a web directory in your Inetpub\wwwroot directory for local viewing and add the name of the directory to the path, or set up a virtual directory to point to the directory of your choice. See “Publishing Active Server Pages” on page 189.

For choosing Publish Settings for publishing to a remote server, see “Publishing to a remote server” on page 188.

To set the local publishing directory:

1 Choose Publish > Publish Settings.

2 Click the General tab, enter the path to the directory in the box for Local Destination Directory, or browse to locate and select the directory.

   If the directory does not exist, Drumbeat will offer to create it for you when you publish the site.

3 For local ASP Server, specify the name of the ASP server. If you are publishing to your own system with Microsoft Personal Web Server, the server name can be left as “My Computer.” Or enter the name of your computer in the network. If you are publishing to a remote server, check with your system administrator for the server name if you do not know it.

4 For Port Number the default port number 80 should be appropriate in most circumstances. Change this only if necessary.

Specifying media directories

You can specify different publishing directories for files of different types, such as images, audio, video, scripts, and Java applets.

To Specify media directories:

1. With the site open in Drumbeat, change Publish > Publish Settings.
2. If you want all of your files published to a single destination directory, select Flat Directory structure. When this option is selected, the media directory options are disabled. All your files will publish to the root directory.
3. In the Media tab, choose a subdirectory name for each type of file from the pop-up menus or enter a custom directory.

   Note: JavaScript is either embedded within the HTML file or saved in separate JS files. Separate JS files may not be allowed by some firewall setups. You can choose to publish JavaScript with the file instead of in separate files by selecting this option in the Publish Settings.

4. If you do not want Drumbeat to copy media files to the publish directories deselect Copy Media. This is useful when your media files are already on the server. When Copy Media is deselected, the media directory options are disabled.
5. If you want Drumbeat to compress JPEG files during the publishing process, choose a compression method from the Low Resolution Algorithm pop-up menu. The methods available do not produce high-quality results, as would an image-editing application such as Macromedia Fireworks. This feature is useful, however, for quickly converting files created for other uses into Web-compatible format. The following choices are available:

- None does not apply any compression.
- Bayer Color reduces a 24-bit or 8-bit image to a 4-bit image.
- 8-Bit Gray converts a 24-, 8-, or 4-bit image to an 8-bit gray scale image.
- 4-Bit Diffusion reduces a 24-bit or 8-bit image to a 4-bit image.
- Median Cut reduces a 24-bit image to an 8-bit image by separating the image into 256 blocks and averaging each block into a color to produce a 256 color palette.

Setting publishing scope

The Options tab in the Publish Settings dialog box allows you to select which pages are published when you use the Publish command or the preview feature, error notification, and whether or not to publish client-side script.

The setting is overridden when you make another choice from the Publish button pop-up menu.

To set publishing scope

1. Choose Publish > Publish Settings.
2. Select to publish all pages in the site, publish only the page or pages selected in the Site Manager or Template Manager, publish selected sections (a parent page and all child pages in the site tree) in the Site Manager, or publish only those pages that have changed since the last publish operation (those flagged as "dirty" in the site tree).
3. Select Notify Errors on Occurrence if you want, Drumbeat to interrupt the publish process and display an error message for each occurrence of an error. If this option is not selected, any errors encountered during the publish process are placed in an error log. When publishing is completed, Drumbeat asks if you want to view the error message log.
4. If you want to disable all JavaScript on your pages, you can deselect Use JavaScript. This prevents JavaScript errors from occurring when pages are opened in browsers that do not support client-side JavaScript. (Another option is to use SmartPages to create a Generic browser page where necessary.)
5. When this option is not selected, all JavaScript will be published to the same file as the HTML source. If your server does not support the JavaScript MIME type or if you are working with a firewall that restricts JavaScript files, you may want to deselect Publish Script to Include Files. Deselecting this option, however, can create other problems with server-side and ASP operation.

Setting CGI scripting options

CGI scripting is used in Drumbeat for server-side redirection for SmartPages on non-ASP sites. If you do not enable CGI scripting, client-side JavaScript will be used for the redirection logic. Client-side JavaScript, however, is not supported by Generic browsers.

In addition, a few CGI SmartElements ship with Drumbeat, including CGI Form Processor, CGI Random Image display, and CGI Random Text display. The scripts are invoked by server side include statements from within the generated HTML page. You may also write your own SmartElements driven by CGI.

To set CGI scripting options:

1. Choose Public > Settings and click the CGI tab.
2. For Destination Directory enter the full path to the root publish directory.
3. Specify the relative path from the root directory to the cgi-bin directory (typically /cgi-bin).
4. For Compiler Directory enter the path to the directory where Perl or other scripting language resides on the server. The UNIX default directory of /usr/local/bin/perl is the default entry. See “Setting the Perl interpreter with IIS” on page 186.
5. Specify the extension used in the scripting language (typically pl or cgi). The scripts generated from the templates for CGI SmartElements use the extension you specify.
6. Enter the page extension used on pages that reference the cgi scripts (typically .html or .htm). Some ISP servers use special page extensions for SSI files, such as .stm or .shtml. With an IIS server, SSI pages must have the .stm extension.

Note: Drumbeat will not copy custom CGI scripts you have created yourself. However, you can create URL links to CGI files through the Link dialog box. You can also insert CGI files as external pages in the site tree, if desired, but you must handle transferring files to the correct directory on the server yourself.

For more information on configuring CGI publish settings, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

Setting the Perl interpreter with IIS

This technique is used only with an IIS server. For a UNIX server the Perl interpreter location is provided in the CGI tab of the Publish Settings.

On a Windows server, the Perl Script extension (pl, cgi) is mapped to the interpreter in the Windows Registry.

To set the Perl interpreter in the Registry:

1 Start Regedit. Then open HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\W3SVC \Parameters\ScriptMap

2 Choose Edit > Add Value.

3 Set registry options:

- The Value name is .pl (or your script extension)
- The Data type is REG_SZ
- The string value is <full path to perl.exe>\perl.exe %s %s

4 Restart the web server.

Note: After installing the Win32 Perl interpreter, if you ran the Install.bat program a ".pl" value may already exist in the Registry mapped to the Perl.dll. One option is to map the Perl.dll with another extension like .plx or delete the dll mapping if you will not be using it.

For a working example of setting up the Perl interpreter for IIS, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

For more information on configuring and testing a Perl script with IIS see the Microsoft Knowledge Base article Q150629 on the Microsoft Web site.

Remote publishing

If you are publishing to a remote location, Drumbeat will do all the work of transferring files to a remote web server through its built-in FTP client. All the settings you need for FTP access to your site can be set in the Publish Settings dialog box.

Setting remote publishing options

When publishing to a remote directory, you will most likely want to set up your staging settings for development and local viewing of your site, and use production settings for the remote location. These instructions assume that this is the case.

When you publish to a remote directory, you must specify both an intermediate destination directory and a remote destination directory. the intermediate destination directory is the local directory in which your files reside and from which they will be transferred using FTP to the remote server. The remote destination directory is the FTP root directory on the server to which you want to copy your files.

To set Remote Publishing options:

1 Choose Publish > Production and then choose Publish > Publish Settings.

2 Click the General tab, select the Remote radio option.

3 Enter an intermediate directory path for staging your site locally in preparation for transfer.

4 Enter a remote server name and Port number, if necessary. (By default, the port number is set to 80, which is the default port number for HTTP requests. Normally this does not need to be changed.)

5 Enter a remote destination directory for publishing the site to the remote server. This is the path from your root publishing directory to which you make the initial connection.

The FTP root and the web root directories on the server must be the same, or the web root must be a subdirectory of the FTP root.

6 Enter your username and password if you want these settings to be saved. If you do not enter a username and password, you will be asked for them when you publish your site.

7 Click the Media tab and select the subdirectories to which you want to publish auxiliary files. Select any other settings relevant to your remote site in the Options and CGI tabs, as described in Publishing Options and CGI Options. If you are publishing to a server with ASP support, you must also provide the information on the ASP tab. See “Publishing Active Server Pages” on page 189.

Publishing to a remote server

The dual publish settings are particularly handy when you are publishing to a remote server, as you will most likely want to transfer your files to the server only when the site is more or less complete. It is recommended that you set up your staging settings to view your site locally. When you are ready to publish your site, switch to your production settings.

When you publish pages remotely, Drumbeat starts the Microsoft FTP client, then uses the remote server name and remote destination directory that you provided through the Publish Settings dialog box to access to the server's file directory system. All the required files are then transferred to the specified destination directory.

In order to publish your site to a remote FTP server, make sure you have write permissions to the directories on which you are publishing on the server. Otherwise, Drumbeat will not be able to create the directories it needs. If you do not know how to set permissions on your FTP server, see your systems administrator.

To publish to a remote server:

1. Select Publish > Production to switch to the Production publish settings, which should contain your remote access information (see “Setting remote publishing options” on page 187).
2. Click the Publish button, or choose the Publish option you want from the pop-up menu on the Publish button, or choose Publish > All Pages (or the scope you wish).
3. If you have not entered your username and password in the Publish Settings dialog box, the Remote Account Information dialog box appears. Enter your account name (username) and password. Select Save Account Name and Password if you want these settings to be saved in the future.

Note: Once you have transferred all media files to the remote server, you can turn off media file transfer in the future and expedite publishing changes by deselecting Copy Media in the Media tab of the Publish Settings dialog box.

Publishing Active Server Pages

To publish Active Server Pages you need to have Microsoft Internet Information Server (IIS) or Personal Web Server (PWS).

ASP is a server-side programming technology that allows you to develop and serve up dynamic, personalized content. The biggest advantage of ASP is that it is browser independent. All the code resides on the server, so you do not have to worry about a browser's ability to run your scripts.

Personal Web Server

To view the Active Server Pages you develop in Drumbeat locally as you develop them, you must install Personal Web Server. Personal Web Server is a modified version of IIS, designed for use on a single computer or small network, and runs in Windows 95/98 and Window NT workstation. Personal Web Server makes it easy to preview your site in development, without posting to a server, or to build web applications for IIS from your own desktop computer. If your development computer uses NT Server, then you will need to use IIS as your primary testing server platform.

Personal Web Server was not designed to develop high volume web sites, so is not as feature complete as IIS, missing such features as Microsoft Site Server Express, Index Server, and Certificate Server. However, Personal Web Server is a great platform for using your own computer to develop Active Server Pages and check links, forms, scripts, and applications before hosting it on the company server, or at an Internet service provider.

If you are publishing directly to an IIS server and do not need to preview sites locally, you do not need Personal Web Server.

To install Personal Web Server for Windows 98:

Personal Web Server is included in Windows 98. Install it from the Windows 98 CD.

To get Personal Web Server for Windows NT or Windows 95:

Go to http://www.microsoft.com/windows/ie/pws/default.htm

Personal Web Server is part of the Windows NT 4.0 Option Pack

Note: The version of Personal Web Server that comes with Microsoft Front Page is not the correct version for working with ASP pages in Drumbeat. You will need to install the full version of PWS, which is included in Windows NT 4.0 Option Pack. Before installing PWS on Windows 95, you must also have the Windows 95 upgrade patch.

Creating a Web directory

When you install Internet Information Server or Personal Web Server, a default directory is created for your web documents at inetpub\wwwroot. The simplest way to publish your ASP pages is to use this default directory and create folders for your web sites under this root directory.

To create a web directory in the default location:

1 In Windows Explorer, locate the directory Inetpub\wwwroot.

2 Click on the wwwrootdirectory and select File > New Folder.

3 Name the new folder appropriately for your site.

For example, if you decide to create a directory named CorpSite in the wwwroot directory, you can publish your files to this location and the local path to your site will be:

C:\Inetpub\wwwroot\CorpSite

Next, you need to tell Drumbeat where the site is located by entering the correct information in the Publish Settings dialog box.

To set up an ASP publishing directory in Drumbeat:

1 Choose Publish > Publish Settings.

2 Click the General tab. For Local Destination Directory, add the name of the folder you created in the wwwroot directory to the default path C:\inetpub\wwwroot\ (for example, C:\Inetpub\wwwroot\CorpSite) for publishing locally. If you are publishing directly to the server on a network, for C substitute the drive on which the server is located and add any additional path information, if necessary.

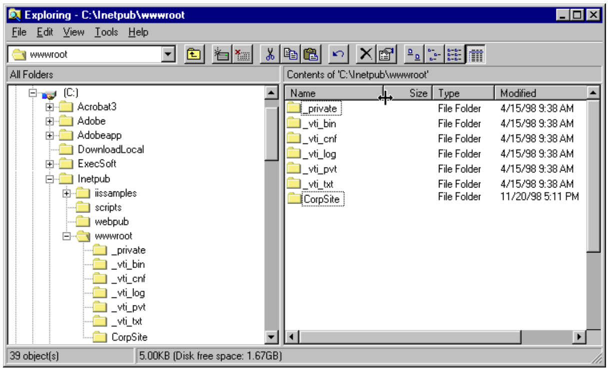
3 Click the ASP tab. For Alias, enter just the name of the folder you created in the wwwroot directory (for example, CorpSite) and select a home page for the site. The alias is the name by which your site is identified with the server.

4 Select Create Virtual Directory to have Drumbeat create the virtual directory for you only when you publish to IIS 4.0 from an NT computer. Otherwise, you need to create the virtual directory yourself (see “Creating a virtual directory” on page 191).

5 If the computer that you are using as the web server is running more than one web server, and you want to publish to a specific web server, enter the number of the server.

Note: The page you select as your Default Page is the page that will always be accessed first when you view your site in a browser. You can change this as often as you like as you work with your site to avoid having to “click through” to the page you want to preview.

Using the example values from the previous steps, the URL to view the site locally would be http://127.0.0.1/CorpSite or http://MyComputer/Corpsite or http://localhost/Corpsite. For the “127.0.0.1” or “localhost” or “MyComputer” you can substitute the name of your computer on the network.



Name	Size	Type	Modified
_private		File Folder	4/15/98 9:38 AM
_vti_bin		File Folder	4/15/98 9:38 AM
_vti_cnf		File Folder	4/15/98 9:38 AM
_vti_log		File Folder	4/15/98 9:38 AM
_vti_pvt		File Folder	4/15/98 9:38 AM
_vti_txt		File Folder	4/15/98 9:38 AM
CorpSite		File Folder	11/20/98 5:11 PM

39 object(s)

5.00KB (Disk free space: 1.67GB)

Creating a virtual directory

On a network it isn’t always convenient or possible to set up a publishing directory in the web server’s root publishing directory. You may want to set up the server to access your web site at a different location on your computer or a shared location on an intranet. To do this, you need to create a virtual directory. When you create a virtual directory, you create an alias for your site and provide the path to the files that the server will access under that name.

If you were creating a virtual directory for a site named CorpSite, the actual path to your site might be C:\Projects\Web\. Or, if it’s elsewhere on the network, perhaps N:\XYZCorp\Web\

To access the files where they actually exist, you can give your site the alias CorpSite and tell the server where to find it. Then the virtual path to your site, and the one you will use to access it by URL will be http://servername/CorpSite.

This is the same as if you had set up an actual directory called CorpSite on your wwwroot directory. In this case, however, it’s a virtual directory because the files actually reside elsewhere.

Drumbeat can automatically create the virtual directory for you only when you are publishing to IIS 4.0 from an NT computer.

To set up a virtual directory in Internet Information Server:

1. Open the Microsoft Management Console by choosing Programs > Windows NT Option Pack > Internet Service Manager.
2. Right-click a folder and choose New > Virtual Directory.
3. Type a name (or alias) for your site and enter the full path (from the server’s point of view) or browse to locate the actual directory.

To set up a virtual directory in Personal Web Server:

1. In Personal Web Server, open the Personal Web Manager by choosing Programs > Microsoft Personal Web Server > Personal Web Manager
2. Select Advanced, and then Add.
3. Type a name (or alias) for your site and enter the full path (from the server’s point of view) or browse to locate the actual directory.

   Next, you need to tell Drumbeat where the site files are located by entering the correct information in the ASP Publish Settings.
4. In Drumbeat, choose Publish > Publish Settings.
5. Click the General tab. For Local Destination Directory, enter the full path to the directory from the server’s point of view (for example, N:\XYZCorp\Web\).
6. Click the ASP tab. For Alias, enter the alias you gave the site in Personal Web Manager and select a home page for the site.

To have Drumbeat create a virtual directory:

1. Select Create Virtual Directory in the ASP tab of the Publish Settings dialog box.
2. Provide an alias as the name for your virtual directory.
3. In the virtual directory path text box, enter the path to the virtual directory from the server's point-of-view. (This is the actual location of the files.)
4. Select the default page for the site. This is the home page by default, but you can change to any page that you would like to designate as a starting page.

   Note: If the Create Virtual Directory check box is checked at publish time, a COM server instance is invoked through DCOM. You must have DCOM security rights or you will receive an error message and the Virtual Directory will not be created.

Developing channels for sites that support Microsoft Internet Explorer 4.0

Microsoft Internet Explorer 4.0 supports a feature called channels that distributes news and information to subscribers on a regular schedule. The browser keeps track of the channels and it automatically accesses the appropriate Web site at scheduled intervals to retrieve new content. The content is stored in the browser cache so that when the user views it, it appears instantly, without further network access.

Channels are defined using a special file format known as CDF, or Channel Definition Format. A CDF file contains the name of the channel and the URL; a description of the channel; graphics associated with the channel, such as logos and icons; and schedule information that tells the browser when to visit the site and what pages to retrieve.

Drumbeat automatically builds channel definitions for your site and creates a CDF file whenever you publish it. No knowledge of CDF is required.

To allow a visitor to your site to subscribe to your channel, you must provide a link on your page for the user to subscribe. When a user clicks the subscription link the CDF file is downloaded and stored on the local computer. The Channel SmartElement in Drumbeat keeps track of the link, so you do not need to manually specify the location of the CDF file.

Setting channel preferences

Channel preferences you can set include the following:

- The top-level page for the channel, the name for the channel, and a channel bar tooltip.
- Channel icons and images to be used.
- Dates and times when the channel content will automatically be updated.
- Logging and crawling preferences. Logging allows you to obtain information about which pages of the channel users are visiting, even though they are not connected to your site when they view them. (The logs are stored on the user's local hard drive, then uploaded to your site at the next channel update). Crawling allows you to have the browser automatically retrieve pages which are linked to pages in your channel during a channel update, even if they are not included in the channel itself, making browsing faster.

To set channel preferences:

1 Choose File > Preferences > Channel.

2 Click the General tab and set the following options:

- Page is the primary entry point to the content in your channel. If your site is your channel, the top-level page in the channel should be your home page. If you wish to provide special content (or content in a special format) for the channel, create a separate hierarchy of pages in your site to store it. In this case, your top-level page will be different from your home page. Any page that is not a SmartPage can be chosen as the top-level page.
- Title is the title of your channel as it will appear in the subscription list.
- Channel Bar Tool Tip displays this text as a tool tip when the user points to the channel in the Channel Bar.

3 Click the Graphics tab to select the various images that will be used to represent your site in Internet Explorer:

- Icon is a 16x16 GIF image that represents your channel in the Channels sub-menu.
- Image is an 80x32 GIF image that represents your channel in the Active Desktop channel bar.
- Wide is a 194x32 GIF image that represents your channel in the Channel pane.

4 Click the Schedule tab to schedule updates:

- Start Date and End Date are the dates on which the schedule begins and expires, respectively.
- Timezone is relative to which the times in this schedule are given. Usually this should be the same time zone where your web server is located. This information ensures that browsers around the world follow the schedule at the same absolute time.
- Interval Time sets the time interval between automatic updates to the channel.
- Earliest Time and Latest Time specify times during the day the browser should request a channel update.

5 Click the Advanced tab to set logging and crawling options.

- Log specifies each action to be logged. Currently, Internet Explorer supports only logging of page views.
- HREF is the URL that contains the log files for the channel. Typically this is a directory, such as http://www.drumbeat.com/ logging/.
- Method specifies the means by which the logs will be uploaded. (Currently, Internet Explorer supports only the POST method, although PUT is also an option.)
- Scope determines whether offline, online, or all page views are logged.
- Purge sets the amount of time logs are kept on the local hard disk before they are deleted.
- PreCache activates crawling. When crawling is activated, pages that are linked to the top-level page are automatically retrieved up to the depth you specify.
- Levels determines the depth of crawling. If crawling is set to 1, any pages linked to channel pages are retrieved and stored in the browser cache. If set to 2, any pages linked to this first group of pages are also retrieved, and so on.

Adding and removing channel pages

Each channel can contain as many additional sub-pages as you need to allow users easy access to the pages of your channel. Keep in mind that you can use regular web links to provide access to other pages in your channel, so it is not necessary to make every page in your site a sub-page of your channel. Make pages sub-channels only when they contain important updated information the user might want to access directly. Any page in your site which is not a SmartPage can be added to the channel.

When you move a page that is being published in a channel, its channel information moves with it. You can cut, copy, and paste pages which are part of a channel like any other. When you copy a channel page, the copy is also included in the channel.

To add a page to a channel:

1 In Site Manager, right-click a page in your site structure tree and choose Properties.

2 In the Web Page Properties dialog box, click the Channels tab.

3 Select Include as Channel Item and define the other attributes of the channel page, as follows.

- Item Title specifies the name for this page that will be displayed in the subscription list.
- Channel Bar Tool Tip appears in a tooltip when the user points at the page in the Internet Explorer Channel Bar.
- Channel Item Icon Graphic is the 16x16 GIF image that represents the page in the Channels submenu of the Internet Explorer Favorites menu and the Start menu.
- PreCache activates crawling from this page. Pages that are linked to this page are automatically retrieved during a channel update, up to the depth you specify.
- Levels determines the depth of crawling. If crawling is set to 1, any pages linked to channel pages are retrieved and stored in the browser cache. If set to 2, any pages linked to this first group of pages are also retrieved, and so on.

To remove a page from a channel (without removing it from the site):

1 In Site Manager, right-click a page in your site structure tree and choose Properties.

2 In the Web Page Properties dialog box, click the Channels tab.

3 Deselect Include as Channel Item.

Channel Subscription

Drumbeat creates a Channel Definition File for your site, whether or not you decide to use it. To allow a user to subscribe to a channel, you only need to include a link to the CDF file from one or more of your web pages. When the browser retrieves the CDF file, it automatically subscribes to your channel.

The Channel Button SmartElement in the SmartElement Library is a standard component you can use for this purpose.

To create a link to the CDF file so that users can subscribe to the channel:

1. Drag the Channel Button SmartElement onto the layout of the page.
2. Drag an appropriate image (a subscription icon) onto the Channel Button SmartElement.

   The Channel Button automatically creates a link to the CDF file. (Drumbeat keeps track of the name of the CDF file it generates and maintains this link for you.)

Publishing with Channels

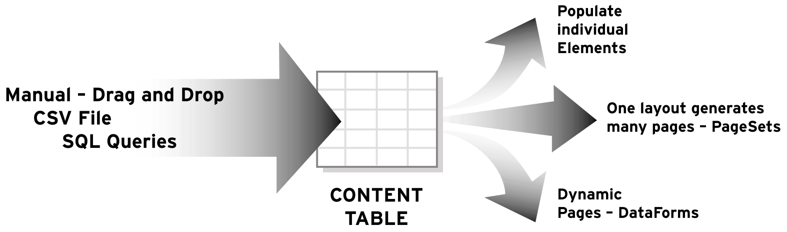
Drumbeat automatically publishes your channel when it publishes your site if you use the Publish All or Publish Changed Pages command after changing a channel setting in the Channel Preferences or Web Page Properties dialog box.

Drumbeat does not update the CDF file when you use Publish Selected Pages or Publish Selected Sections.

CHAPTER 9
Content Tables

A content table is a gridlike representation of content items. Content tables give you a convenient way to organize the media files you use in your site, such as text, images, audio, and video (or URLs to these files). There are three ways to add content items to content tables: by manually inserting or pointing to media, by importing data from an external file in the CSV (comma-separated value) format, and by querying a database. These three types of content tables each have specific uses.

- Manual content tables are useful for organizing assets you want to use in several places and for keeping frequently used assets handy.
- CSV content tables derive their content from CSV files exported from a spreadsheet, a database, or other applications. The data may be used in any way you like in your pages, but they are especially useful for generating PageSets. For more information about PageSets and CSV files, see “Database-Driven PageSets” on page 271.
- ODBC data source content tables provide a visual representation of database content derived from queries and allow you to create ASP applications (dynamic database-driven web applications). For information about ODBC content tables and ASP applications, see “Accessing Databases and Creating Queries” on page 289.


Manual – Drag and Drop
CSV File
SQL Queries
CONTENT TABLE
Populate individual Elements
One layout generates many pages – PageSets
Dynamic Pages – DataForms

Use content tables to bring data into the working environment.

Viewing content tables

Content tables are displayed in the Content Center, which shares (toggles) the Attic space above the layout with the Interactions Center and the Script Center.

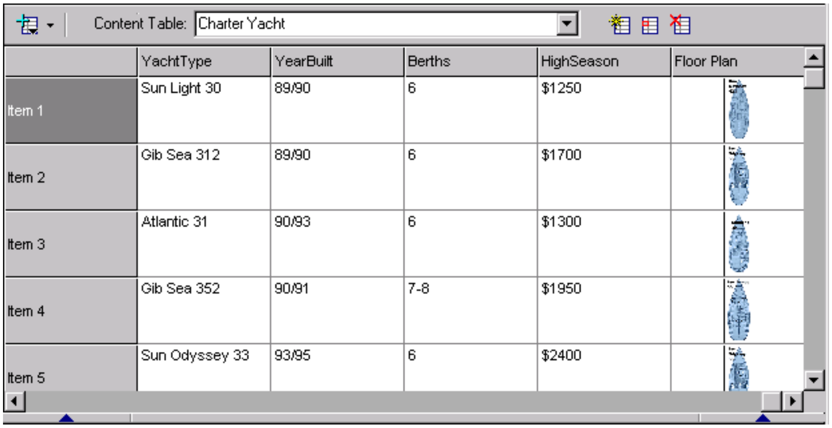
To view the content tables available in your site:

1 From the main menu, choose View > Content Center or from the toolbar, click the Content Center button or press Ctrl+D to open the Content Center.

2 Choose a content table to view from the Content Table pop-up menu.

Understanding the Content Center interface

The Content Center displays all the content tables available to your site. Using the Content Center toolbar, you can create, edit, or delete content tables.



Content Table: Charter Yacht

	YachtType	YearBuilt	Berths	HighSeason	Floor Plan
Item 1	Sun Light 30	89/90	6	$1250	
Item 2	Gib Sea 312	89/90	6	$1700	
Item 3	Atlantic 31	90/93	6	$1300	
Item 4	Gib Sea 352	90/91	7-8	$1950	
Item 5	Sun Odyssey 33	93/95	6	$2400	

A content table can contain several different types of data.

Item	Description
Content Table pop-up menu	This button invokes a pop-up menu where you can specify if you want to add a new column or row.
	This pop-up menu contains the names of all the content tables in your site.
	This button invokes the New Content Table dialog box, where you specify what kind of content table you want to create and
	This button invokes the Content Table dialog box where you can rename the content table, modify the number of rows and/or columns, or change query associations with ODBC tables.
	This button deletes the current content table.

To move between cells in a content table:

Do one of the following:

- Use the Tab key to move from cell to cell. The Tab key in the Content Center moves focus to the next right cell (Shift+Tab to the left cell).
- Use the arrow keys to navigate between content table cells.

Manual content tables

Manual content tables can be populated by inserting data directly by dragging and dropping or by pointing to media files. Manual content tables are useful in organizing and updating sitewide content items. Organizing all the text, image, audio, and video files you use throughout your site in a content table allows fast and easy updates. When you change the content item in the content table, all SmartElements that reference that cell's content also reflect the update.

To create a manual content table:

1. Open the Content Center, and click the New Content Table button.
2. In the New Content Table dialog box, enter a name and number for rows and columns.

Click the Edit Content Table button to edit the table, or to change the name of the content table or the number of rows and columns.

Some tips on organizing content items in manual content tables.

- Give each row a theme that represents the media.
- Give each column a function, such as buttons, text, navigation.

Content tables from external files

Content tables can be created and populated by importing an external file in the CSV (comma-separated values) format. CSV files can be exported from most word processor, spreadsheet, and database applications. Content tables from CSV files are especially useful for generating PageSets, which are sets of pages with a common layout but unique data. For more information, see "Database-Driven PageSets" on page 271.

Content tables built by importing an external file are treated as static content and can be edited and added to within Drumbeat once they have been imported. These changes do not affect the original CSV file. You can also create a dynamic ODBC content table from a CSV file by setting up a Data Source Name for the CSV file. In this case the content will not be editable within Drumbeat but will always reflect the current content of the external CSV file.

To create a content table from a CSV file

1 Open the Content Center, and click the New Content Table button.

2 In the New Content Table dialog box, select Populate From External File. Then, browse to a CSV file.

3 To automatically name the columns of the content table using the first row in the CSV file, select Use First Line for Column Names.

4 Click OK, and the CSV content table will be generated.

Adding and editing content items in CSV tables

You can add content items to content tables generated from external CSV files. Any changes made or items added to the table will not be reflected in the external CSV file after it has been imported to Drumbeat.

Some common examples of adding content items to CSV tables are:

- Adding an image to correspond to each record or row in the table.
- Adding a URL or file path to audio and video files. These can be used to set up linking fields in PageSets or AutoTables. See "Using AutoTables to organize data" on page 107 or "Linking to PageSets" on page 286.

Using ODBC content tables

ODBC content tables are used when building Active Server Pages applications. They are generated from SQL queries built to access your database. ODBC content tables are different from those you build manually or generate by importing data from CSV files. They cannot be edited within Drumbeat's Content Center, but you may edit the underlying SQL query to change the contents retrieved from the database.

To edit the query that the ODBC table is based on:

Click the Edit Content Table button on the Content Center toolbar. In the ODBC Driven Content Table dialog box, click the Select button next to the ODBC Query Name box. This opens the Query Manager dialog box, where you can edit the query manually or with the Query wizard.

An ODBC content table provides you with a visual representation of your data, which you can then use in your pages to build dynamic ASP applications. For more information on using ODBC content tables and building Active Server Pages applications, see "Accessing Databases and Creating Queries" on page 289.

Using text, image, audio and video files in content tables

There are several ways to add media to table cells when building manual tables or customizing a content table built from an external file. You can drag and drop media from the Asset Center, use the a cell's right-click menu and browse to files, or type URLs and file paths.

Adding media files

To add a file to a content table cell from the Asset Center:

1. Use media queries in Asset Center to locate files by type.
2. Drag and drop the files into content table cells.

When you drag and drop files into content table cells, the column properties will automatically be set to the type of media file dropped into the cell. If column properties have previously been set and the type of file does not match the column properties, you will be not be able to insert the file. See "Setting row and column properties" on page 206.

To set the column properties for media files:

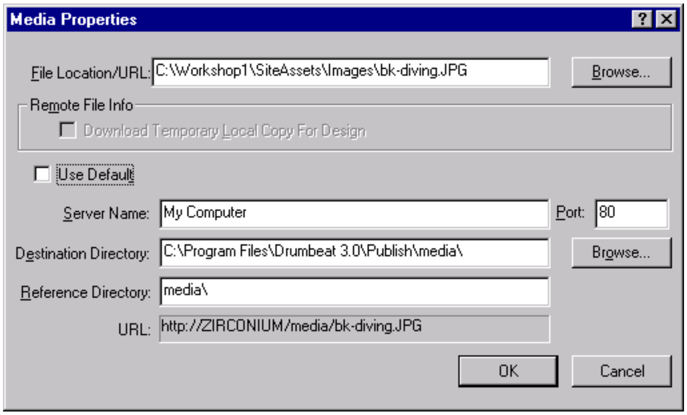
1. Right-click a content table header and choose Properties. In the Column Properties dialog box, set the Type to Media and select the media type (image, audio, video, document, etc.).
2. If the column is enabled for multiple content, you will be able to add multiple files to the cell. To stack multiple media files in cells, select the Allow Multiple Content option in the Column Properties dialog box.

If you want to insert a media file into a content table cell that is not part of your local assets, you can insert it by referencing it by path or URL.

To insert a file in a content table cell by path or URL:

1. Double-click a cell in the column or right-click and choose Media Properties.
2. In the Media Properties dialog box, enter the path to the file in the File Location/URL box or browse to the file you want to insert.

The Media Properties dialog box also contains several options for publishing the referenced file, as shown in the following two figures.



Media Properties

File Location/URL: C:\Workshop1\SiteAssets\Images\bk-diving.JPG

Remote File Info

□ Download Temporary Local Copy For Design

□ Use Default

Server Name: My Computer

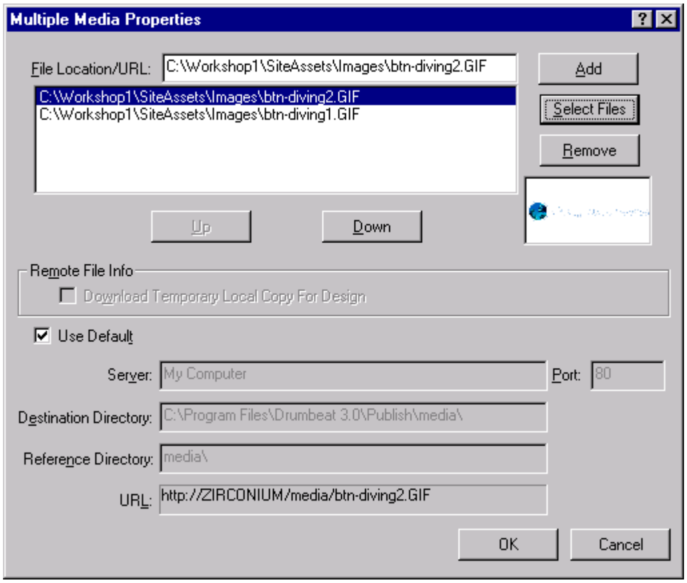
Port: 80

Destination Directory: C:\Program Files\Drumbeat 3.0\Publish\media\

Reference Directory: media\

URL: http://ZIRCONIUM/media/bk-diving.JPG

The Media Properties dialog box for cells containing one file.



Multiple Media Properties

File Location/URL: C:\Workshop1\SiteAssets\Images\btn-diving2.GIF

C:\Workshop1\SiteAssets\Images\btn-diving2.GIF
C:\Workshop1\SiteAssets\Images\btn-diving1.GIF

Remote File Info

□ Download Temporary Local Copy For Design

☑ Use Default

Server: My Computer

Port: 80

Destination Directory: C:\Program Files\Drumbeat 3.0\Publish\media\

Reference Directory: media\

URL: http://ZIRCONIUM/media/btn-diving2.GIF

In the Multiple Media Properties dialog box you can change the stacking order of the files in the cell.

Remote File Info if you populate a content table cell with remote content (by referencing a URL), you will have the option to download a local copy on your layout for design purposes. If you do not choose to do this, the file will not be visible in the layout at design time.

Use Default when Use Default is selected, media files will be published to the default directories defined in the Publish Settings (Publish > Publish Settings). Deselect this option to publish the file to a different location, specifying the server, destination directory and reference directory, as necessary.

Entering Text into content table cells

To enter text into a content table cell:

Right-click a content table cell and choose Insert. The text editor will appear for entering text. The column type automatically defaults to Text, Formatted.

Formatted and unformatted text

The default text type is Formatted. In certain circumstances you may want to choose Unformatted for the text type. Choose Unformatted as the text type if:

- The text you are entering is a URL or file path.
- The text includes HTML tags that you want to be interpreted as HTML code rather than text.

Referencing media files by URL

To reference a media file by URL in a content table:

1. Set the column properties to Type: Text and Text Type: Unformatted.
2. In the content table cell, enter the path to the media file. This can be a complete URL or the relative path from the root publish directory.
3. To use the file on the layout, place the appropriate SmartElement for the media type on the layout. In the Attributes tab for the SmartElement, for content, select Static Content. In the Content dialog box, choose the content table and the row and column that contains the file path.

If you use a path to point to a media file in this manner, you must post the file to the appropriate directory on the server yourself, since Drumbeat understands the content to be text rather than a file. On the other hand, files that are dropped into a content table cell from the Asset Center will be published by Drumbeat at publish time.

Setting row and column properties

You can assigning your own names to the rows and columns in a content table to aid organization. Each column in a Drumbeat content table has not only a name but a set of properties. These properties include the type of content and media or file type. You may also specify if you would like to allow multiple files in one cell, such as multiple images for a multistate graphic or animation sequence. All cells in a column must have the same media type. You cannot place a text file, for example, in a column that is designated as an image type.

To rename a row in a content table:

Right-click the row header. In the Row Name dialog box, enter the new name.

To view and set the properties of a column:

1 Do one of the following:

- Right-click on the first column header and select Properties from the pop-up menu.
- Double-click the column header.

2 In the Column dialog box, specify a name for the column and select the media type. You can allow multiple content items in the column cells by selecting this option.

Media file types include:

- Unspecified: Allows you to add any of the media types into a column (the first item added to the column will determine the type).
- Animation
- Audio
- Document
- Executable
- Html
- Image
- Metafile
- Video

Column types are determined at the time of import for content tables generated from external CSV files and generally do not need to be changed. However, you might want to rename the columns and rows in the table. Column properties for ODBC content tables cannot be changed.

Editing content tables

You can add rows or columns to any content tables that you have created manually or have generated from an external CSV file. (You cannot add rows or columns or otherwise change the content of ODBC content tables, whose content resides in the external database.)

To add a row or column to a content table:

1 Do one of the following:

- On the Content Center toolbar, click the menu arrow on the Add Row/ Column button and choose Add a New Row or Add a New Column (or simply click the button to use the current default selection). Select the header of a column or row first, then click the button to insert at the chosen point (insertion will be made before the selection).
- With the CSV content table in view, click the Edit Content Table button on the Content Center toolbar. In the Content Table dialog box, using the spin control for columns, increase the number of columns by one. (This adds a row or column to the end of the table.)
- Right-click the header of a column or row at the point you want to insert a new column or row and choose Insert (or Insert Row). The new column or row will be inserted before the selected column or row.

2 Do one of the following:

- For a new column, in the Column Properties dialog box, specify the name of the column (required) and the type of content (optional at this time).
- For a new row, specify the row name (the name will default to autonumbered if you do not provide a name).

Rows and columns can also be deleted or copied and pasted to reorganize the table as you wish. In addition, you can clear the contents of all cells in a row or column.

To cut, copy, delete, or insert a column or row:

Right-click the header of the column or row and choose the option you want: Delete, Cut, Copy, or Clear Contents (to leave the row or column intact but remove any content from cells).

The content of cells can be copied or moved from one cell to another within a table (in manual and CSV tables only). The cells must have the same column type properties.

To move media from one cell to another in a content table:

Click and drag the contents from one cell to another.

Displaying thumbnails in content tables

The display of thumbnail images in content tables can be turned on or off and tailored to your preferred display method.

To turn off display of thumbnail images in content tables:

1 Select File > Preferences > Drumbeat and the Content tab.

2 In the Content tab, deselect Display.

When the thumbnail display is turned off, images will be referenced by local path in the content table cell.

If you want images to fill the cell space (images will not preserve their original dimensions), you can select Display and Stretch to Fit in the Content tab of Drumbeat Preferences.

To rebuild the thumbnail previews of images in a content table:

Select Tools > Rebuild Thumbnails from the main menu.

Using content items from content tables

Items in content tables can be associated with SmartElements on the layout. The same item may be used multiple times to populate SmartElements on different pages. There are several different ways to designate the content of a SmartElement from a content table cell:

- Drag the contents of the cell to the layout. The content autogenerates the appropriate SmartElement and sets the content attribute for the SmartElement to Static Content binding to the correct cell in the table.
- Drag the contents of a cell to a SmartElement on the layout. The SmartElement must be of the appropriate type for the media.
- Set the Content attribute on the Attributes tab for a SmartElement. When populating site-level elements (recordSets, Cookies) this is the only way to bind the content to the element. Another example is when you want to set the Content attribute to the Current Row, which allows for fast and easy updates.

Populating page-level SmartElements from a content table

Content tables can be used to populate page-level SmartElements. Page-level elements are client-side elements that live and die from page to page. Using tables to populate elements is particularly useful if you will be using a content item multiple times on your site. If you need to update or replace that piece of content in every location where it appears on your site, you can place the new file in the same cell as the existing file. Each SmartElement in your site that references that content table cell will update to reflect the new piece of content.

The simplest way to use the contents of a content table cell is to simply drag and drop the contents to the layout. You can drag images, text, audio, and video files to the layout. Drumbeat automatically generates the appropriate SmartElement and binds the element to the table cell. You can then change the attributes of the element in the Attributes tab for the element.

You can also set or change the existing contents of a SmartElement already on the layout to derive its content from a content table cell.

To set or change the contents of a SmartElement to a content table cell:

1. Right-click a SmartElement on the layout and choose Attributes.
2. From the Attributes tab, select Static Content from the Content pop-up menu. (If you want to change the contents, click the ellipsis button next to the Content pop-up menu.)
3. In the Content dialog box, select the content table from the pop-up menu that contains the content you want to reference. Select the row and column locations of the content table cell that contains the content you wish to use. A thumbnail display of the content appears in the Content Preview area. Click OK.

Populating site-level Smart Elements

Content tables can be used to populate site-level SmartElements. Using tables to populate elements is particularly useful if you will be using a piece of content multiple times on your site. However, there is an added advantage when populating site-level elements. You can pass data to one page, change the data, and pass the modified data set to another page. This means that in Drumbeat a single instance of the site-level SmartElement can be used on multiple pages. Examples of these objects include COM objects, recordsets, and the Request and Session objects.

A very useful and common site-level element is the recordset, which derives its content from an ODBC content table. To use recordsets, the Server Application Support for the site must be set to ASP.

To populate a Recordset SmartElement:

1. Build an ODBC content table.
2. From the Asset Center or the SmartElement toolbar, drag a recordset element to the layout.
3. The recordset appears in the Basement. Right-click the recordset and choose Attributes.
4. In the Attributes tab, give your recordset a name. Choose a content table from the pop-up menu. Only ODBC content tables in your site will be displayed.

You won't see the recordset on the page. It is just holding data to be used or modified on a page. Now you can populate both page-level and site-level elements with items in the recordset. If you want to use the same recordset on other pages, query the Asset Center for site-level elements and drag the recordset to a page.

Using the Current Row content setting for quick changes

When using content from content tables, you can use the Current Row setting to enable changes of multiple elements at once on a page (or throughout your site). Items of the same type with different themes can be placed in the same column, with each row containing associated groups of elements.

For example, you can have a graphic theme represented on each row, with different backgrounds, buttons, and images. As long as each SmartElement that derives its content from that content table is set to Current Row, selecting a new row in the content tables will change all the elements at once.

To change content using the Current Row content setting:

1. In the Attributes tab, set the Content property for an element to Static Content.
2. In the Content dialog box, choose the content table and the column that relates to the SmartElement. For row, select Current Row.
3. To change the content for all items with this association, open the content table and select a new row by clicking on the row header.

Creating AutoTables from content tables

Content tables can be listed in the Asset Center. You can drag content tables from the Asset Center to the layout to create AutoTables.

To create an AutoTable from a content table:

1 Do one of the following:

- Query the Asset Center for content tables, then select a table and drag and drop it to the layout.

  The content table automatically creates an AutoTable SmartElement.

- Drag and drop an AutoTable SmartElement from the SmartElements toolbar onto the layout.

2 In the Attributes tab for the AutoTable, for content, select Static Content (for a manual or CSV table) or Recordset (for an ODBC content table). Click the ellipses button to bring up the Source dialog box.

3 In the Source dialog box, choose the columns you want to display in the AutoTable by selecting those you don't want to display and clicking Remove. You can also change the order of the display by selecting columns and clicking Move Up or Move Down.

Note: Columns of unspecified type are not included in the AutoTable.

You can add styles to the AutoTable by dragging a style from the Asset Center or assigning styles to rows and headings in the Attributes tab. For more about applying styles to tables, see "Using styles" on page 106.

CHAPTER 10
SmartElements

SmartElements are the building blocks used to assemble web applications. In Drumbeat, SmartElements are divided into two types: standard and registered. These elements display their content at different times: standard SmartElements display content at design time, and registered SmartElements display their content at browsetime.

Drumbeat ships with a number of both standard and registered SmartElements. Standard SmartElements are built into Drumbeat. When you first open Drumbeat, most of the standard SmartElements will appear on your SmartElement toolbar. Others are available in the Element library. Registered SmartElements are either components that have been pre-registered when you install Drumbeat 2000 or components you have manually registered into Drumbeat.

Standard SmartElements

Standard SmartElements display their content in the layout at design time and are available in the Scripting Tree (see “The Scripting Tree” on page 436). In the layout, you can add content and styles to standard SmartElements and actually view how they look before browsing. Standard SmartElements can not be customized, imported, or exported.

Standard SmartElements in Drumbeat 2000:

SmartElement	Description	Browser Support
Animation	Used to create animated images. You can add multiple-image files (in GIF format), or a single image in which there are multiple frames. This SmartElement has been engineered for efficient and fast downloading.	IE 4.0 Any Version 4.0 Nav. 3.0
Audio	Used as a container for audio files that play as background sound. Files are autoplay only, with a loop property that can be set. Common audio file types supported include au, mid, midi, ra, ram, and wav extensions. This SmartElement appears in the Basement.	IE 4.0 Any Version 4.0 Nav. 3.0
AutoTable	Enables display of content in a table format. The content for an AutoTable must come from a content table, which may be populated manually, from a CSV file, or from a database query (see "Content Tables" on page 199). The AutoTable SmartElement allows for linking to a URL column within the content table.	IE 4.0 Any Version 4.0 Nav. 3.0 Generic
Checkbox	Can be used for true/false values in a form.	IE 4.0 Any Version 4.0 Nav. 3.0 Generic
Command	The Command element is used for a recordset generated by a stored procedure that returns values. For more information, see "The Command object" on page 303. The Command element appears in the Basement and is only available to ASP sites.	IE 4.0 Any Version 4.0 Nav. 3.0 Generic
Cookie	Contains such information as passwords, lists of visited pages, or time-stamps. The Cookie element is a site-level element. It appears in the Basement of the layout, allowing you to apply interactions to set and retrieve the data stored in the Cookie.	IE 4.0 Any Version 4.0 Nav. 3.0
DataBinding	Copies a data file to a visitor's machine so that browsed pages can be dynamically populated locally without returning to the server. The DataBinding element appears in the Basement.	IE 4.0 only

SmartElement	Description	Browser Support
Dropdown List	Added to a form to allow selection from a list of predetermined options. The Dropdown List element can be populated manually, from a content table of any type, or via a server expression.	IE 4.0 Any Version 4.0 Nav. 3.0 Generic
DynaImage	Holds multiple image files. Scripting can then be applied via point-and-click interactions to animate the image in various ways.	IE 4.0 Any Version 4.0 Nav. 3.0
Edit Box	Can be used on forms to allow for free-form text entries.	IE 4.0 Any Version 4.0 Nav. 3.0 Generic
Form Button	Used on forms and can be of three types: Submit, Reset or Button. Use the Attributes tab to label and select the type of button. Form buttons with the appropriate scripting are automatically generated on DataForms and PageSets.	IE 4.0 Any Version 4.0 Nav. 3.0 Generic
Hidden	A nonvisible data field that holds name-value pairs. The value associated with the hidden field can be modified using client-side JavaScript. When the form is submitted the Hidden Form element is submitted as part of the form in the same way that visible form elements are submitted. The Hidden Form element appears in the Basement, as do all nonvisible elements.	IE 4.0 Any Version 4.0 Nav. 3.0 Generic
Horizontal Rule	Used to add a rule to visually divide a page. Corresponds to the <HR> HTML tag.	IE 4.0 Any Version 4.0 Nav. 3.0 Generic
HTML Passthrough	Allows you to use precomposed HTML code on a Drumbeat page, which Drumbeat will publish unaltered. You can use your own code by dropping an HTML file from the Asset Center, or by assigning an HTML file in the Attributes tab.	IE 4.0 Any Version 4.0 Nav. 3.0 Generic
Image	Used as a container for all single images used on the layout. Drumbeat supports all of the most common image files. Supported file formats: BMP (bitmap) GIF (graphic interchange format) JPG (joint photographic experts group)	IE 4.0 Any Version 4.0 Nav. 3.0 Generic

SmartElement	Description	Browser Support
Image Button	Contains multiple images that correspond to the on/off states of a rollover image button.	IE 4.0 Any Version 4.0 Nav. 3.0
Image Checkbox	Contains multiple dynamic images creating a graphical element with standard checkbox functionality.	IE 4.0 Any Version 4.0 Nav. 3.0
Image Map	Contains a single image to which multiple hotspots or hyperlinks can be added.	IE 4.0 Any Version 4.0 Nav. 3.0
Image Radio Button	Contains multiple dynamic images comprising a graphical button with standard radio button functionality.	IE 4.0 Any Version 4.0 Nav. 3.0
List Box	Added to a form to create a scrollable list of predetermined options. The List Box may be populated manually, from a content table of any type, or via a server expression.	IE 4.0 Any Version 4.0 Nav. 3.0 Generic
Looper	Allows you to create a data loop on a page to display multiple records from a database. See “Database editing interactions” on page 364 for further information. The Looper SmartElement is available only to ASP sites.	IE 4.0 Any Version 4.0 Nav. 3.0 Generic
Radio Button	Used to provide a set of options on a form from which users may select only one appropriate choice. They can be related to preset values to correspond to a database value in the Attributes tab.	IE 4.0 Any Version 4.0 Nav. 3.0 Generic
Recordset	Represents a data source for content on a page. One recordset is created for each database query that is used on a page. See “Using recordsets” on page 335 for information on setting the attributes for recordset elements. The recordset element is available only in ASP sites.	IE 4.0 Any Version 4.0 Nav. 3.0 Generic
SmartSpacer	Creates an area of dynamic space on the layout that adjusts the required spacing based on the content. The SmartSpacer element is used effectively on PageSets and DataForms to adjust the content for varying record size.	IE 4.0 Any Version 4.0

SmartElement	Description	Browser Support
T Text	Used as a container for text on the layout. Content can be added manually or may be derived from a content table or external text file. Hypertext links can be created within the Text SmartElement.	IE 4.0 Any Version 4.0 Nav. 3.0 Generic
⏲ Timer	Starts a timer and executes JavaScript events or jumps to an HTML page (HREF) on a timer expiration. The power of this SmartElement is in its use as a scripting tool.	IE 4.0 Any Version 4.0 Nav. 3.0

Registered SmartElements

Registered SmartElements are those components that have been preregistered when you install Drumbeat or that you have manually registered into Drumbeat. They display their content at browse time, after they have been published. You can edit the code of registered SmartElements in the Element library, as well as export and import them. See "Using the Element Library" on page 234.

SmartElements	**Description**
Acrobat Viewer	This SmartElement allows you to embed Adobe Acrobat files (PDF files) in your web pages. See the document in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\AcrobatViewer) for attributes and implementation information.
Business Tools	Calendar ActiveX control: The Microsoft Access Calendar ActiveX Control is a preregistered SmartElement. To use this element you must import the SDE file into the SmartElement Library. You can access the SmartElement Library from the Drumbeat Tools menu. Only for IE 4.0 and above. Calendar Scriptlet: This element uses Microsoft's scriptlet technology. Hex Calculator: This element is an applet and works in all Java-capable browsers. As its name suggests, the calculator is capable of doing computations in hexadecimal, in addition to binary and standard decimal calculations. Pocket Calculator: This is an applet that works in all Java-capable browsers. It provides simple arithmetic computations and a square root function. See the documents in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\BusinessTools) for attributes and implementation information.

SmartElements	Description
CGI	CGI FormProcessor SmartElement: Creates a Perl script to process the form elements on the page. CGI RandomImage SmartElement: Creates a Perl script to generate random images on a page reload/refresh with an optional URL link or text associated with each image. CGI RandomText SmartElement: Creates a Perl script to generate random text on a page reload/refresh. See the documents in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\CGI) for attributes and implementation information.
DayNight	This SmartElement is configured to divide the day into time-frames such as morning, afternoon, etc. Associated DayNight contracts allow interactions between other SmartElements to dynamically update display properties based on the time of day that the page is browsed. See the document in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\DayNight) for attributes and implementation information.
EmailCDO	This SmartElement is an ASP server-side SmartElement that sends e-mail messages. This SmartElement allows you to send e-mail using the COM NewMail object, which is part of the Microsoft Collaborative Data Objects (CDO) package. This package comes with Microsoft Internet Information Server (IIS) 4.0 and the NT Server Options pack. See the document in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\EmailCDO) for attributes and implementation information.
File Upload	These SmartElements allow a file to be uploaded from the client (the computer running a web browser) to the host site (the computer running the http: server). There are two ways to do this. One is to use the Microsoft proprietary mechanism embodied in an ActiveX control and corresponding server software. The other way is to use the W3C proposed standard. Both of these techniques are available as distinct Drumbeat SmartElements. There is one SmartElement that implements the Microsoft ActiveX File Upload control. There are two SmartElements that implement the W3C file upload mechanism. Of the two W3C file upload SmartElements, only one should be used on a page. See the documents in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\FileUpload) for attributes and implementation information.

SmartElements	Description
Floating Element Manager	The Floating Element SmartElement in conjunction with the "Float element relative to browser, managed by FloatManager" point-and-click interaction supports "floating" SmartElements in specific locations of the client browser window, no matter how the window is scrolled or resized. See the document in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\FloatingElementManager) for attributes and implementation information.
FrontPage	FrontPage Ad Rotator: This Drumbeat SmartElement implements the FrontPage Ad Rotator. This is an applet that takes a set of images as input and essentially presents them as a slide show. You can click on the applet to take you to a URL. FrontPage Form Handler: This Drumbeat SmartElement implements the FrontPage Form Handling bot. To use this SmartElement you must have the FrontPage extensions installed on your server. FrontPage Hit Counter: This Drumbeat SmartElement implements the FrontPage Hit Counter. To use this SmartElement you must have the FrontPage extensions installed on your server. See the documents in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\FrontPage) for attributes and implementation information.
LivePicture Viewer	This suite contains four LivePicture Viewer SmartElements: Plug-in version for Flashpix, Panoramas and 3D objects Java version for Panoramas Universal Viewer via a hypertext link Universal Viewer via an image link See the documents in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\LivePicture) for attributes and implementation information.
Maps	This suite contains four Map SmartElements: Vicinity Map is designed for drawing street maps to locate an address, like a business address. Tiger Map is designed to draw maps on a much wider scale and includes more land use information than street information. National Weather Service Map is a clickable National Weather Service map. NASA Satellite Map plots the location of a variety of satellites in real time. See the documents in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\Maps) for attributes and implementation information.

SmartElements	Description
Menus	SiteMenu ActiveX Control: To use the Menu, you must first install the control on your system. Using Internet Explorer 4.0 or later, go to http://www.quiksoft.com/sitemenu/samples/horiz/ to see an example and install the control on your system. DHTML Cascading Menu DHTML Menu: This element will work in both 4.0 browsers. It drops down as soon as the mouse moves over the element. This makes a great navigation element that uses a small amount of screen real estate and gives pages a professional look. See the documents in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\Menus) for attributes and implementation information.
Miscellaneous	ASP Include File SmartElement: This is much like the Passthrough HTML element in that you are on your own as far as the contents of the file to be included. If you want the include statement to occur before other HTML code and there are no visible elements, you can check the hidden checkbox for this SmartElement. If you want the file included before other ASP code there is a contract for this purpose. Data Table JavaScript SmartElement: This is a JavaScript element that exposes the contents of a content table on the client in JavaScript. During publish the contents of the content table is rendered in JavaScript. On the client, this element has methods to access the data programmatically. JavaScript Random Image: This JavaScript element creates JavaScript to generate random images on a page reload/refresh with an optional URL link or text associated with each\ image. Universal Embed PlugIn: This element generates a generic EMBED tag that only supports the SRC attribute of the EMBED tag. Any file type can be assigned to this component via the Drumbeat content attribute. For example, you can point it to a PDF file and the browser will bring up Acrobat Viewer (if it is installed on the user's system). PassThrough HTML SmartElement: This element allows you to enter anything you want and the Drumbeat publishing engine will output whatever you entered into the web page. See the documents in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\Miscellaneous) for attributes and implementation information.

SmartElements	Description
Navigation	DHTML Recordset Navigation: To be used in sites that target Navigator 3.0 and later browser support. DHTML Outline: This custom element generates DHTML for IE 4.0. It is a user-expandable outline. Scriptlet Recordset Navigation: This Recordset Navigation SmartElement uses the Scriptlet technology and is available for IE4.0 and above browsers only. See the documents in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\Navigation) for attributes and implementation information. You can also read about these Navigation elements in Chapter 4, Navigation.
NetCharts	This is NetFactory, Inc.'s charting and graphing applets package. It is an excellent package for visually displaying data See the documents in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\NetCharts) for attributes and implementation information.
QuickTime	This package contains the QuickTime and QuickTimeVR PlugIn SmartElements. These SmartElements revise earlier versions by simplifying the specification of links and frame targets consistent with QuickTime-specific methods for handling link parameters. See the documents in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\QuickTime) for attributes and implementation information.
Real Media	The SmartElement RealMedia ActiveX generates an EMBED tag nested within an OBJECT tag and supports all browsers that support either the EMBED or OBJECT tag. However, scripting to this SmartElement will not work in Netscape browsers. See the documents in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\RealMedia) for attributes and implementation information.

SmartElements	Description
Scrolling Applets	Scrollable Image Map: The Scrollable ImageMap SmartElement is an applet that allows the scrolling of a large image map either by scroll bars or by grabbing and dragging. The image map is implemented as a collection of rectangular hotspots. Hotspots can optionally be configured to display a border or to be highlighted on mouse over. Scrolling Text Applet: The Scrolling Text SmartElement is an applet that provides a configurable window of scrolling text that can be formatted using Drumbeat style definitions. Drumbeat styles specify formatting attributes for text such as font name, size, style, and color. Ticker Tape Applet: The Ticker Tape SmartElement is an applet that provides a configurable window of horizontally scrolling text that can be formatted using Drumbeat style definitions. See the documents in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\ScrollingApplets) for attributes and implementation information.
User Interface	DHTML Media Controller: The DHTML Media Controller can be used in sites that target Navigator 3.0 browser support and above. All you have to do is drop the SmartElement onto the layout along with any type of Media Player (such as the Real Video Plug-in). Then you assign the interaction, Control Media Playing, to the Media Player and the Media Controller. DHTML Tabbed Dialog: This element implements a Windows-style tabbed dialog box. DropZone: This SmartElement defines a rectangular region that simulates a target. The SmartElement is used in conjunction with the "Set [target] the drop zone for [source]" interaction. The interaction can be found under the DragDrop category in the Interactions library. Standard Microsoft Controls: These SmartElements expose the standard Microsoft Controls to the Drumbeat author. These controls are implemented as ActiveX controls and as such are only available in IE4.0 and above. Spin Button: Use interactions to bind this control to an edit box. This yields a spin button that can modify the numeric value of the edit box. ScrollBar: Use interactions to bind this control to an edit box. This yields an edit box whose numeric value is changed by interacting with the scroll bar. Radiation Button: This element implements a glowing button. It is implemented as an element and a Microsoft glow filter that varies over time to get a pulsating effect. It works with text and images that have transparency. It does not work on images that do not have transparency. See the documents in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\UserInterface) for attributes and implementation information.

SmartElements	Description
Macromedia Shockwave	Macromedia Shockwave Authorware PlugIn: The Macromedia Shockwave Authorware PlugIn SmartElement supports the embedding of Macromedia Authorware map files (.aam). Macromedia Shockwave Director Plugin: The Macromedia Shockwave Director Plugin SmartElement supports the embedding of Macromedia Director files (.dcr). Macromedia Shockwave Flash Plugin: The Macromedia Shockwave Flash Plugin SmartElement supports the embedding of Macromedia Flash files (.swf), as well as files (.spl) from its predecessor, Future Splash. See the documents in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\ShockWaveFlash) for attributes and implementation information.
Thing Player	This element implements the Parable Thing Player. Thing Player is a multimedia player that plays proprietary format files. For detailed information on this technology and a trial version of the Thing authoring software visit parable’s web site at www.thingworld.com. See the document in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\ThingPlayer) for attributes and implementation information.
VRML World Plug-In	VRML World Plug-In Package: Contains the VRML World Plug-In SmartElement. The VRML World Plug-In SmartElement supports the embedding of an animated or static three-dimensional VRML World file (.wrl). Client viewers need an installed VRML browser in order to view the VRML World in the published web page. The Attribute tags are not supported in all VRML browsers but will not affect the loading or viewing of the VRML World in browsers, which do not support the tags. See the documents in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\VRML) for attributes and implementation information.
Windows Media Player	The Windows Media Player is capable of playing a wide variety of multimedia content. This includes AVI video files, QuickTime video files, Real Media Streaming media files, Microsoft NetShow streaming media files, Macromedia Future Flash media files, and more. It is an all-in-one solution for rendering a plethora of media to your web browsers. See the documents in the SmartElements directory of the Drumbeat program files (Drumbeat 2000\SmartElements\WindowMediaPlayer) for attributes and implementation information.

For detailed information about registering SmartElements and the Element Library, see “Registering SmartElements” on page 240.

Importing registered SmartElements

Once a component has been registered into Drumbeat, it becomes a SmartElement that can be exported (creating an SDE file) and imported for anyone to use.

To import a SmartElement:

1. In the Asset Center, click Add Assets and select SmartElements.
2. In the Element library, click Import.
3. Navigate to an SDE file and click OK. By default, SDE files are saved under the SmartElements directory in the Drumbeat installation directory structure.

Once it has been imported, you can add the SmartElement to the SmartElement toolbar if you wish, or you can use it by dragging and dropping it from the Asset Center.

Working with standard and registered SmartElements

Customizing the SmartElement toolbar

You can customize your SmartElement toolbar to display any standard or registered SmartElements you choose.

To add SmartElements to the SmartElement toolbar:

Do one of the following:

- Click the Element Library icon to open the Element library, and then click to select any SmartElement you want to add to the toolbar.
- Drag a SmartElement from the list of SmartElements in the Asset Center to the SmartElement toolbar.

To delete a SmartElement from the SmartElement toolbar:

Do one of the following:

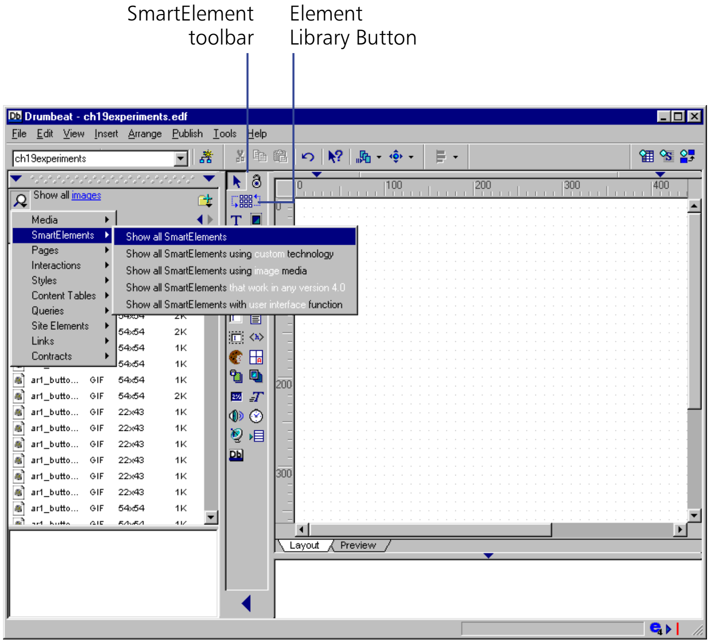
- Drag the SmartElement icon to the Element Library icon on the SmartElement toolbar.
- Deselect the SmartElement in the Element library.

Displaying and sorting SmartElements in the Asset Center

All SmartElements, whether shipped or later registered in Drumbeat, are assets just like media, styles, and contracts. They can be listed in the Asset Center.

To list SmartElements in the Asset Center:

1. Click the Locate Assets button (magnifying glass icon).
2. Select SmartElements > Show All SmartElements.


SmartElement toolbar

Element Library Button

Drumbeat - ch19experiments.edf
File Edit View Insert Arrange Publish Tools Help
ch19experiments
Show all images
Media
SmartElements
Pages
Interactions
Styles
Content Tables
Queries
Site Elements
Links
Contracts
Show all SmartElements
Show all SmartElements using custom technology
Show all SmartElements using image media
Show all SmartElements that work in any version 4.0
Show all SmartElements with user interface function
Layout
Preview

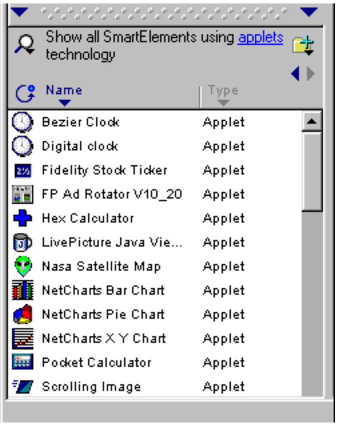
Bringing SmartElements into the Asset Center

Since the entire list of SmartElements is so long, they are classified in several ways so that you may sort the Asset Center to display only those SmartElements that are likely to interest you for the task at hand.

The major categories are:

- Technology Type. This divides the SmartElements into classes such as scriptlets, plug-ins, activeX, COM, and element collections. Custom and standard are also SmartElement types.
- Media. The classes are animation, audio, video, image, and text. You can query for SmartElements based on content type.
- Browser compatibility. You can also select for SmartElements that work in a specified browser type or that are specifically designed for a browser type.
- Function. Classes such as CGI, clocks, business tool, text, command. Standard is also a functional category. You can create new functional categories of your own.

The following figure shows the SmartElement asset list sorted for the technology type applets. The list is alpha-sorted in descending order on the element name. As with all such asset listings, the sort term may be changed by clicking on different column labels in the header, and the sort toggles between descending and ascending as you continue clicking. The current sort term is in bolder type than are the other headers.

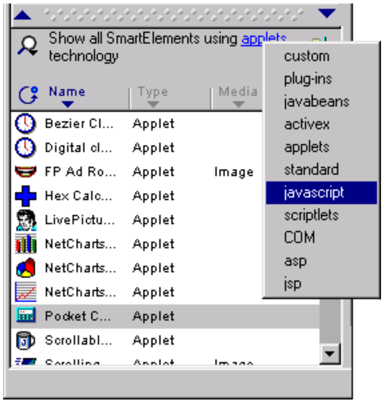


Show all SmartElements using applets technology

Name	Type
Bezier Clock	Applet
Digital clock	Applet
Fidelity Stock Ticker	Applet
FP Ad Rotator V10_20	Applet
Hex Calculator	Applet
LivePicture Java Vie...	Applet
Nasa Satellite Map	Applet
NetCharts Bar Chart	Applet
NetCharts Pie Chart	Applet
NetCharts X Y Chart	Applet
Pocket Calculator	Applet
Scrolling Image	Applet

SmartElement list sorted for applets

The following figure shows how to switch classes within one category by clicking on the hyperlinked class name in the list header and selecting from the pop-up menu.



Show all SmartElements using applets technology

Name	Type	Media
Bezier Cl...	Applet	
Digital cl...	Applet	
FP Ad Ro...	Applet	Image
Hex Calc...	Applet	
LivePictu...	Applet	
NetCharts...	Applet	
NetCharts...	Applet	
NetCharts...	Applet	
Pocket C...	Applet	
Scrollabl...	Applet	

Switching classes within a category

Using the list in the Asset Center, you can do one of two things to deploy new SmartElements. You can drag them and drop them onto your SmartElement toolbar to make them readily accessible for layout or, if you believe this is a one-time need, you can drag the object directly into the layout.

Each instance of a SmartElement from the Element library is a separate object with its own attributes, which define the behavior of the element on the page. Although you might set up an ActiveX control as a SmartElement, for instance, the way it acts can be partially determined by the attributes you give it after it is placed. These can be different from other instances of the same SmartElement somewhere else in the site.

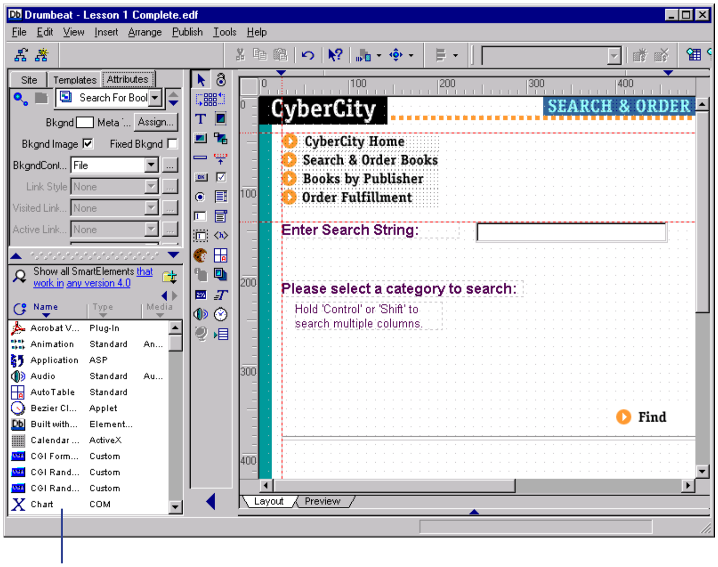
To activate a SmartElement after it has been placed on the layout:

Do one of the following:

- Right-click a SmartElement on the layout and choose Possible Interactions to display a list of available point-and-click interactions for this SmartElement. (For more information, see “Interactions Center” on page 397.)
- Examine, edit, and create the contracts that underlie the element interactions. You need to understand JavaScript to create contracts, but not to use them. (For more information, see “Contracts” on page 417.)
- Write free-form JavaScript that acts on your SmartElements in ways that may be unique to this application. You need to be familiar with JavaScript to use the Script Center. For more information, see “Scripting in the Script Center” on page 437.

Adding SmartElements to pages

SmartElements can be dragged onto the layout from either the SmartElement toolbar or the Asset Center.

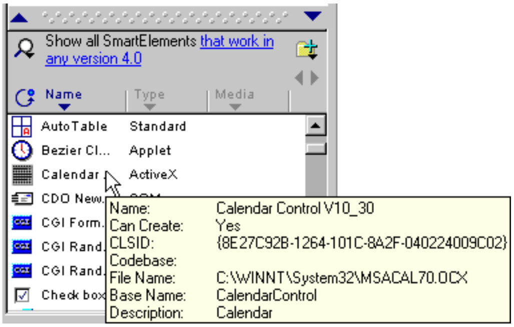

Drumbeat - Lesson 1 Complete.edf
File Edit View Insert Arrange Publish Tools Help
Site Templates Attributes
Search For Bool
Bkgnd Meta Assign...
Bkgnd Image ☑ Fixed Bkgnd □
BkgndConl... File
Link Style None
Visited Link... None
Active Link... None
Show all SmartElements that work in any version 4.0
CyberCity
SEARCH & ORDER
CyberCity Home
Search & Order Books
Books by Publisher
Order Fulfillment
Enter Search String:
Please select a category to search:
Hold 'Control' or 'Shift' to search multiple columns.
Find
Layout
Preview

Name	Type	Media
Acrobat V...	Plug-In	
Animation	Standard	An...
Application	ASP	
Audio	Standard	Au...
AutoTable	Standard	
Bezier Cl...	Applet	
Built with...	Element...	
Calendar ...	ActiveX	
CGI Form...	Custom	
CGI Rand...	Custom	
CGI Rand...	Custom	
Chart	COM	

Asset Center

SmartElements are added to pages from the SmartElement toolbar or the Asset Center.

The SmartElement toolbar is where you place commonly used SmartElements. It is a customizable holding area (see “Importing registered SmartElements” on page 224). The SmartElement toolbar displays icons for each SmartElement. If you hover over an icon, a tooltip will display the name of the SmartElement.



Name	Type	Media
AutoTable	Standard	
Bezier Cl...	Applet	
Calendar	ActiveX	
CDO New...	COM	
CGI Form...		
CGI Rand...		
CGI Rand...		
Check box		

Name:	Calendar Control V10_30
Can Create:	Yes
CLSID:	{8E27C92B-1264-101C-8A2F-040224009C02}
Codebase:	
File Name:	C:\WINNT\System32\MSACAL70.OCX
Base Name:	CalendarControl
Description:	Calendar

Display basic information about a SmartElement in a tooltip by hovering over the name in the Asset Center.

In the Asset Center, you can locate any SmartElement at any time. The Asset Center lists all the SmartElements that are accessible to your site (see “Displaying and sorting SmartElements in the Asset Center” on page 225).

To add a SmartElement to the layout from the Asset Center:

1. Click on the Locate Assets button (magnifying glass icon).
2. Select SmartElements > Show All SmartElements or choose the filter you want from the pop-up menu.
3. Select the desired SmartElement and drag and drop it to the layout.

To add a SmartElement to the layout from the SmartElement toolbar:

1. From the SmartElement toolbar, select the SmartElement.
2. Do one of the following:

- Click the layout to add the element to the page.
- Drag the element to the layout.

Setting SmartElement attributes

Each SmartElement has its own set of attributes appropriate to that element. Once the SmartElement is placed on a page, you can access its attributes in the Attributes tab.

To set the attributes of a SmartElement:

1. Select the SmartElement container on the layout or in the Basement, if it is a Basement element. Click the Attributes tab or right-click and choose Attributes.
2. In the Attributes tab, set the desired attributes for the SmartElement.

Using the HTML Passthrough SmartElement

Standard SmartElements are very easy to use, as what you see in the layout is usually what you get when you view the page in a browser. The HTML Passthrough standard element is an exception to this element rule, in that you cannot view the content at design time. You need to publish the page to view the output of the HTML code.

To use the HTML Passthrough element:

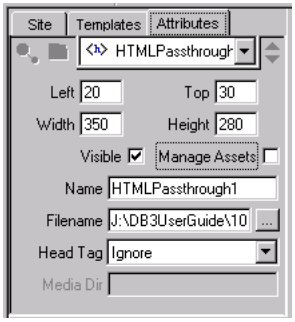
1 Do one of the following:

- Select the HTML Passthrough SmartElement on the SmartElement toolbar and drag and drop it onto the layout or click the layout to drop it where you want.
- Select the HTML Passthrough SmartElement, then click the layout and hold the mouse down as you draw a box of a desired size.

2 From the Asset Center, locate and drag the HTML file to the container.

3 While the HTML Passthrough element is selected in the layout, click the Attributes tab in Site Manager.

4 Click the ellipses button next to the Filename text box to assign a file to the element.



Left 20 Top 30

Width 350 Height 280

Visible ☑ Manage Assets ☐

Name HTMLPassthrough1

Filename J:\DB3UserGuide\10

Head Tag Ignore

Media Dir

The attributes of the HTML Passthrough SmartElement.

The file you assign to the HTML Passthrough element must be a text file with proper HTML syntax. The file must include a <BODY> tag. Head tag information is optional (see below).

Other attributes specific to the HTML Passthrough element are:

- Manage Assets: Specifies whether to copy any referenced media to the publishing destination.
- Head Tag: Specifies whether to ignore or to include the information in the Head tag of the HTML file. If you choose to include, the information is added to the head of the page.
- Media Dir: Specifies the publishing destination of the media files.

Page- and site-level SmartElements

SmartElements are further classified as page-level or site-level. Both standard and registered SmartElements can be page- and site-level SmartElements. This means that they can be referenced (set or retrieved) at either the page or the site level.

There is a major difference between site-level elements that have been built into Drumbeat, referred to as standard, and those that have been registered, referred to as preregistered or registered. The difference lies in the way each are added to a page and how you set their attributes.

Once you have added a site-level element and set its attributes, you can use it on any page in the site without losing the set attributes. This technique works only with standard SmartElements. Site-level standard SmartElements are queried for in the Asset Center. To do this, select the Site Element option on the asset list from the Locate Assets button.

Registered SmartElements that are set or retrieved at the Site-level must be added to each page and have their attributes manually set each time. You must be very methodical, verifying that you have set the identical attributes on each page.

There are some basic differences between page- and site-level elements:

- Page-level elements are client-side elements that live and die from page to page. Since HTTP is a stateless protocol, it is impossible for the browser to track information from page to page.
- Site-level elements live on the server, with the one exception being the Cookie element. Site-level elements are part of the application server (ASP) and persist from page to page.

Displaying and sorting site-level elements

There are two standard site elements: recordset and cookie. These two elements are retrievable from the Asset Center. The third element, also retrievable from Asset Center, is the DataForm. The DataForm is the product of the DataForm wizard and is not a stand-alone element. A DataForm element is created when you step through the DataForm wizard.

To display site-level elements:

In the Asset Center click the Locate Assets button (magnifying glass icon) and choose Site Elements > Show All Site-level Elements.

Site-level SmartElement: Using a cookie

Here is an example of a standard site-level element using a Cookie SmartElement. You can set the value of the Cookie, how long it will persist, and its name in the Attributes tab. You can request values from the Cookie from page to page by querying for site elements in the Asset Center, and thereby using the same Cookie element on various pages, so you maintain the state of the value.

To add a Cookie to a page:

1. From the Asset Center, click the Locate Assets button, and choose SmartElements > Show All SmartElements.
2. Select the Cookie element and drag it to the layout. The Cookie will appear in the Basement of the page.

   Note: Elements are listed in alphabetical order. Click inside the Asset List. Type the letter 'C' to jump to elements beginning with that letter. Then, scroll down to the Cookie element.

Once you have set the Cookie on a page, it is available to all pages in the site and displays as a site-level element, much like a recordset.

To set or request a value from a Cookie using point-and-click interactions:

From the Asset Center, query for site elements. Then, select and drag the Cookie element from the Asset Center to the layout.

To set the Attributes of the Cookie element:

In the Basement, right-click the Cookie element and choose Attributes.

Secure	Makes the Cookie a secure Cookie (applies the ";secure" parameter to the Cookie string). Secure Cookies are sent to the server in an encrypted format via a secure channel (shttp).
Expires	This is the lifetime of the Cookie in days. After this number of days has elapsed, the browser will delete the cookie.
Domain	Enter the publishing domain. You need to change the domain each time you publish to a different server.
Path	Enter the path of the publishing folder. You need to change the path each time you publish to a different folder.

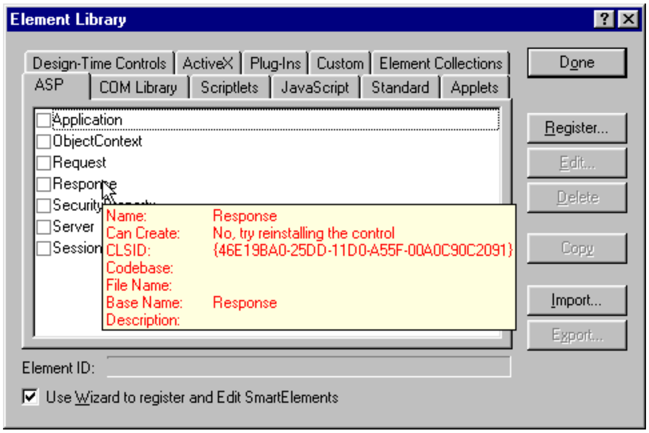
For a working demonstration of using a cookie to preserve visitor information, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

CHAPTER 11
Element Library

The Drumbeat Element Library is a tool for managing SmartElements. Using the Element Library, you have access to scores of SmartElements of various types, including Navigator plug-ins, Java applets, ActiveX controls, COM and DTC controls, JavaScript scripts and Scriptlets, and custom text or HTML. The Element Library also allows you to import SmartElements from other users or create new SmartElements or element collections of your own. In Drumbeat, you register each element (including class or file names and user-selectable properties) using either a tabbed dialog box or a wizard. After you have given Drumbeat this information, you can use the element like any other Drumbeat page element, without needing to know anything about HTML, Java, or JavaScript.

Using the Element Library

To examine, edit, manipulate and create SmartElements, use Element Library. To open the Element Library, click the Element Library button on the SmartElement toolbar, choose Tools > Element Library. SmartElements marked with a check are displayed on the SmartElements toolbar.



□ Application
□ ObjectContext
□ Request
□ Response
□ Security
□ Server
□ Session

Name:	Response
Can Create:	No, try reinstalling the control
CLSID:	{46E19BA0-25DD-11D0-A55F-00A0C90C2091}
Codebase:	
File Name:	
Base Name:	Response
Description:	

Element ID:

☑ Use Wizard to register and Edit SmartElements

Tooltips show the status of SmartElements in the library.

The tabs at the top of the window display the technology types supported by Drumbeat SmartElements. Click a tab to display the list of SmartElements for the selected technology type. Click Copy to create a new SmartElement, using the selected element as a starting point.

The Import and Export buttons let you share elements with other Drumbeat users. When adding a Java applet, for instance, you can pre-configure all the properties that control the applet. Once that information has been added, it can be exported to a file and sent to another Drumbeat user, who can import and use the same applet SmartElement on their computer.

When exporting an element, enter a name for the file in which the element will be saved and specify a directory location. The filename does not need to match the name of the SmartElement within Drumbeat, so you can add version information or other notes as part of the name. The file will be saved with an extension of .sde.

Imported SmartElements retain their original Drumbeat name, not their file names, and are placed into the currently displayed technology type.

SmartElement technology types

The SmartElement technology types recognized by Drumbeat include all of the standard web technologies supported by popular browsers. With the exception of the standard SmartElements and the ASP objects, these technology types ASPextend the range of interactive objects you can include on your web sites.

Extension of the Drumbeat working environment in this way is made possible by component architecture that, in most cases, can bring third-party applications into the Drumbeat arena—reading in all necessary methods and properties by a process known as introspection.

Standard

These are the SmartElements that are the core part of Drumbeat, and mostly the ones that are on your SmartElement toolbar by default. Unlike all other technology types, the standard SmartElements are generated by HTML and ASP only, and they cannot be edited. In general, standard SmartElements appear in the Document Object Model when placed on a page, and are available in the Scripting Tree for creating scripted interactions (see “The Scripting Tree” on page 436).

ASP

These are objects that are built into the ASP engine. They are part of the asp.dll file and are used to maintain the application and user session state of an ASP application. Objects such as form elements are stored in the Request object and each user gets a unique instance of the Session object to maintain their use of your application. The Session and Application objects have capturable events `onStart` and `onEnd`. ASP SmartElements appear in the ASP Object Model when placed on a page, and are available in the Scripting Tree for creating scripted interactions (see “Script Center” on page 433).

For an example of how to use the ASP Session Object to create multi-page forms, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

COM Library

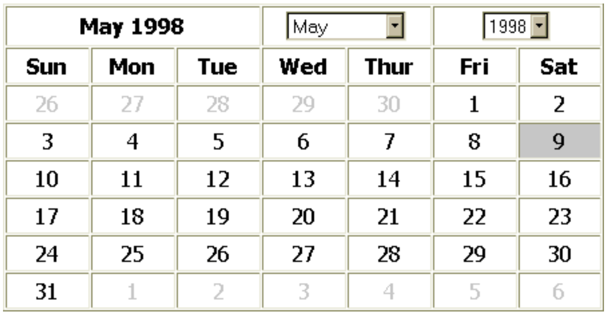
COM components are binary objects that adhere to the Component Object Model Interface and Communication protocols administered by Open Group. COM is a specification for inter-object communication. COM components are similar to the better-known ActiveX controls.

COM architecture allows developers to encapsulate middle layer business logic on small and nimble components. Drumbeat exposes each object's functionality in the development environment just as it does with a recordset. See also "ASP & COM Objects" on page 242.

For two working examples of using COM objects in Drumbeat (on sending a form using CDO NewMail COM and on using page counters) see the Drumbeat Support Center (macromedia.com/support/drumbeat).

Scriptlets

Scriptlets are HTML files with included JavaScript provide a particular functional component of a web page. One of the scriptlets included with Drumbeat, for instance, is a calendar page. Scriptlets are available only for pages targeted at Internet Explorer 4.0.



May 1998			May ▾		1998 ▾	
Sun	Mon	Tue	Wed	Thur	Fri	Sat
26	27	28	29	30	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	1	2	3	4	5	6

The Calendar Scriptlet

JavaScript

JavaScript type elements are JavaScript objects which can be contained in an HTML source page. Whereas Scriptlets can include HTML code to define how they are to be displayed, JavaScript elements do not display on a page at all. Instead, a JavaScript element is a convenient way of embedding methods of behavior in an HTML document so that they can be used by other parts of the page. The scripts need not have any connection with a displayed element. By making a SmartElement out of a collection of JavaScript functions, you make it easy to use the same scripts for all objects on all the pages you create. This also makes it easier to update all your pages when the scripts change.

Applets

An applet is a small application written in the Java programming language. Java applets can be run on any computer that supports a Java Virtual Machine, which makes them ideal for use on web sites. Java applets can produce animation, gather information, perform calculations, access network data, and more. While creating Java applets requires an in-depth knowledge of the programming language and a Java development environment, a huge number of useful Java applets have already been created by third parties. Using the Element Library, you can import applets and make them all work with Drumbeat. A number of Java applets are included with Drumbeat, including a calculator, a digital clock, an ad rotator, and scrollable text and image applets.

For an example of how to use the FrontPage ad rotator applet, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

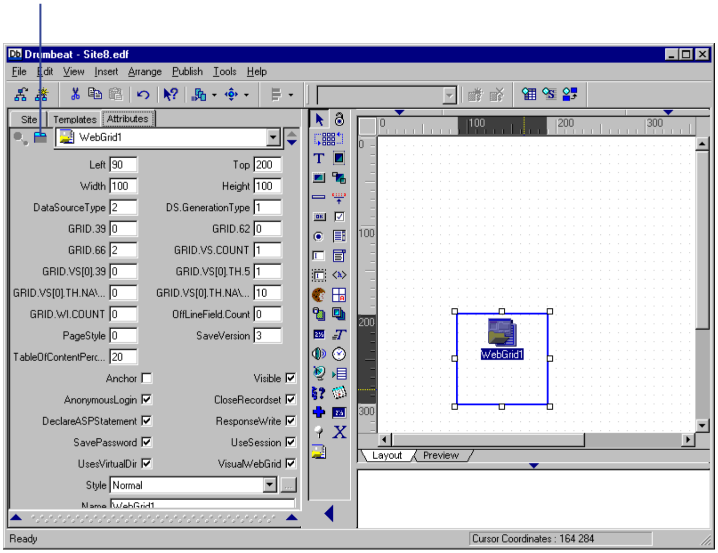
Design-time controls (DTCs)

Design-time controls (DTCs) are specialized ActiveX controls. They are native to Microsoft Visual InterDev and run within that environment on the server. A DTC typically generates HTML and ASP code that is then inserted into an ASP page by the control. In Drumbeat 2000 for JSP you can register either client-side DTCs or DTCs that generate JSP.

When you register a DTC Drumbeat scans and analyzes the registry to retrieve all of the methods and properties associated with the application.

Once a DTC has been registered as a SmartElement, Drumbeat can make its properties available for editing in the Attributes tab just like any other page object.

Native properties



WebGrid1

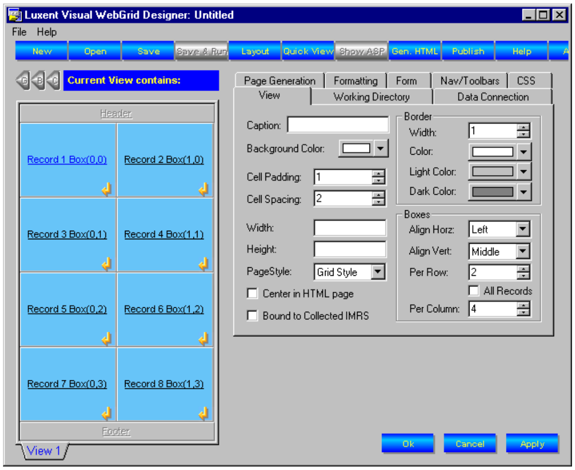
Left 90 Top 200
Width 100 Height 100
DataSourceType 2 DS.GenerationType 1
GRID.39 0 GRID.62 0
GRID.66 2 GRID.VS.COUNT 1
GRID.VS[0].39 0 GRID.VS[0].TH.5 1
GRID.VS[0].TH.NA\... 0 GRID.VS[0].TH.NA\... 10
GRID.WI.COUNT 0 OffLineField.Count 0
PageStyle 0 SaveVersion 3
TableOfContentPerc... 20
Anchor ☐ Visible ☑
AnonymousLogin ☑ CloseRecordset ☑
DeclareASPStatement ☑ ResponseWrite ☑
SavePassword ☑ UseSession ☑
UsesVirtualDir ☑ VisualWebGrid ☑
Style Normal
Name WebGrid1

Ready
Cursor Coordinates : 164 284

Drumbeat Attributes tab for the Luxent Visual WebGrid

At the top left of the Attributes tab is a button that you can click to display the properties native to the application.

Note: Introspection of DTC and ActiveX SmartElements can be a complex process, depending on the particular control and how it is written. It is not uncommon for introspection to take a significant time—several minutes, perhaps.



Luxent Visual WebGrid Designer: Untitled

Caption:
Background Color:
Cell Padding: 1
Cell Spacing: 2
Width:
Height:
PageStyle: Grid Style
□ Center in HTML page
□ Bound to Collected IMRS

Border
Width: 1
Color:
Light Color:
Dark Color:

Boxes
Align Horz: Left
Align Vert: Middle
Per Row: 2
□ All Records
Per Column: 4

Native properties for the Luxent Visual WebGrid Designer

ActiveX

An ActiveX control is a Win32 mini-program that can be embedded in web pages. Since it is written in code native to a particular processor, it generally attains better performance than an equivalent Java applet. Because it can download and install itself automatically, it can be more convenient than a browser plug-in for handling various types of file formats. RealVideo, Macromedia Flash, and Adobe Acrobat are all supported by ActiveX controls. Drumbeat works with ActiveX as with DTCs. Note that ActiveX controls are only supported in Windows browsers.

Note: If you are upgrading from Drumbeat 2.0, re-register any DTCs and ActiveX controls that were previously registered under Drumbeat 2.0. This is because introspection is more complete in Drumbeat 2000 and allows access to the native properties tab. Please do not change the names of properties listed by introspection of these types of controls, or delete any properties in the list. Drumbeat does not prohibit these options but they are likely to lead to difficulty in launching and using the native property tab.

Plug-ins

Plug-ins extend the capabilities of browsers. First supported by Netscape Navigator and later by other browsers, plug-ins make it possible to add support for streaming audio and video, various animation formats (such as Shockwave), and more to browsers that had only basic media-handling capabilities.

Custom SmartElements

Custom SmartElements include HTML tags that reference a type of object Drumbeat has not already been designed to use. You can use these for many purposes by writing the basic scripting code that generates the HTML yourself. This covers all non-standard web technology groups. See "Registering Custom Elements" on page 243.

Element collections

The Element Collections tab is the place where you can add groups of elements that you want to treat as a single entity. "Creating SmartElement collections" on page 244.

Registering SmartElements

SmartElements cannot be entered directly into Drumbeat in a scripting window or dialog box. They are mostly pre-written components such as applets and plug-ins. Drumbeat, however, needs to be told about the characteristics of each new SmartElement you add, in order to deploy it correctly. The process of adding SmartElements and describing them to Drumbeat is called registration.

Note: Drumbeat comes with several standard SmartElements, which are displayed under the Standard tab of the Element Library. You cannot edit, delete, import, and export this technology type.

To register and edit SmartElements:

1. Choose Tools > Element Library.
2. Click a tab and select a SmartElement.
3. To use the wizard, select Use Wizard and the bottom of the Element Library. The wizard helps you registers a SmartElement.
4. Click Register. If you are using the wizard, follow the instructions on screen. Otherwise, set the following options, which are standard for most SmartElements:

General sets the name, base name (the name from which instances of the SmartElement dragged into the layout will be derived), description, browser compatibility, and functional category. Name and Base Name fields are required. As the Category pop-up menu is empty when you come to this panel, you may create a new category or select an existing one. Browser compatibility defaults to the lowest common denominator for acceptable browsers for this SmartElement type.

Source points to the file containing the SmartElement. Click Browse for a standard file selection dialog box, or enter the path if you know it. Some SmartElement types (DTCs, ActiveX) ask you to select from a list of available com classes instead of defining a source file. The tab for ActiveX also has a field for the element's codebase.

Appearance sets the dimensions, visibility, scalability, icon, and display image of the element when it is placed in the layout view. The icon appears in the Asset Center and the Scripting Tree. The display image is used in the layout view to allow the page designer to get a better feel for what the page will look like without previewing it.

Properties sets the properties of the element. Each instance of an element can be made to behave in a different way by setting its properties. Properties are displayed in their own panel, called the Property tab. While many SmartElement property names are automatically generated, you will probably want to edit their characteristics by selecting them in the list and clicking Edit. You can edit the name of the property as displayed by Drumbeat, the tag by which the property is represented in an HTML document, its description, its type (a wide variety of types are available), and its default value. The type allows Drumbeat to ensure that page authors select the proper sort of value for each property, and in some cases even to provide a more user-friendly method of setting the value (for example, a pop-up color menu). You can also choose where the property is available (in the Scripting Tree, the Attributes tab, or both) and whether the property has a wrapper.

Methods recognizes method names in the JavaScript code as scriptlets are being imported, so there is no need to add them manually. However, you may want to edit a few, to allow them to be hidden from the Scripting Tree or to have wrappers.

Events define which methods or functions the SmartElement will call when certain conditions are met. User actions (for example, a mouse click) are typical conditions that trigger events. For example, clicking a plug-in could cause another element on the page to flash. Events that a SmartElement is capable of capturing are automatically entered into the Events panel, but you can edit them to add pseudo-events.

Publishing requires an understanding of programming in BASIC. Drumbeat takes care of assigning the proper HTML formatting to the elements you place on a page. However, you can also create your own elements and write your own BASIC code to render the HTML tags and JavaScript code. This code does not become a part of your final web site; instead, it is used by Drumbeat to place the element into your HTML documents. You can return the result by assigning it to the function name displayed in the prototype.

Note the following special cases about registering these types of SmartElements:

ASP & COM Objects

ASP and COM SmartElements have abbreviated registration procedures because the files describing these objects are Dynamic Link Libraries (DLLs) that you need to pre-register on your system using either Regsvr32 or MTS (Microsoft Transaction Server). The latter process is preferred because an MTS installation allows for superior caching and process management on the web server. Drumbeat automatically retrieves all of the properties, methods and events from the DLL by a process called introspection. The entire registration process is automatically completed for you. If you want to tweak some of the settings after the introspection process is complete, you can modify the desired changes in this window. Both of these technology types, when applied on a page, appear in the ASP object tree in the Script Center and are available as script objects.

Applets can be either on your local hard drive, or anywhere on the web. Enter the appropriate path or URL. Use the Browse button to choose a local file. If you are specifying a remote component, enter the name of the component's primary class file in the Class field, and the URL to the directory where that class file is stored in the URL field. If the component is in a compressed archive (ZIP or JAR), enter the path to the archive in the Archive field.

ActiveX Control lists only the ActiveX controls that have been registered with Windows (that is, ones that are already installed on your computer because you have visited a page that uses them with Internet Explorer). You should also specify a URL from which the control can be downloaded if the user's computer does not already have it.

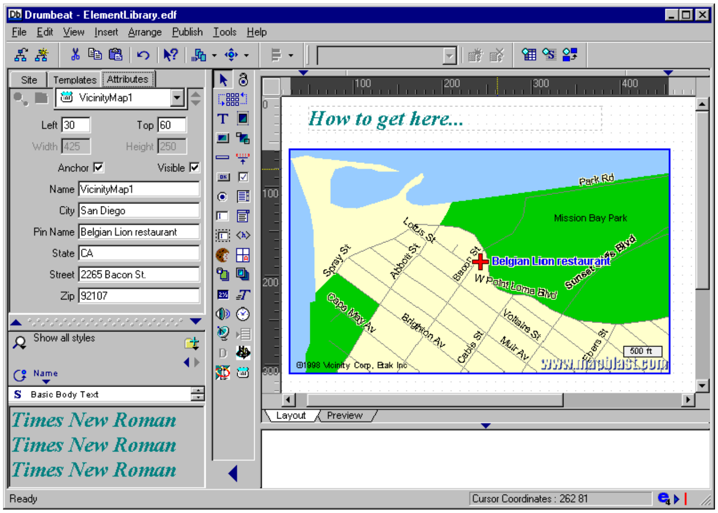
Plug-Ins require the URL where a copy of the necessary plug-in can be downloaded, and how the palette's plug-in should be rendered. Background means that the colors displayed by the plug-in will be forced to fit the palette being displayed by the browser; foreground means that the palette requested by the plug-in will be used instead, forcing all other images on the page to be adjusted to its palette.

Registering Custom Elements

Designing custom elements requires familiarity with the BASIC programming language to write rendering code—that is, a short subroutine in which you tell Drumbeat how to create the HTML code needed to include the SmartElement in your document. Sophisticated knowledge of HTML is also a requirement.

A different type of custom SmartElement is one that uses the resources of an external vendor with a customized query string that needs to be built onto the URL that retrieves the resource. A good example is the Vicinity map, which you will find already in the list of custom SmartElements. The Vicinity Corporation (www.vicinity.com) supplies Web site services such as plug-in interactive maps and driving directions.

The following illustration shows how required properties can be filled in on the Attributes tab of the Vicinity Map SmartElement. Drumbeat then builds the query string in the required format to embed the map on a web page. In this particular example the map image itself has been customized for the sake of clarity—when you use this type of SmartElement normally you will see a generic map in your Drumbeat layout and the actual map of the desired location will be substituted only when the page is published. This lack of immediate feedback is typical of component architecture.


Drumbeat - ElementLibrary.edf

VicinityMap1

Left 30 Top 60
Width 425 Height 250
Anchor ☑ Visible ☑
Name VicinityMap1
City San Diego
Pin Name Belgian Lion restaurant
State CA
Street 2265 Bacon St.
Zip 92107

How to get here...

Ready
Cursor Coordinates : 262 81

You can also use custom elements to support server-side includes and other server-interpreted directives, or custom CGI programs such as hit counters, ad banners, and the like. You might, in the future, even use custom elements to support new HTML features. Drumbeat will not be left behind as the browsers accumulate new HTML tags, since you can support them through your own custom SmartElements.

Creating SmartElement collections

This type of SmartElement is intended to be a repository of interacting SmartElement sets that you have created yourself and that you think you might want to reuse. The menu that pops up when you right-click on any SmartElement or group of SmartElements in layout includes an option Add to Element Library. If you select this option you will be asked to enter a name for the collection, but no other information will be required.

Note: When creating SmartElement Collections, bear in mind that all elements in the interaction set must be selected if the interaction is to be retained.

The power of a SmartElement collection lies in the fact that all attributes and relationships, including activations and interactions, are preserved intact when the collection is applied to a new page. Another common example is sets of pictorial navigation buttons, complete with all their first, previous, next, and last logical relationship to a data table.

You may select a special icon for a new SmartElement collection by selecting it in the Element Library and choosing the Edit function. There is no Registration process for collections.

CHAPTER 12
SmartPages

With browser-sensitive SmartPages, you build adaptive sites that take advantage of the most advanced browser features, while also efficiently supporting older browsers. There is no need to know the idiosyncrasies of each browser or the complex coding techniques necessary to make a cross-browser site work. SmartPages are multiple versions of the same page that are optimized for different browser types. When a browser-specific page is created from a SmartPage, Drumbeat shows which SmartElements and interactions are compatible with the selected browser. Drumbeat automatically generates redirect pages that detect the visitor's browser and redirects to the correct version of the SmartPage.

Creating SmartPages

Each SmartPage is a collection of up to four separate pages which can be designed and otherwise manipulated independently within the layout and in Template Manager, but which are treated as a single page in Site Manager. This enables you to link to SmartPages as you build your site and have the visitor get the correct page version appropriate to their browser when they visit your site. Drumbeat handles all the code for the redirection for you.

When a SmartPage version has not been created for a user's particular browser, they will be directed to the browser at the next level down that is compatible. Conversely, the visitor will be directed upward to the highest level of compatibility should their browser support it. This means, if they are using Internet Explorer 4, they will be directed to the highest level version created for them (preferably an Internet Explorer 4 only page) and not a lower level version, even though the browser may be downwardly compatible with all other versions.

Since SmartPages require extra design effort and can also place additional demands on your web server, you may not want to use them for every page on your site. Instead, set the browser site preference to the most common browser type for your audience, and then create SmartPages for pages with special functionality.

Redirection is done through either client-side or server-side scripting, according to your site preferences. If you choose client-side redirection, visitors may experience a short delay before reaching a SmartPage as the redirection process occurs. To use server-side redirection you need to have ASP or CGI technology available on your server. Drumbeat uses META tag redirection for browsers that do not support JavaScript version 1.1.

To create SmartPages:

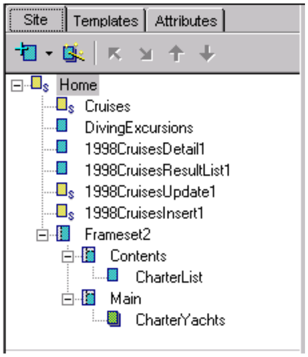
1. Design a page for the optimum browser you want to target.
2. Convert the page to a SmartPage in Site Manager.
3. Create additional versions of the page for the browser types you want to support using the SmartPage toolbar. You have the following options for SmartPage versions:

- Internet Explorer 4.0 is for use in Internet Explorer 4.0 and later. The page can include ActiveX controls, CSS positioning, multimedia effects, scriptlets, design time controls, and data binding.
- Any Version 4.0 is for use in either Internet Explorer 4.0 and later and Netscape Navigator 4.0 and later. The page can include CSS positioning, styles, and JavaScript for both browsers.
- Navigator 3.0 is for use in Netscape Navigator 3.0 or compatible browsers. The page can include DynaImages, and framesets, and JavaScript. Positioning is done with HTML tables.
- Generic includes Internet Explorer 3.0, Navigator 2.0, and some others. The page will also be served to older browsers that do not support JavaScript or Java.

To insert SmartPages:

In Site Manager, right-click any existing page or page set in your site structure and choose Convert to SmartPage. (Framesets and external pages cannot be converted to SmartPages.)

The page will be converted to a SmartPage and will appear in the site structure as a yellow page icon with an S subscripted to it.



Creating and deleting browser-specific pages

Once you have converted a page to a SmartPage, the SmartPage toolbar will be enabled and you can use it to create individual browser-specific pages.

Browser-specific pages are displayed only in Template Manager, where you can assign them to different templates for design purposes, if desired. The SmartPage, which represents the collection of all versions of the page, is displayed only in Site Manager, for site organizational purposes.

To create browser-specific pages:

1. In Site Manager, select the SmartPage.
2. On the SmartPage toolbar, select the browser type for which you want to create a page from the pop-up menu.
3. Click the Create Page button on the SmartPage toolbar.

To delete browser-specific pages:

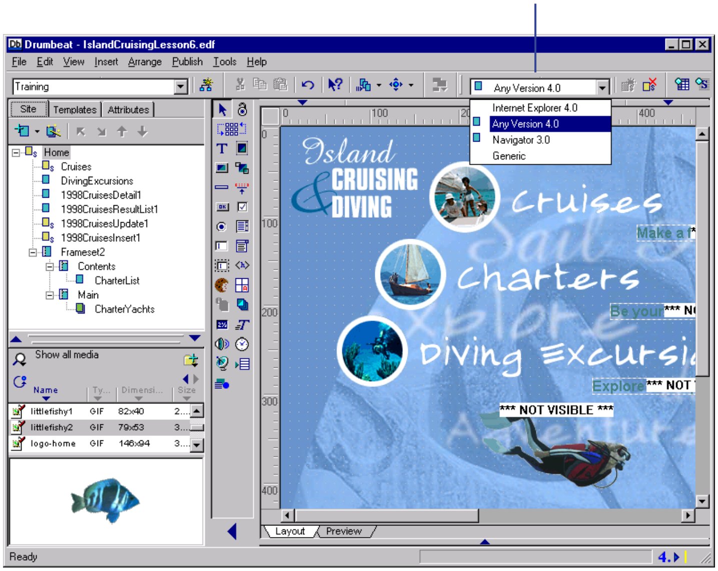
In Template Manager, select the browser-specific page and press the Delete key, or right-click and choose Delete.

Note: Any SmartElement from the original page layout that is not supported by the new browser type appears outlined in red with an X through it.

To viewed browser-specific pages in Site Manager:

Use the pop-up menu on the SmartPage toolbar to choose which version of the page you wish to edit.

SmartPage toolbar



In the pop-up menu, existing browser-specific pages have a blue page (or page set) icon displayed next to them. When you choose Convert to SmartPage, the existing page layout is designated as the browser-specific page for the default browser you have defined for your site (File > Preferences > Site). The pop-up menu initially displays a page icon next to this browser.

Setting SmartPage properties

SmartPage properties are set in the site and template trees located in Site Manager. In the Site Manager tree, the collection of pages is treated as a single page. Therefore, you can only set the shared properties for all the pages that make up the SmartPage. However, in the Template Manager tree, each page in the collection is treated separately so that you can specify the page-specific.

To modify properties in Site Manager:

1. Right-click a SmartPage and choose Properties.
2. Click the General tab to rename the SmartPage.
3. Click the Publish tab to set publishing locations, file options, and file extensions.
4. Click the Meta Text tab to specify Meta tags common to all browser pages here (rather than duplicating this on each browser-specific page).

To modify properties in Template Manager:

1. Right-click a browser-specific page of the SmartPage collection and choose Properties.
2. Click the General tab to rename the SmartPage and view its target-browser setting.
3. Click the Publish tab to set publishing locations, file options, and file extensions. See “Page attributes” on page 39.
4. Click the Channel tab to set channel specifications.

For information on publishing options and using channels, see “Developing channels for sites that support Microsoft Internet Explorer 4.0” on page 193.

Using data binding in SmartPage PageSets

To use data binding, you must create an Internet Explorer 4 SmartPage from a Pageset. (Or you can have Internet Explorer 4 as the default browser in the Site Preferences dialog box.) To learn about PageSets, see “Database-Driven PageSets” on page 271.

To set data binding:

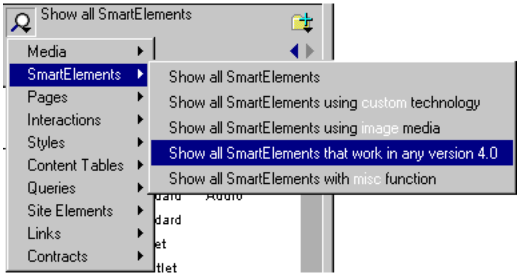
1. In Template Manager, right-click an Internet Explorer 4 SmartPage version of a PageSet and choose Properties.
2. Click the Publish tab and select Use IE 4.0 Data Binding.

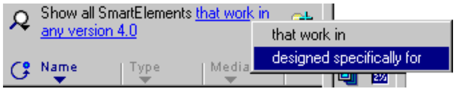
Adding SmartElements and interactions to SmartPages

Once you have added SmartPages, you can then customize them using any SmartElements and interactions. While Drumbeat uses the default layout as a starting point for the new layout, you are free to redesign the page as much as you like. Many SmartElements and interactions work automatically across browser versions. Others may invoke browser-specific functionality.

To display the available browser-specific SmartElements in Asset Center:

Choose the Show all SmartElements by Browser Support query in Asset Center.


Show all SmartElements
Media
SmartElements
Pages
Interactions
Styles
Content Tables
Queries
Site Elements
Links
Contracts
Show all SmartElements
Show all SmartElements using custom technology
Show all SmartElements using image media
Show all SmartElements that work in any version 4.0
Show all SmartElements with misc function


Show all SmartElements that work in any version 4.0
that work in
designed specifically for
Name
Type
Media

Publishing SmartPages

For your visitors to be served the correct SmartPage version, redirection logic must be made available to detect the visitor's browser. Drumbeat enables you to choose among the techniques for doing this (client-side redirection, server-side redirection using ASP, and server-side redirection using CGI). You should set the SmartPage redirection method in the Site Preferences dialog box (File > Preferences > Site).

Redirection method	How it redirects	Advantage	Disadvantage
Client-side redirection	Browser redirection occurs on the client-side, so a redirection page loads first. The JavaScript code determines which browser is being used and directs the browser to load a different version of the page, if appropriate.	It is compatible with all web servers. It supports search engine indexing support as redirect page has Meta tags common to all browser pages and HREF tags to satisfy newer search engine algorithms.	It increases network traffic, since viewing a SmartPage through anything but a generic browser will require the page to be loaded twice.
Server-side redirection using ASP	JavaScript is embedded on a server-side ASP page that gets executed on the server. Then, the appropriate browser-specific page is sent to the visitors browser.	It will load the page only once because the redirect request is executed before the page loads.	Increases network traffic, since browser must load redirect page to determine which browser page to load.
Server-side redirection using CGI	A CGI program on the server is used to detect the browser type and return a redirect instruction to the appropriate page.	This is compatible with all servers and the page loads only once because the redirect request is executed before the page loads.	Requires storage space on the server.

Reverting SmartPages

If you decide you no longer need cross-browser support for different browsers on your site, you can revert a SmartPage to the original page type. When you revert a SmartPage, the current site browser-preference is restored.

To revert a SmartPage to a normal page:

In Site Manager, right-click a SmartPage and choose Revert SmartPage.

When a Smart Page is reverted to a normal page or PageSet, the reverted page matches your current site-browser preference.

If the SmartPage doesn't contain an instance of the current site-browser preference, the page is reverted to the next lower browser preference. If there are no browser pages lower than the current browser preference, the nearest browser page above the current browser setting is returned.

To ensure that a specific browser page is preserved over other browser pages, delete all browser pages except the one to be kept before reverting the SmartPage.

CHAPTER 13
Frames

Framesets provide a way to present multiple pages within a single browser window by dividing the browser window up into separate windows with their own content. Framesets are often used to keep navigational links and banners constant while the main content area changes. They can also be used to provide footnotes or related material while keeping the main content in focus.

Framesets, however, can often be confusing to configure and use for HTML novices. Drumbeat makes creating and using frames easy.

In Drumbeat you can:

- Choose from a number of default configurations encompassing a wide variety of uses.
- Customize a frameset by dragging and dropping frames to the size you want and use frameset attributes to control border colors and widths and margins.
- View all frames in a set in the layout while you work with the design of one page.
- Create your own frameset configurations for others to use.

Creating framesets

Framesets can be created quickly using any of the default configurations available in Drumbeat. You can also create your own custom configurations and add them to Drumbeat. See “Creating your own frameset configuration” on page 268.

Framesets are not supported by generic browsers. However, you can use a noframes redirection for generic browsers if desired. See “Noframes redirection for generic browsers” on page 269.

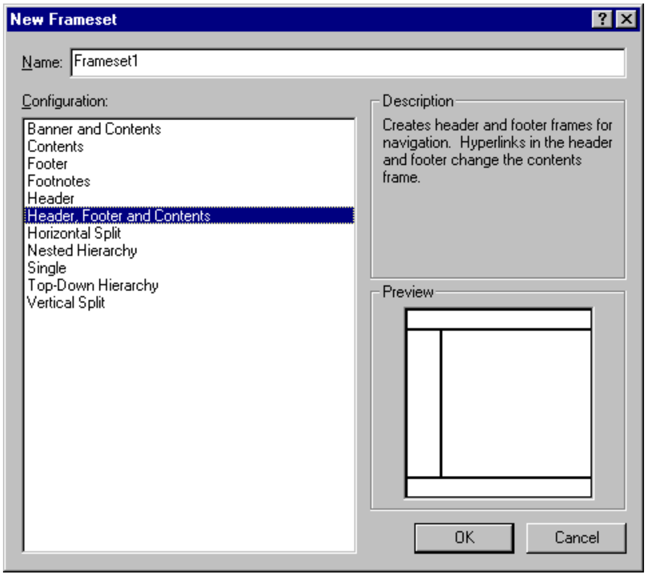
To create a new frameset:

1 Do one of the following:

- From the Insert Page button pop-up menu, select Frameset.
- In Site Manager, right-click on the page where you would like to insert the frameset in the site tree and choose Insert > Frameset. This will nest the frameset under the selected page in the site tree.
- On the main menu, choose Insert > Frameset.

2 In the New Frameset dialog box, enter a name for the frameset in the Name field.

3 Choose a configuration for the frameset. Scroll through the list to see a preview and description of each configuration.



New Frameset

Name: Frameset1

Configuration:

Banner and Contents
Contents
Footer
Footnotes
Header
Header, Footer and Contents
Horizontal Split
Nested Hierarchy
Single
Top-Down Hierarchy
Vertical Split

Description

Creates header and footer frames for navigation. Hyperlinks in the header and footer change the contents frame.

Eleven default frame configurations are available when you create a new frameset.

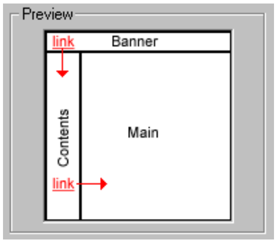
Frameset configurations

Drumbeat offers eleven default configurations for framesets. Additional frameset configurations can be found on the Drumbeat Element Exchange online at http://exchange.drumbeat.com.

The preconfigured framesets in Drumbeat contain default targets for links and are fully customizable.

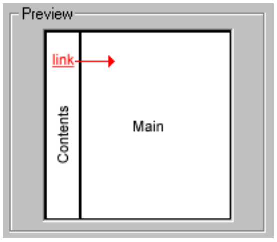
Banner and Contents:

A Banner frame at the top, a Contents frame for navigation on the left, and a Main frame on the right. Default target for links within the Banner frame is the Contents frame. Default target for links within the Contents frame is the Main frame.



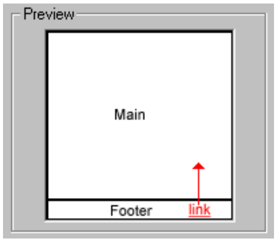
Contents:

A Contents frame on the left for navigation and a Main frame on the right. Default target for links within the Contents frame is the Main frame.



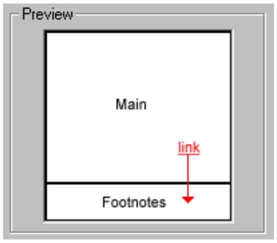
Footer:

A Main frame with a Footer frame for navigation underneath. Default target for links within the Footer frame is the Main frame.



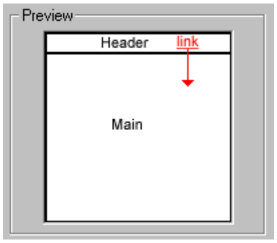
Footnotes:

A Main frame with a Footnotes frame below it. Default target for links within the Main frame is the Footnotes frame.



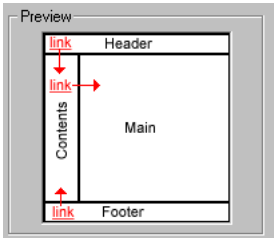
Header:

A Header frame for navigation with a Main frame underneath. Default target for links within the Header frame is the Main frame.



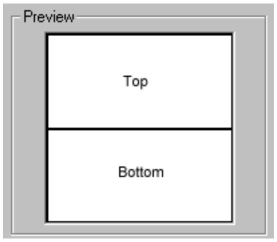
Header, Footer, and Contents:

Header and Footer frames for independent navigation. A Contents frame on the left and a Main frame on the right. Default target for links within the Header and Footer frames is the Contents frame. Default target for the Contents frame is the Main frame.



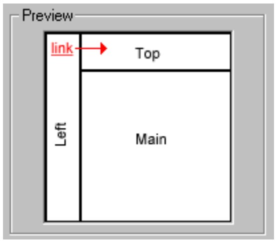
Horizontal Split:

Independent Top and Bottom frames. The Bottom frame is the default main frame.



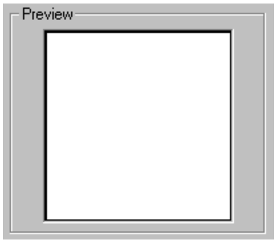
Nested Hierarchy:

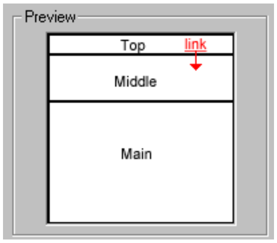
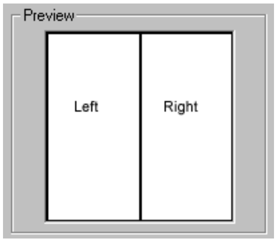
Top frame, Left frame, and Main frame. Default target for links within the Left frame is the Top frame. The Main frame is independent of the other two frames.



Single:

A frameset with a single frame can be used to override any browser default margins, by setting the Border attribute to 0.



Top-Down Hierarchy: Top, Middle and Main frames. Default target for the Top frame is the Middle frame. The Main frame is independent of the other two frames.	
Vertical Split: Two independent Left and Right frames. The right frame is the default main frame.	

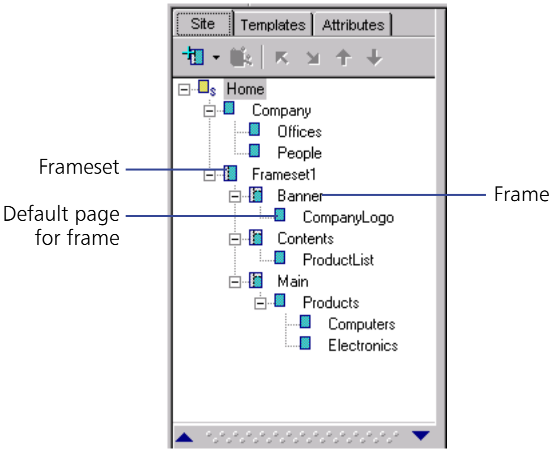
Assigning pages to frames

Once you have created a frameset, you must assign a default page to each frame within the frameset. The default page for a frameset is the initial page that loads into the frame window.

To assign a default page to a frame:

In Site Manager, drag and drop an existing page under the frame, or right-click on a frame and choose Insert > Page to insert a new page as the default page.

The default target for the frame will be the first page underneath the frame. For organizational purposes, you can place all pages that load into this frame under it in the site tree, but only the first one is treated as the default page.



The frameset structure in Site Manager shows the frameset, frames, and default pages assigned to each frame.

Note: The Home page cannot be assigned to a frameset. If you would like the default page in your site to be a frameset, choose the frameset as your Home page in the Publish Settings dialog box. (You can also turn off the publish property of the Home page by unchecking Publish This Page in the Web Page Properties dialog box for the page. See "Page attributes" on page 39.)

For a working example of how to browse a remote URL in a Drumbeat frame, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

Working with frames

Framesets and frames are viewed in the layout just as they will appear in the browser, with all the frames represented in the layout. Only one page is active in the layout at a time; the pages in the other frames will appear with a hatched pattern, similar to inherited elements on templates.

To select an active page of a frameset in layout mode:

Select the page in Site Manager, or if the frames arrangement is already showing in the layout, click the layout of the page you want to work with.

In Site Manager you can view and work with the framesets and frames, as well as the pages that belong to the frameset. To access the Attributes tabs for framesets and frames, you must be in Site Manager. Template Manager displays only the pages that belong to the frames, since the frameset and frame cannot themselves contain design elements.

Modifying framesets

The default frameset configurations can be modified through the Attributes tab for the frameset and the frames.

To view frameset or frame attributes:

Do one of the following:

- Select the frame or frameset in Site Manager, then click the Attributes tab.
- Right-click the layout of any page in the frameset and choose Frameset Attributes or Frame Attributes.

Most attributes for frames are set through the Attributes tab of the individual frames. The frameset has only one attribute, the Border attribute, which sets a border width for all frames in the frameset.

The frame and frameset attributes are described in the following sections.

Frame and frameset attributes

Frames and framesets have their own set of attributes that govern the appearance of the frames. These are described in the following table.

Attribute	Definition
Border Width	Sets the width of borders for all frames in the frameset in the Attributes tab of the frameset. If no borders are desired, this should be set to 0, and Frame Border should be deselected in the Attributes tab for the individual frames.
Border Color	Color of the border between frames (if any). (4.0 browsers and above only.)
Height	Height of the frame in pixels, if vertical. This attribute is enabled only when the Size Type attribute is Pixels.
Width	Width of the frame in pixels, if horizontal. This attribute is enabled only when the Size Type attribute is Pixels.
Percent	Specify the width of the frame as a percentage of the total width of the browser window (enabled only when the Size Type attribute is Percentage).
Margin Height	Controls the top and bottom margins of the frame in pixels.
Margin Width	Controls the left and right margins of the frame in pixels.
Frame Border	Enables or disables the display of a border on the frame. (You can also disable borders for all frames in the frameset by setting Border to 0 in the Attributes tab of the frameset.)
Resizable	Controls whether users can resize the frame in the browser.
Name	The name of the frame (used as a target for links).
Default Target	The default target frame for links created in the frame (can be changed individually).
Default Page	The default page that appears in the frame when the frameset first loads.
Size Type	Determines the sizing method for the frame. Options are Pixels, Percentage or Leftover. Leftover specifies that the frame will use the window area left after the width or height of other frames with specific values are subtracted.
Scrolling	Determines whether scrollbars are included for the frame. The options are Yes, No, and Auto (scrollbars provided only if necessary).

Frame borders

The border width is set through the Attributes tab of the frameset and applies to all frames in the set. The presence of the border between frames is controlled through the Attributes tab of the individual frames.

To turn off frame borders on all frames in the frameset:

1 Do one of the following:

- Select the frameset in Site Manager, then click the Attributes tab.
- Right-click the layout of any page in the frameset and choose Frameset Attributes.

2 In the Attributes tab, for Border enter 0.

3 Do one of the following:

- Select a frame in Site Manager and click the Attributes tab.
- Right-click the layout of the page in the frameset and choose Frame Attributes.

4 In the Attributes tab for the frame, deselect Frame Border. Repeat for each frame in the frameset.

The border change is not visible in the layout design, but it will show up when you publish.

To turn off borders on individual frames:

1 Do one of the following:

- Select the frame in Site Manager, then click the Attributes tab.
- Right-click the layout of the default page in the frame and choose Frame Attributes.

2 In the Attributes tab, deselect Frame Border.

You can also set width and color for frame borders (in 4.0 browsers and later).

To set the width of a frame border:

1 Do one of the following:

- Select the frameset in Site Manager, then click the Attributes tab.
- Right-click the layout of any page in the frameset and choose Frameset Attributes.

2 In the Attributes tab, for Border enter the desired width as a numerical value in pixels.

To set the border color on a frame:

1 Do one of the following:

- Select the frame in Site Manager, then click the Attributes tab.
- Right-click the layout of the default page in the frame and choose Frame Attributes.

2 In the Attributes tab, click the color bar next to Border Color, then choose a color from the Color palette, or click Define Custom Color to select a color from the color spectrum or by entering RGB values.

Sizing frames

The size of frames within the frame configuration can be changed as desired, using the Attributes tab for each frame.

Frames can be sized using any of three methods:

- Percentage: A percentage of the total screen width or height.
- Pixels: An absolute value in pixels.
- Leftover: The “leftover” area of the total screen width or height after the absolute value applied to another frame or frames in the set is subtracted.

To choose the sizing method for a frame:

1 Do one of the following:

- Select the frame in Site Manager, then click the Attributes tab.
- Right-click the layout of the default page in the frame and choose Frame Attributes.

2 In the Attributes tab, select the Size Type from the pop-up menu.

To change the size of a frame in pixels:

1 Do one of the following:

- Select the frame in Site Manager, then click the Attributes tab.
- Right-click the layout of the default page in the frame and choose Frame Attributes.

2 Do one of the following:

- In the Attributes tab, with the Size Type selected as Pixels, enter the pixel value you want for the width (for a vertical frame) or height for a horizontal frame).
- Drag and drop the frame border on the layout to the desired width, using the ruler as a guide. (As you do this, the corresponding value changes in the Attributes tab.)

To change the size of a frame as a percentage of the browser window:

1 Do one of the following:

- Select the frame in Site Manager, then click the Attributes tab.
- Right-click the layout of the default page in the frame and choose Frame Attributes.

2 Select the percentage for Size Type, then enter a percentage value.

Frames are normally resizable by the user. This means users can drag the frame border in the browser to make the frame larger or smaller. If you want to prevent the user from resizing frames, you can disable resizing.

To prevent users from resizing a frame:

1 Do one of the following:

- Select the frame in Site Manager, then click the Attributes tab.
- Right-click the layout of the default page in the frame and choose Frame Attributes.

2 In the Attributes tab, deselect Resizable.

Frame scrolling

By default, each vertical frame in a frameset has a separate scrollbar for scrolling contents individually. You can prevent scrollbars from appearing on a frame to save screen real estate and hide the frame divisions. You should do this only if the content of the frame page is certain to fit within the default browser screen.

To turn off scrolling on a frame:

1 Do one of the following:

2 Select the frame in Site Manager, then click the Attributes tab.

3 Right-click the layout of the default page in the frame and choose Frame Attributes.

4 From the Scrolling pop-up menu, choose No.

If you are uncertain whether the contents of the frame page will fit in the default browser window, particularly at different screen resolutions or font sizes, select Auto and scrollbars will be provided only if necessary when the contents exceed the given window.

Targeting links in frames

Since all links open by default in the same window as the link, it is important that all links created within frames contain the appropriate target if you want the new page to load in a different frame. The default frame configurations in Drumbeat include default targets for links appropriate for their hypothetical uses. However, the default target for links within a frame can be changed as appropriate.

To change the default target for links within a frame:

1 Do one of the following:

- Select the frame in Site Manager, then click the Attributes tab.
- Right-click the layout of the default page in the frame and choose Frame Attributes.

2 In the Attributes tab, select a default target from the pop-up menu.

Target options include:

- An existing frame within the frameset: Frames within the current frameset are listed by name in the Target pop-up menu in the Link dialog box when linking from a frame.
- None: Links will open within the current frame.
- Full Window: Links will open in the full browser window, and no frameset will be active.
- New Window: Links will open in a new browser window, leaving the current browser window intact.
- Parent Frame: Links will open in the area that the current frameset occupies on the screen. The current frameset will be replaced, but any higher level framesets in a nested scheme will still be active.

You can change the target for individual links whenever the default target is not appropriate.

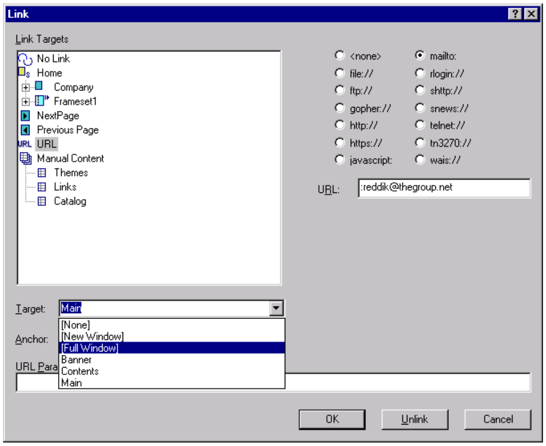
To change the target for an individual link within a frame:

1 Select the element that forms the link and click the Attributes tab, or right-click and choose Attributes.

2 In the Attributes tab, click Link Assign. In the Link dialog box, select the appropriate target from the Target pop-up menu.

To change the target for a link that is a text string within a text element:

1. Select the text and click the Link button on the Text Formatting toolbar.
2. In the Link dialog box, select the appropriate target from the Target pop-up menu.



● mailto:

URL: :reddik@thegroup.net

Target: Main

Change the target for a specific link within a frame in the Link dialog box.

Importing framesets

If you choose to import a site that contains frames, Drumbeat Site Import will not import the frameset structure itself. A blank page will be created to represent each frameset. All pages that are assigned to framesets are imported.

Once the import process is complete, you can quickly create framesets and reassign pages to the frames to reproduce the original structure. See “Creating framesets” on page 256 and “Assigning pages to frames” on page 260.

For more information on importing sites, see “Importing pages and sites” on page 45.

Creating your own frameset configuration

Frameset configurations can be modified and created at the code level, if you're an advanced user. Existing frameset configuration files can be found in the Framesets folder of the Drumbeat application folder. A typical frameset configuration file looks like the one below. This is the INF file for the Banner and Contents configuration:

```
[info]
title=Banner and Contents
description=Creates a banner frame at the top, with a contents and main frame. Hyperlinks
in the banner change the contents frame.
noframesURL=
layout=[R(64,*)F("Banner",[C(150,*)F("Contents","Main")])]
```

Create your own configuration by following the conventions of the info file above where:

- R = rows
- F = frame
- C = column

To create your own frameset configuration:

1 In Notepad or any text editor, create your info file following the INF file conventions.

2 Save the file in the Framesets folder of the Drumbeat application folder as a text file with the .inf extension.

The new configuration will show up in the default configuration list, with the title provided in the INF file, the next time you choose to insert a frameset.

For example, you can create a four-part division of the screen, similar to the Banner and Contents configuration but with an added division of the banner frame. This is the code that will create the new configuration file:

```
[info]
title=Quadruple
description=Creates two contents frames on the left and a header frame above a main
frame on the right.
noframesURL=
layout=[C(75,*)F([R(150,*)F("LTop","LBottom")],[R(150,*)F("RTop","RBottom")])]
```

Save the file to the Framesets folder of the Drumbeat application folder and the new configuration will show as a choice in the Frameset window, including a preview pane.

Note: You can copy an existing INF frameset configuration file and customize it to suit your needs.

Noframes redirection for generic browsers

Generic browsers do not support frames. If you create SmartPages for generic browsers in a site with frames, you must create separate links to pages outside of the frameset. You can create direct links to the pages within your frames, instead of to the frameset. Drumbeat does not support the NOFRAMES tag in HTML.

14

CHAPTER 14
Database-Driven PageSets

Not too long ago, web sites with databases were rare species. Now, a web site without a database is almost a dinosaur. The ease and convenience of creating large amounts of content and updating it all rapidly with a database back end presents an overwhelming advantage to the web developer.

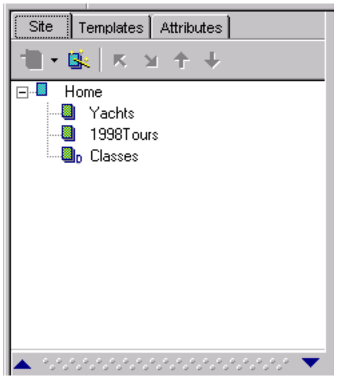
Drumbeat accommodates the savvy web developer who wants to use database-driven content on a web site in two ways:

- PageSets: These are a series of pages that use the same layout and style but present different data on each page. PageSets can be published as standard HTML pages, so they do not require ASP support and will work on any server.
- ActiveServer Pages: These are fully interactive, database-driven pages that include a wide range of server-side functions. ASP server support is required.

PageSets make it easy for anyone to generate large amounts of content and disseminate data from a spreadsheet, database or custom file, without the need for a database server or any special tools. However, you can also use PageSets with ASP server support for dynamic, server-side pages created on-the-fly. This chapter shows you how easy it is to create PageSets whose content comes from database sources and customize them for web use. Chapters 15-18 discuss Active Server Pages in detail.

Creating and managing PageSets

A PageSet is a series of pages that share a common layout but have different content. When you publish a PageSet, the content of the content table is used to generate as many pages as there are rows in the content table. Since a PageSet is treated as one page for structural and design purposes, PageSets are very easy to manage in the Site Manager or Template Manager. In both views, PageSets are represented as one page, even though they may represent several, or even dozens of, different pages.



Three PageSets in Site Manager. Each of these PageSets represents dozens of pages. The D next to the PageSet named Classes indicates that it is database-driven (an ODBC PageSet).

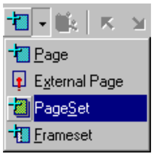
Creating PageSets

To create a PageSet you must first have a content table on which to base the PageSet. Content tables for PageSets can be of several types, each with its own uses. The publishing options differ according to the type of content table you choose. These options are covered in the next section, Types of PageSets and Publishing Options.

Once you have the content table you want to use, the process of creating the PageSet is identical.

To create a PageSet:

1 Click the Insert Page button pop-up menu and choose PageSet, or choose Insert > PageSet.

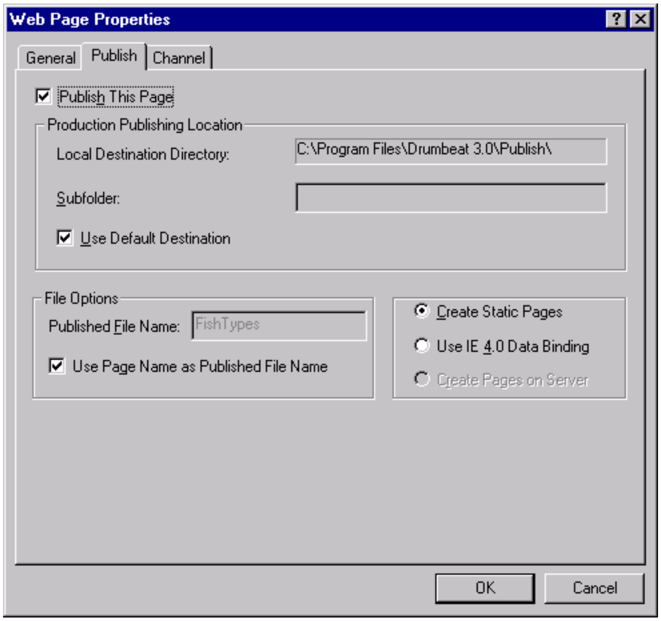


Insert a new PageSet.

2 In the Web Page Properties dialog box, give the PageSet a name and choose a content table for the content from the pop-up menu. Then click OK.

Note: Drumbeat does not allow you to give the PageSet the same name as the content table.

If you intend to use data binding or ASP to publish your site, set this up in the publish tab of the web page properties dialog box. see the using data binding section in this chapter or, for ASP publishing, see “PageSets with ASP server support” on page 279.



☑ Publish This Page

Production Publishing Location

Local Destination Directory: C:\Program Files\Drumbeat 3.0\Publish\

Subfolder:

☑ Use Default Destination

File Options

Published File Name: FishTypes

☑ Use Page Name as Published File Name

● Create Static Pages

○ Use IE 4.0 Data Binding

○ Create Pages on Server

Set up Publishing Options for the PageSet in the Publish tab.

Types of PageSets and publishing options

Data for a PageSet comes from a content table, which can be of any type. However, the type of content table you choose will affect the way you can publish and maintain the PageSet.

PageSets can be basically static as a series of HTML files or they can be fully dynamic when published as Active Server Pages. The advantage of using static PageSets over Active Server Pages is that the pages are standard HTML pages that can be served up by any web server and made to target any browser. Static PageSets are useful for content that does not change or changes infrequently and to which you may want to add new content without altering the existing file. The disadvantage is that static PageSets are not instantly updateable when the database changes. However, if you base your static PageSet on an ODBC content table, you can simply republish when you add, delete, or make changes to the data in the database or CSV file.

PageSet content

The content for a PageSet comes from a Drumbeat content table, which may be of three types:

- Manual content table: A custom file created by the developer that can contain any site assets.
- CSV content table: Generated from a CSV (comma-separated values) file. This creates a static table that does not further reference the original file. You may edit or add new content components without affecting the original CSV file.
- ODBC content table: May be created from either a CSV file or a relational database file, referenced by DSN.

For more information on the different types of content tables and their uses, see “Content Tables” on page 199.

The type of content table you use to generate your PageSet determines the updateability and other features of the PageSet.

Types of PageSets and the publishing options.

PageSet Type	Server Support	Page Output / Browser Support	Content Table Update	Publish Setup
Static PageSets				
Static PageSet with CSV content table or manual content table	None	HTML All browsers	Republish pages whenever content of content table in Drumbeat is changed	Choose Create Static Pages in Publish tab of Web Page Properties dialog box
Static PageSet with ODBC content table	None	HTML All browsers	Republish pages to update automatically from local database	Choose Create Static Pages in Publish tab of Web Page Properties dialog box
Server-Side Dynamic PageSets				
Dynamic (server-side) PageSet (ASP) with ODBC content table	ASP	ASP All browsers	Automatic updates with on-the-fly page creation on server (requires DSN to be set up on server)	Check Create Pages on Server in Publish tab of Web Page Properties dialog box
PageSets with Data Binding (Client-Side Dynamic)				
PageSet with data binding & CSV content table or manual content table	None	HTML with CSV file on server (published by Drumbeat) IE 4 only	Republish pages whenever content of content table in Drumbeat is changed OR Replace the CSV file on server to update page content on the server	Use IE 4.0 data binding in Publish tab of Web Page Properties dialog box
PageSet with ODBC content table and data binding	None	HTML with CSV file on server (published by Drumbeat) IE 4 only	Republish pages to update automatically from local database OR Replace the CSV file on the server to update page content on the server	Choose Use IE 4.0 data binding in Publish tab of Web Page Properties dialog box

Static PageSets with manual or CSV content tables

To create a PageSet whose content comes from a manual content table, you must first create the manual content table and populate it with the assets you want to use in your pages. For information about creating and populating manual content tables, see “Manual content tables” on page 201.

CSV files are a popular and convenient source of content for PageSets, since the basic content can be generated from an existing CSV file, which can then be modified as you please. CSV content tables can be generated easily from a spreadsheet or database file.

You can create a CSV file from:

- An Excel spreadsheet (or other spreadsheet program)
- A table, query, or view created in a database management program

Check the documentation in your application for the procedures to use to export a file in CSV format.

To create a PageSet whose content comes from a static CSV file, you must first create a CSV content table.

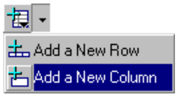
To create a content table based on a CSV file:

1. Open the Content Center, and click the Create Content Table button.
2. In the New Content Table dialog box, select Populate From External File. Then, browse to select a CSV file.
3. Select Use First Line for Column Names, if you would like the field names in the file to be used as the content table column header names. Then click OK to generate the content table.

Once you have generated a CSV content table, you can add additional columns and rows, add new content to cells, or change the existing content if you want before generating a PageSet. For example, if you want to include images for items in the content table, you can add a column to the table and place image files for each item in the appropriate cells of the content table. When the PageSet is generated, the images will be included.

To add media to a CSV content table:

1 Click the Add Row/Column pop-up menu and choose Add a New Column.



2 Drag and drop image files from the Asset Center into the appropriate content table cells.

Note: Changes you make to the CSV content table in Drumbeat are not reflected back in the original CSV file. Likewise, any changes made to the original CSV file will not be reflected in the Drumbeat CSV content table unless you re-import the file.

PageSets with ODBC content tables

PageSets based on ODBC content tables are the most dynamic PageSet type and are the easiest to maintain as they can be easily updated as the database changes. To create an ODBC content table, you must create a Data Source Name (DSN) for your data file. Create a DSN for a CSV file the same as you would for an ODBC database.

Creating a DSN is a step that begins outside of Drumbeat in the ODBC Data Source Administrator. For more information about setting up a DSN for ODBC databases, see “Configuring Drumbeat Publish Settings for ASP” on page 319. The following instructions explain how to create a DSN specifically for a CSV file.

To create a DSN for a CSV file:

1 Open ODBC Data Source Administrator from the Windows Control Panel, or in the Drumbeat SQL Query wizard, click Data Sources.

2 Select the System DSN tab, and then click Add.

3 In the Create New Data Source dialog box, choose Microsoft Text Driver and click Finish.

4 In the ODBC Text Setup dialog box, enter a data source name for your file and a description (optional). Deselect Use Current Directory and click Select Directory.

5 Browse to locate the directory that contains your CSV file. (You will see the file in a grayed-out list, but you will not be able to select the file, just the directory.) Click OK, then click OK again and close the dialog box.

Note: When publishing to a remote server, you need to be sure that an identical DSN is set up on both the local and the remote server. You will need to transfer the file manually to the server, if it is not already referenced by the server through your network.

Next you need to create a query to the CSV file, using Query Manager.

To create a query to a CSV file:

1 In Drumbeat, open the Query Manager (click the Add Media button and choose Queries from the pop-up menu). In Query Manager, click New Query.

2 In the SQL Query window, choose the data source name from the pop-up menu. In the Tables list, select the file.

3 Do one of the following:

- In the SQL Statement window, enter SELECT * FROM Filename
- Use the SQL wizard to create the SELECT ALL statement for you.

4 Click Test to preview the content. Give the query a name and click Save Query.

5 In Query Manager, click Create Table.

Once you have created an ODBC content table from your CSV file, it will automatically be refreshed from the current CSV file each time you open the site file in Drumbeat. Or you can easily refresh the content table while you are working in Drumbeat.

To update an ODBC content table:

1 Do one of the following:

- Close and reopen the site file in Drumbeat.
- In the Content Center, select the content table, and then click the Edit Content Table button.

2 In the ODBC Content Table dialog box, click OK.

You will have to republish your PageSet to reflect the changes (unless you are using ASP server support).

PageSets with ASP server support

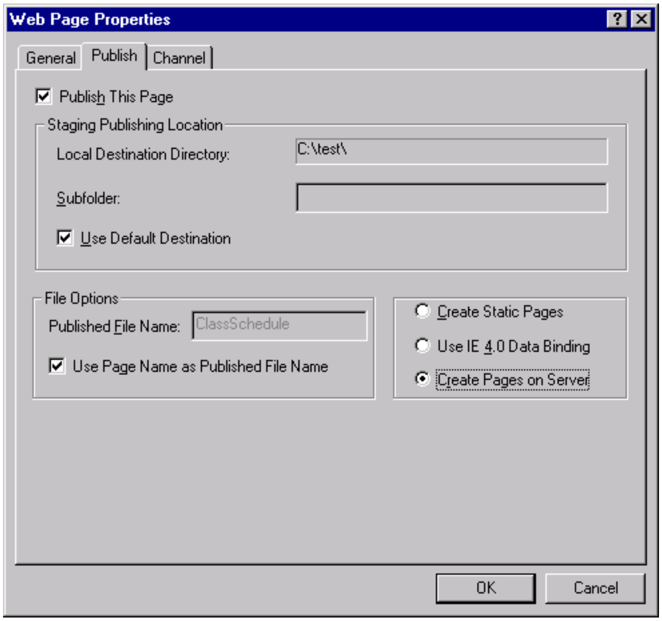
To set up ASP publishing for a PageSet:

1 Make sure your Site Preferences are set for ASP server support. Check that ASP Server Support is selected in File > Preferences > Site.

2 Create a new PageSet based on the ODBC content table. In the General tab of the Web Page Properties dialog box, select the ODBC content table from the pop-up menu.

Note: If you have already created a PageSet with a CSV content table, you can change the content table used in the PageSet to the new ODBC content table after you create a DSN for your CSV file. You do not need to regenerate the PageSet. Right-click on the PageSet in the Site Manager site tree and choose Properties. In the General tab, choose the new content table from the pop-up menu.

3 Click the Publish tab. Select Create Pages on Server.



☑ Publish This Page

Staging Publishing Location

Local Destination Directory: C:\test\

Subfolder:

☑ Use Default Destination

File Options

Published File Name: ClassSchedule

☑ Use Page Name as Published File Name

○ Create Static Pages

○ Use IE 4.0 Data Binding

● Create Pages on Server

Choose Create Pages on Server for PageSets using WebShpere supportASP.

Note: The option Create Pages on Server will be grayed out unless you have chosen ASP server support and the content table on which the PageSet is based is referenced by DSN.

Whenever you open your Drumbeat site file, the content table will be refreshed from the current CSV file. You can also refresh your content table from within Drumbeat if changes occur while you are working with the file or if you are uncertain if it is current. (See “PageSets with ODBC content tables” on page 277.)

Using data binding

Data binding can be used with either static PageSets or PageSets from ODBC content tables. Data binding is specific to Internet Explorer 4.0 and is not supported by Netscape or generic browsers.

To use data binding with PageSets:

1. Make sure the target browser is Internet Explorer 4.0 (choose File > Preferences > Site to set).
2. In Site Manager, select the PageSet. Right-click and choose Properties.
3. In the Publish tab, select Use IE 4.0 Data Binding.

   Note: The Use IE 4.0 Data Binding option will be grayed out for sites or pages targeting browsers other than IE 4.0.

When you use data binding, Drumbeat creates the CSV file on the server at publish time. This file is downloaded by the client and used to create pages on-the-fly in the browser, without further access to the server.

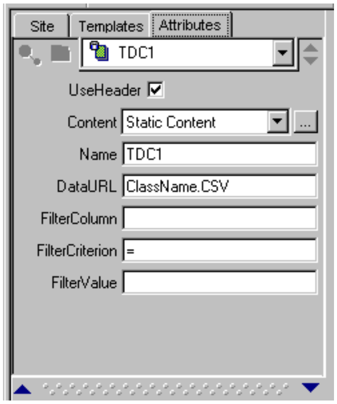
If you are using a static CSV file, the CSV file on the server is created once and will not automatically update. However, you can manually edit the CSV file on the server, if you wish.

If you are using an ODBC table for the PageSet (a relational database file or a CSV file referenced by DSN) Drumbeat will automatically update the CSV file on the server when you republish. However, you can manually update the CSV file, if you wish, avoiding the need to republish.

The Tabular Data Component (TDC)

The Tabular Data Component (TDC) appears in the Basement of any page that uses data binding. The TDC is a Microsoft ActiveX control that can be used to display data stored in a delimited text file. Using the TDC you can display data either within tables or within controls found in a form. Data from the CSV file can be bound to form elements or displayed in HTML table format, where the content of the table is repeated for each record of the data set.

The attributes of the TDC element can be seen in the Attributes tab for the element.



UseHeader ☑

Content: Static Content

Name: TDC1

DataURL: ClassName.CSV

FilterColumn:

FilterCriterion: =

FilterValue:

The attributes of the TDC element can be set from its Attributes tab.

The attributes of the data binding element include:

- UseHeader: Specifies whether the first line of the data file in the published CSV will contain the column heading names of the content table.
- Content: The content table used as the basis of the PageSet.
- DataURL: The path to the data file.
- FilterColumn: Column on which to set up a filter, if desired.
- FilterCriterion: The filter parameter (=, = >, < =, etc.)
- FilterValue: Value of the filter column.

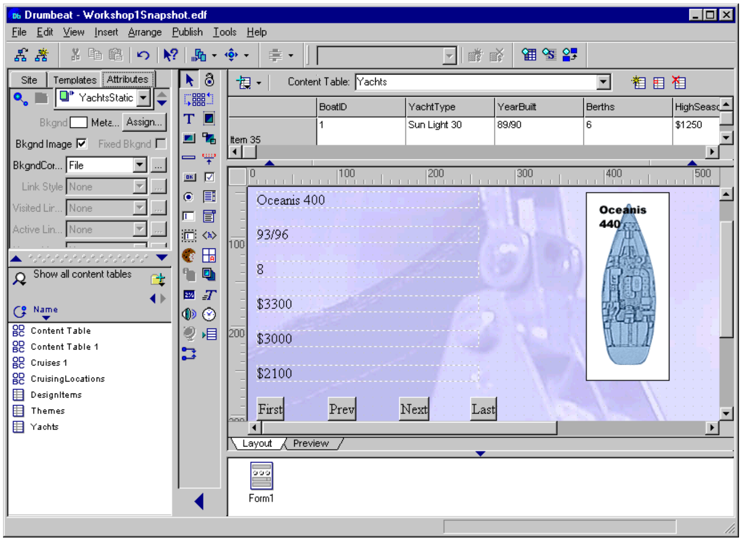
In the Content dialog box, you can choose the columns in the file that you want to display.

To choose the columns to display:

1. In the Attributes tab for the TDC, click the Assign Content button (the ellipses button next to Content).
2. In the Content dialog box, select the columns in the right pane (Show these columns) that you don't want to display, and click Remove.

Using PageSets

Drumbeat creates a default layout for the PageSet, importing elements from the content table and placing them on the page as shown in the following figure. Drumbeat also adds four navigation buttons (First, Last, Next, and Previous) to the layout for browsing through records. Once the PageSet is created you can add design elements and other elements and work with it like any page or template.



Content Table: Yachts

BoatID	YachtType	YearBuilt	Berths	HighSeas
1	Sun Light 30	89/90	6	$1250

Bkgnd Image ☑ Fixed Bkgnd □
BkgndCol... File
Link Style None
Visited Lin... None
Active Lin... None

The initial output of a static PageSet generated by Drumbeat.

The content of the first row of the content table is displayed in layout mode for design purposes. Drumbeat creates additional pages for the rest of the rows when you publish the site.

Designing with PageSets

Use the layout tools to move the dynamic elements where you want on the page. If you want labels on the fields, add text boxes and type in your labels. Add any additional elements that will be the same on all of your pages, such as titles, headers, footers, etc.

The following are some of the design tasks you may want to do to your PageSets:

- Add labels to the generated fields.
- Apply styles to the text.
- Use SmartSpacer to allow for variable-length fields.
- Replace the navigation buttons with custom image buttons.

To apply styles quickly to the text fields:

1. Selecting the text elements on the layout that you wish to change.
2. In Asset Center, choose Styles > Show All Styles.
3. Browse the styles to find the one you want to apply. Drag the style from the Asset Center onto any one of the selected elements in the layout.

You can also assign styles to elements individually through the Attributes tab for each text SmartElement.

If you are designing for 4.0 browsers or later, you can use SmartSpacer to allow for variable-size fields. Older browsers do not require SmartSpacer, as they will automatically reflow the page when the content changes. (The introduction of Z-order with CSS requires the SmartSpacer logic.)

To allow for variable-size fields:

Drag the SmartSpacer element from the toolbar onto your page and drop it at any point within the horizontal area of the display of the field.

You can replace the page navigation buttons at the bottom of the page with image buttons of your own. The method for doing this depends on whether you have a static PageSet or an ODBC PageSet.

To replace the auto-generated page navigation buttons:

1 Select the navigation buttons (First, Previous, Next and Last) on the layout and delete them.

2 Drag the Image Button SmartElement from the SmartElement toolbar to the layout.

3 Drag an image from the Asset Center or from a content table cell onto the image button on the layout. (To use multiple rollover image buttons, you can drag a second image onto the first. For more information on this, see “Creating a rollover image button” on page 165.)

4 Right-click the image button and choose Possible Activations.

5 In the Interactions Center, find the Activation in the PageSet Static category that says When [ImageButton] Is Clicked Go to the Next Page in Pageset. Double-click the activation or right-click and choose Apply.

Replace image buttons for the First, Previous, and Last buttons in the same manner, assigning the appropriate interactions (go to previous page, etc.) to each.

To replace the auto-generated page navigation buttons on an ODBC PageSet:

1 Replace the buttons in the same manner as for a static PageSet above.

2 Select the Next button and the recordset in the Basement. Right-click and choose Possible Interactions.

3 Locate and apply the following interaction in the Database Navigation category: Go to the Next Record in [Recordset] When [ImageButton] Is Clicked. (If this is a different navigation button, use the interaction appropriate to the button.)

4 Drop a Hidden Form element in the Basement. (Both the Form element and a Hidden Form element will appear.)

5 Select the button and the Form element in the Basement. Right-click and choose Possible Activations. Double-click the following activation: Submit form When [ImageButton] Is Clicked. In the Parameters dialog box, click OK to accept the default form action.

PageSet attributes

A PageSet has the same attributes as a page or template, since its output is simply a series of pages. The PageSet layout can be considered similar to a template, with all elements that are applied to the PageSet appearing on all pages that are part of the set. PageSets can also be assigned to templates and can inherit elements through the template structure in Template Manager.

PageSet attributes include:

- Background image: An image file used as the page background.
- Background color: Page background color.
- Fixed background: Determines whether a background image is fixed (nonscrolling) or scrolls with the page.
- Link Style: Includes font, font size, and color of text links and border, if any, around links (anchor elements).
- Visited Link Style: Includes font, font size, and color of text links and border, if any, around visited link elements.
- Active Link Style: Includes font, font size, and color of text links and border, if any, around active link elements (as it is clicked).
- Hover Link Style: Includes font, font size, and color of text links and border, if any, around link elements when the mouse is over (Microsoft Internet Explorer 4.0 only).

To view and change the attributes of a PageSet:

1. Select the page in Site Manager or Template Manager so it is active in the layout. (If it is already selected, click an empty place in the layout to deselect any currently selected elements.)
2. Click the Attributes tab or right-click on an empty spot in the layout and choose Attributes.

Assigned attributes are displayed in the Attributes tab. Attributes that are inherited from a template are grayed out. To change any attribute, you must first cancel any inheritance from higher templates. For more information on setting and changing page attributes, see “Page attributes” on page 39.

Linking to PageSets

Links can be created in various ways within PageSets, to the first page or last page in the PageSet or to individual pages within the PageSet.

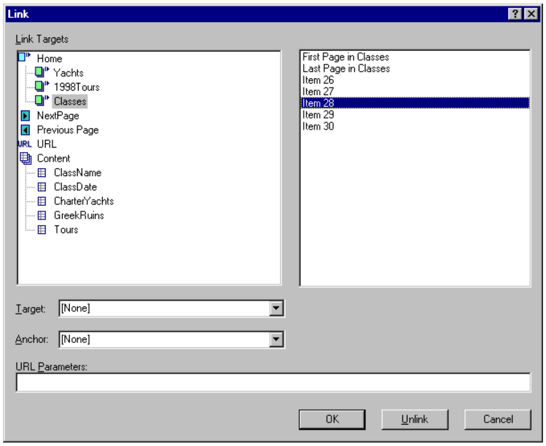
To create a link to an individual page in a PageSet:

1 Select the element that forms the link anchor.

2 Do one of the following:

- In the Attributes tab for the element, click Link Assign.
- If the link anchor is a text string within a text element, click the Hot Link button on the Text Formatting toolbar.

3 In the Link dialog box, select the PageSet. A list of items in the PageSet will appear in the right pane. Select the item to which you want to create the link and click OK.



Target: [None]

Anchor: [None]

URL Parameters:

Create a link to the first or last page in a PageSet or to any record within the PageSet.

The links that can be made to and within PageSets depend on the published type (see “Types of PageSets and publishing options” on page 274).

Types of links that can be created to PageSets

PageSet Type	First Page	Last Page	Next Page	Previous Page	None	Row Number
Static to Itself	Yes	Yes	Yes	Yes	N/A	N/A
Regular Page to Static	Yes	Yes	No	No	N/A	N/A
Server-side to Itself	Yes	Yes	Yes	Yes	Yes	N/A
Regular Page to Server-side	Yes	Yes	No	No	Yes	N/A
Databound to Itself	Yes	Yes	Yes	Yes	N/A	Yes
Regular Page to Databound	Yes	Yes	No	No	N/A	Yes

CHAPTER 15
Accessing Databases and Creating Queries

Active Server Pages allow you to share live, up-to-date database information company-wide over an intranet or provide interactive information to public users over the Internet. With Drumbeat you can create Active Server Pages with any ODBC–compatible database.

Setting up database access

To use your ODBC database in Drumbeat, you must first set up access to the data source by creating a data source name (DSN) for it, if this has not already been done on your system. If you are publishing to a remote server, you must create identical data source names on both the development and Web servers.

Creating a DSN is similar to creating a virtual directory (see "Creating a Web directory" on page 190), and allows you to access the database by name from anywhere in your system or network. You can create a DSN for your database from the ODBC Data Source Administrator in Windows.

To open the ODBC Data Source Administrator:

Do one of the following:

- From the Windows Start menu, choose Settings > Control Panel > 32-bit ODBC.
- Within the Drumbeat SQL Query window, click Data Sources.

As indicated by the tabs on the ODBC Data Source Administrator, there are three types of DSNs you can set up:

- A user DSN is available only to a user on a computer and is stored in the user's personal settings.
- A system DSN is available to anyone who uses the computer where the DSN is defined.
- A file DSN is created as an independent file that can be easily moved from one system to the next.

If you set up a file DSN correctly you may be able to use it to set up the DSN on the Web server itself, so that it doesn't have to be recreated on the server. Check with your Web server administrator to see if this is possible on your system. For most purposes, however, a system DSN is adequate.

To set up a system DSN for your database within the ODBC Data Source Administrator:

1. Click the system DSN tab. Installed databases will be listed in the System Data Sources window. To create a new DSN, click Add.
2. In the Create New Data Source dialog box, click the appropriate driver for your database (for example, Microsoft Access Driver, Microsoft FoxPro Driver, SQL Server, and so on.). Click Finish.
3. In the ODBC Microsoft Setup dialog box, type a name for your database in the data source name text box, and a description (optional). Click Select. Browse to select the database, and then click OK.
4. Click OK again to return to the System DSN dialog box where you will see your new data source listed. Click OK. (If you have accessed the ODBC Data Source Administrator from within Drumbeat, you will be returned to the SQL Query window.)

If you are publishing to a remote server, you need to create identical data source names (DSNs) on both the server and development computer. If the database resides on your company intranet, your systems administrator may already have configured a DSN on the server. You will need to check with your systems administrator for the correct DSN to configure on your own system.

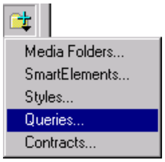
Note: To properly set up a DSN and access your database in Drumbeat you need to have ODBC database drivers version 3.51 or later. Drumbeat includes these ODBC database drivers in its installation program and automatically installs them for you if you have older drivers. This may create a conflict with some older programs that require old driver versions. See the Installation Notes for more information if you suspect conflicts with your database drivers.

Creating database queries

ODBC databases are accessed in Drumbeat through SQL queries you create in Drumbeat. You can use SQL queries that already exist as part of your database management system, or create new ones in Drumbeat. Queries can be both created and modified through the Query Manager.

To open Query Manager:

In the Asset Center, click the Add Assets button and choose Queries.



You can also access the Query Manager when creating a new content table that you want to come from an ODBC data source.

To open the Query Manager from the Content Center:

1 In the Content Center, click the Create New Content Table button.

2 In the New Content Table dialog box, select Populate from ODBC Source, and then click Select to select a data source.

Query Manager

The Query Manager displays all the queries currently available to the site. Queries that have been created specifically for this site, as well as any global queries (queries available to all sites), are listed by name, DSN, and modification date.

To create a new query:

In Query Manager, click New.

To edit an existing query:

In Query Manager, select a query in the list and click Edit.

The SQL Query dialog box opens, where you can select your data source and construct your query (see “Using the SQL Query dialog box” on page 293).

To export a query:

1. In Query Manager, select the query you want to export in the list and click the Export button.
2. Give the query file a name and select the folder in which you want to save it, then click Save.

   Query files that have been exported from Drumbeat have the .dql extension by default. You should save your files with this extension so that Drumbeat will recognize it.

To import a query:

1. In Query Manager, click the Import button.
2. Navigate to the folder that contains the query file you want to import and select it. Click OK.

Using the SQL Query dialog box

The SQL Query dialog box allows you to specify the data from your data source that you want to include. There are several ways you can define a query.

To create a new query, you can use any of these techniques:

- Type a SQL statement into the SQL Statement window.
- Copy and paste a SQL statement from another source into the SQL Statement window.
- Use a view or query stored with your database, or make a call to a stored procedure in the database management system.
- Use the SQL wizard to build a simple SQL statement for your query.

Building a simple query with the SQL wizard

You can use the SQL wizard to quickly construct a simple query.

To build a simple query with the SQL wizard:

1 From the SQL Query dialog box, select a data source. If your database is password protected, or you want to add password protection now, enter the required user name and password (otherwise, leave this blank).

2 To view the columns in a table in the data source, select the table you want to view in the Table list. The columns appear in the Columns list. (This feature allows you only to view the column content, but it does not select the table for building your query.) To begin creating your query, click the SQL Wizard button.

3 In the From Tables dialog box, select the table or tables you want to include in the query and click Next.

4 In the Select Columns dialog box, select the columns you want to include in the query. If you want all of the columns to be included, you can click Next, since this is the default condition (that is SELECT * FROM ClientStatus). However, if you want to exclude any columns you must deselect all the columns you want to include. When you have finished, click Next.

Note: The order in which you choose the columns will be the order in which they appear in the SQL statement and in the content table that you create for the query in Drumbeat. This can affect the order of their appearance in pages and AutoTables you later generate based on the content table.

5 In the Select Unique Key Columns dialog box, check the column or columns that make a unique key (primary key) in your database. If your query is based on only one table, you can click Next and skip this step. However, when using related tables, Drumbeat needs to know which columns are the unique keys. Click Next.

6 In the Where Clauses dialog box, define any filters you want for your query. Also, define any joins (related columns) between tables.

7 Click Finish and you will be returned to the SQL Query dialog box. The completed SQL statement will be displayed in the SQL Statement window.

8 To test the results of the query, click Test.

9 If you have two columns in different tables with the same name, you will be prompted to qualify the column names (see “Qualifying column names” on page 297.) Choose the tables from which each column comes from and click OK.

The results of you query appear in the SQL Query Result window. You can edit the statement directly in the SQL Statement window if necessary. Use the Find and Replace option to quickly change values in the SQL statement.

To save your query:

1 Provide a query name if you have not already done so.

2 Select Local if you want this query to be available to this site only. Select Global to make it available for use in other Drumbeat site files.

3 Click Save Query.

Disconnecting and reconnecting a data source

If you do not want the query to be executed during development in Drumbeat, you can click the Disconnect button in the SQL Query dialog box. This can speed up design when working with large databases. Advanced queries that use application objects like Session and Request *must* use the Disconnect option, as these objects are only valid when executed in the browser environment.

Sometimes when working with a data source your connection to the data source may become disconnected due to network failures.

To reconnect a data source when it has been disconnected:

1 Open Query Manager, select a query to the data source and click Edit.

2 In the SQL Query dialog box, click Disconnect, and then click Connect (the same button).

3 Click Save Query and then click Done.

Building a Complex Query

In many cases when you are using a large corporate database, the queries you will want to use as the basis of your database-driven pages are too complex for it to be practicable to build them in the SQL wizard. Complex queries can be created in two ways in Drumbeat:

- Use a pre-constructed view (or query) constructed in your database application by creating a SQL statement to select the view.
- Copy and paste a query from another source into Drumbeat.

Views and stored procedures in your database management system are shown in the Table window along with database tables. (A view is the same as a query in some database applications.) Select the view to show the columns it includes in the Columns window.

You can use the SQL wizard to construct a SELECT ALL statement for the View, if you prefer. You can also copy and paste a SQL statement from another application directly into the SQL Statement window. For example, if you are using Microsoft Access, you can build your query in Access, and then choose View > SQL View. Copy the SQL statement from Access and paste it into the SQL Query dialog box in Drumbeat. Once you have copied a query in this manner, you can edit it and add to it manually in Drumbeat.

To use a pre-constructed view (or query):

1. In the SQL Query dialog box, enter a query name, and select the data source.
2. Enter SELECT * FROM VIEWNAME in the SQL Statement window (substitute the name of the view or query for VIEWNAME).

To create a SQL statement to a view with the SQL wizard:

1. Click the SQL Wizard button.
2. In the From Tables dialog box, select the view. Click Next in all subsequent dialog boxes and Finish in the final dialog box.
3. Click the Test button to preview the results.

To copy and paste a SQL statement:

1. In the SQL Query dialog box, enter a query name and select the data source.
2. Copy the SQL statement from the other application and paste it into the SQL Statement window. View the results by clicking the Test button. Click Save Query.

Qualifying column names

If you have two field names in different tables with the same name, you need to qualify the column names to define which column belongs to which table. Drumbeat will prompt you to qualify column names in a query if necessary when you use the SQL wizard or click Test to test the results. The DataForm wizard will also prompt you to qualify column names if necessary when you create database-driven pages using the DataForm wizard. However, if you create your SQL query by hand, you can take the extra step to qualify column names in the SQL Query dialog box. If you are constructing pages manually, rather than using the wizard, you may need to do this.

To qualify column names:

1 Click the Qualify Column Names button in the SQL Query dialog box.

2 Columns with duplicate names will be displayed with an active, blank Table field next to them. Choose the correct table from the pop-up menu for each column.

Using stored procedures

Stored procedures that are part of your database application can be used as the basis of DataForm pages as well as views. Stored procedures are subprograms made up of SQL statements which typically take sets of parameters to perform a group of actions. When you use a view, you expect a result set, typically a list. Stored procedures can return a result set, or a group of output values, or both, when they are executed.

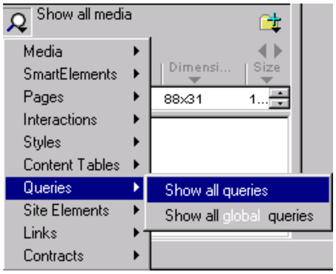
When you select a stored procedure in the SQL Query dialog box, the available parameters are revealed in the Parameters window. Parameter names and types are displayed in addition to the value. See “Using stored procedures” on page 303.

Listing, sorting, and editing queries

The different queries you create for your site are assets that can be listed and sorted in Asset Center.

To list queries in the Asset Center:

Click the Locate Assets button and choose Queries > Show All Queries.



Filter for global or local queries by choosing Show All Global Queries. Click the underlined word to change the filter criteria from global to local.

You can also edit a query that you have previously created by double clicking it in the Asset Center list. For more information, see “Editing queries” on page 300.

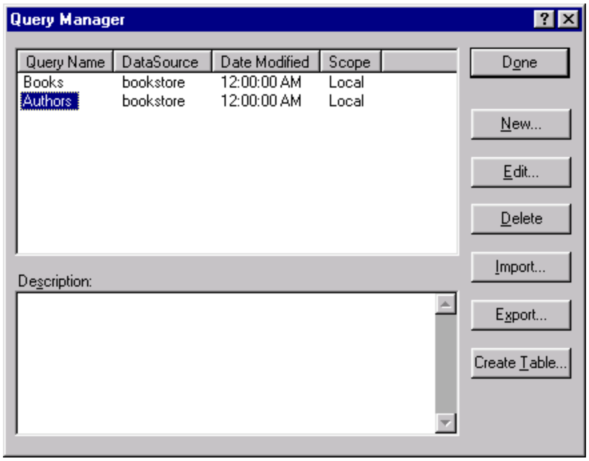
Creating an ODBC content table

Once you have created a query, you must create an ODBC content table based on the query to use the database information in your site. The content table is the visual representation of your data. The ODBC content table may contain references to images, other media, hyperlinks, and text elements. Any change made to the database, either from an external application or from DataForm pages (insert or update pages), is reflected in the content table and all pages that reference it.

When you create your query and click Save Query in the SQL Query dialog box, you are returned to the Query Manager where you create content tables. If you do not create your ODBC content table immediately after creating your query, you can create it later in the Content Center. You will not be able to build pages using this database content until you create a content table, however.

To create an ODBC content table, after creating a SQL query:

In Query Manager, select the query from the list and click Create Table.



Query Name	DataSource	Date Modified	Scope
Books	bookstore	12:00:00 AM	Local
Authors	bookstore	12:00:00 AM	Local

To create an ODBC content table from a data source in the Content Center:

1. In the Content Center, click the New Content Table button.
2. In the New content table dialog box, choose ODBC Data Source and click Select.
3. In Query Manager, select the query, and then click Done.
4. Name the content table and click OK.

To view your ODBC content table, open the Content Center and choose the table you want to view from the pop-up menu. If there are a large number of records created by the SQL query, you can limit the number of records displayed to improve efficiency.

To limit the number of results displayed in an ODBC content table:

1. Choose File > Preferences > Drumbeat.
2. Click the Content tab, specify the maximum number of displayed rows.

Hiding records

Turn off display of *all* records in Drumbeat if you do not need to see results during development. This can improve efficiency when working with large databases and hide established Web sites. When you publish your pages, all records will be included.

To ride records:

1 Choose File > Preferences > Drumbeat.

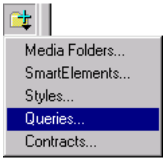
2 Click the Content tab and deselect Display Results for SQL.

Editing queries

Queries may be edited at any time. You can revise a query to incorporate new data even after generating pages with DataForm pages.

To edit a query:

1 Click the Add Assets button and choose Queries.



2 In the Query Manager, select the query you want to edit and click Edit.

3 In the SQL Query dialog box edit the query and click Save Query. In Query Manager, click Done.

You can also select a query in the Asset Center and enter the SQL Query dialog box directly to edit it.

To edit a query from Asset Center:

1 Click the Locate Assets button. From the pop-up menu, choose Queries.

2 Double-click the query you want to edit in the Asset Center list, or right-click the query and choose Properties.

3 In the SQL Query dialog box, edit the query, and then click Save Query.

Refreshing the records in a content table

After editing a query, you may need to refresh the content table made from the query in order for it to show the new results. The content table will automatically be updated when you close and open the site file. You can also refresh it within Drumbeat.

To refresh the records in an ODBC content table:

1 Choose the table from the pop-up menu in the Content Center.

2 Click the Edit Content Table button.

3 In the Edit dialog box, click Select. Query Manager opens with the query selected.

4 Click Edit.

5 In the SQL Query dialog box, click Disconnect, and then click it again. This refreshes the connection to the database and updates the content table.

6 Click Save Query, and then Done in Query Manager. Click OK in the Edit Content Table dialog box.

Note: To add new fields to DataForm pages built on a query you must either rerun DataForm wizard and select the columns you want to add to the pages (see "Rerunning the DataForm wizard" on page 332), or add SmartElements to the pages manually and bind them to the appropriate columns in the record set (see "Using form elements" on page 349). How you choose to do this will depend on how extensive the changes are. If the new fields must participate in any interactions, such as update and insert, you must also set up the interactions.

Creating dynamic queries with SQL parameters

Using SQL parameters, you can create SQL queries with parameters that can be changed dynamically on the page. Dynamic SQL queries can dramatically reduce access time over conventional filter operations when working with large databases, as filter operations can work on small subsets of data. It can also reduce the number of queries you need to create for your Web site, as the same query can be reused with different parameter values.

To create dynamic SQL queries:

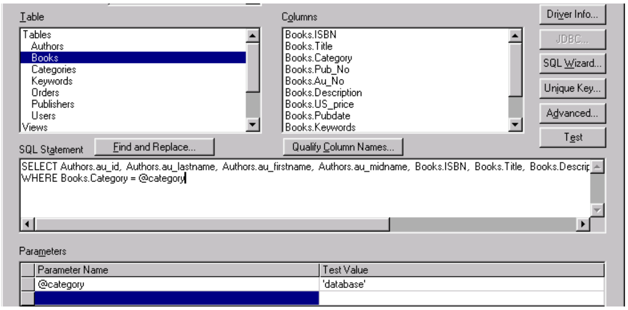
1 In the SQL Query dialog box, select a Table or view.

2 Right-click the Parameter Name column head (near the bottom of the dialog box) and choose Append Parameter.

3 Enter the name of the parameter you want to create.

4 Enter a test value for the query. The test value becomes the default value used on the layout, which you can then change on the page by applying interactions. If the test value is a text string, you must enclose it in single quotation marks.

5 Create a SQL statement for the query (for example, SELECT * FROM TableName) with a WHERE clause that includes your parameters.



```
SELECT Authors.au_id, Authors.au_lastname, Authors.au_firstname, Authors.au_midname, Books.ISBN, Books.Title, Books.Descrip
WHERE Books.Category = @category
```

Parameter Name	Test Value
@category	'database'

6 Click Test to view the query results. Correct the statement if necessary.

7 Click Save Query and create a content table in the Query Manager.

By default the test value for the query will be used on the pages. Apply interactions or custom scripting to change the parameter values. See also “Filtering data with SQL parameters” on page 391.

Using stored procedures

Stored procedures that are part of your database application can be used to create Active Server Pages. You can use stored procedures to create pages with the DataForm wizard or manually. Stored procedures are subprograms made up of SQL statements that typically take sets of parameters to perform a group of actions.

Stored procedures either return a result set, a value or values, or both values and a result set.

The Command object

The Command object is used for connecting to stored procedures that return output values rather than result sets. The Command object can be found in the Standard tab of the Element Library.

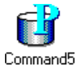
To add the Command Object to the SmartElements toolbar and use it on a page:

1. Click the Element Library button on the SmartElements toolbar.
2. Click the Standard tab, and select Command object.
3. Drag the Command object from the SmartElements toolbar into the Basement.
4. In the Attributes tab, choose the query to which it should be bound and rename it if you wish.

To add the Command object to the page from Asset Center:

1. Click the Locate Assets button and choose SmartElements > Show All SmartElements.
2. Find the Command object in the list and drag it from the Asset Center to the Basement.
3. In the Attributes tab, choose the query to which it should be bound and rename it if you wish.

After you have added the Command object to a page, it will appear as a site element in Asset Center, just like the recordset.



The Command object as it appears in the Basement.

The content of the Command object will always be a query instead of a content table, since you cannot create a content table for a stored procedure that does not return a result set. Assign the query that you want to bind to the Command object through the Attributes tab. The attributes of the Command object are as follows:

- The name to use for the Command object.
- The name of the query that calls the stored procedure.
- The Command Time Out option indicates how long to wait while executing a command before timing out.
- The Prepared option indicates whether or not to save a compiled version of a command before execution.
- The Returns Result Set option indicates that the stored procedure returns a result set. This will allow you to link a recordset to the stored procedure to display the result set.

Using stored procedures that return a result set

When you select a stored procedure as the basis of a query in the SQL Query dialog box, the available input parameters are shown in the Parameters window.

To create a query to a stored procedure that returns a result set:

1 In the SQL Query dialog box, select a data source.

2 Select the stored procedure from the table list. Stored procedures have a built-in CALL statement that will be shown in the SQL statement window. The CALL statement is grayed out, since you cannot edit this statement, but you can assign default parameter values.

3 In the Parameters window, add a test value. The test value lets you to test the results with the Test button. The test value becomes the default value used on the page until the value is changed with an interaction.

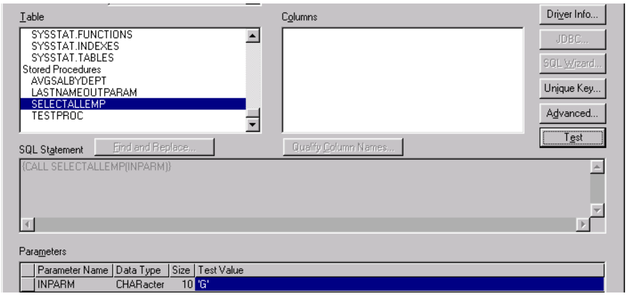


Table:

SYSSTAT.FUNCTIONS
SYSSTAT.INDEXES
SYSSTAT.TABLES
Stored Procedures
AVGSALBYDEPT
LASTNAMEOUTPARAM
SELECTALLEMP
TESTPROC

SQL Statement

```
{CALL SELECTALLEMP(INPARM)}
```

Parameters

Parameter Name	Data Type	Size	Test Value
INPARM	CHARacter	10	'G'

The purpose of the query to the stored procedure shown in the previous illustration is to search a database of employees by last name. The input argument is a string that will match last names via a "quick search," For instance, entering "P" will return all employees whose last names begin with P, or entering "Pul" will return all employees whose last names begin with or include "Pul".

4 Name and save the query.

5 In Query Manager, click Create Table to create a content table for the query.

If you do not want to execute the procedure during development (for instance, if the database is unavailable), click the Advanced tab and deselect Execute SQL at Design Time. In this dialog box, you can also add the columns that the procedure returns, in order to create a content table and work with the results for design purposes. See "Using stored procedures that return values" on page 309.

To use the sample stored procedure and display the results in an AutoTable:

1. Place a Command object in the basement of the page.
2. In the Attributes tab for the Command object, select Returns Result Set and select a query.
3. Place a recordset element in the basement. In the Attributes tab, for the content table, choose the content table you created from the stored procedure.
4. Place an AutoTable on the page to hold the results. (Alternatively, you can display the results on a separate page. See “Displaying results of a stored procedure on a separate page” on page 315.)
5. In the Attributes tab for the AutoTable, for Content choose Recordset.
6. In the Content dialog box, choose the columns you want to display.
7. Select the recordset and the Command object in the basement. Right-click and choose Possible Interactions. Apply the interaction in the stored procedure category:

   Link [Recordset] to [Stored Procedure]

   The names of your recordset and stored procedure will be substituted for the tokens in brackets.

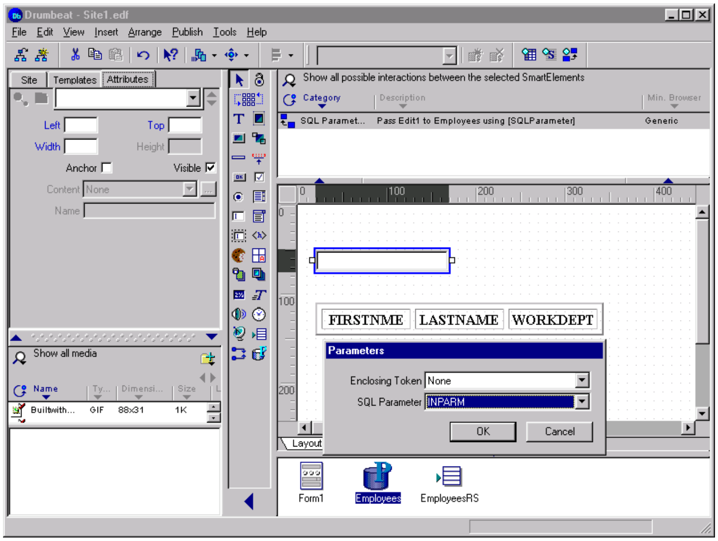
Next, you need to create the element on the layout that will take the input parameter.

To create the input parameter:

1. Place an edit box on the page.
2. Select the edit box and the Command object in the basement. Right click and choose Possible Interactions.
3. Apply the interaction in the SQL Parameters category:

   Pass [Edit1] to [Command] using [SQLParameter]

4. In the Parameters dialog box, choose the SQL parameter that is being passed. (All available parameters will be shown in the pop-up menu.) Choose the enclosing token, if required. (This will depend on your database and the type of entry. In this example, no enclosing token is necessary.)



Drumbeat - Site1.edf

Show all possible interactions between the selected SmartElements

Category	Description	Min. Browser
SQL Paramet...	Pass Edit1 to Employees using [SQLParameter]	Generic

Parameters

Enclosing Token: None

SQL Parameter: INPARM

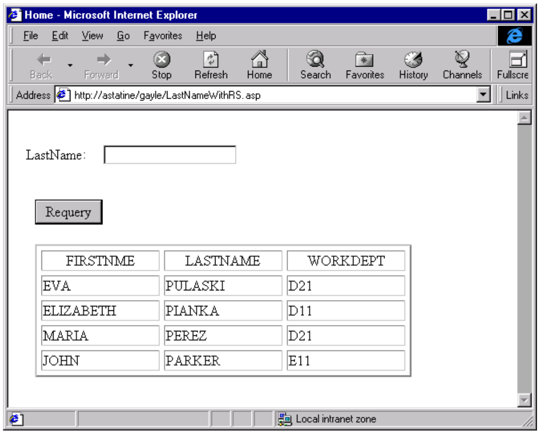
Name	Ty...	Dimensi...	Size
Builtwith...	GIF	88x31	1K

Setting up the interaction to pass the input value from the edit box to the stored procedure.

5. Add a form button for submitting the form.

6 Right-click the Command object in the Basement and choose Possible Activations. Find and apply the interaction in the Stored Procedures category:

Execute BeanSP


Home - Microsoft Internet Explorer
File Edit View Go Favorites Help
Back Forward Stop Refresh Home Search Favorites History Channels Fullscre
Address http://astatine/gayle/LastNameWithRS.asp
Links
Requery
Local intranet zone

LastName:

FIRSTNME	LASTNAME	WORKDEPT
EVA	PULASKI	D21
ELIZABETH	PIANKA	D11
MARIA	PEREZ	D21
JOHN	PARKER	E11

The published page that calls the stored procedure is shown. The edit box takes the input parameter (an initial letter) and the AutoTable displays the results of the procedure (all employees whose last names begin with the letter).

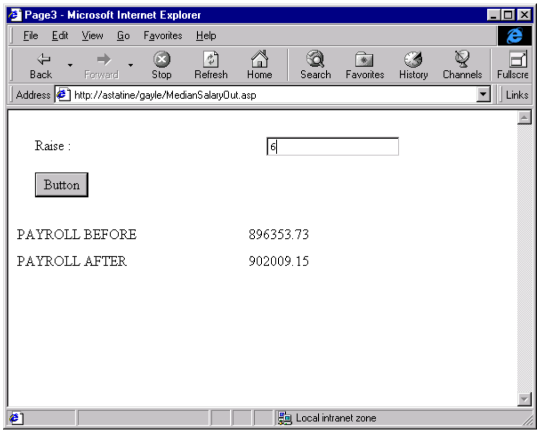
You can also use DataForm wizard to create DataForm pages from a stored procedure that returns a result set, just as you do with other queries. In the DataForm wizard, however, search and sort are not available for stored procedures.

Using stored procedures that return values

Drumbeat cannot display test results for a stored procedure that returns a value rather than a result set. Therefore, you should disable design-time execution. Click the Advanced button in the SQL Query dialog box and deselect Execute SQL at Design Time.

Since a stored procedure that returns values does not return a result list, you also cannot create a content table from it. You will want to add a test value for any input parameters, however, to create a default value. The Command object, instead of a recordset, is used to retrieve the value from the stored procedure.

The following illustration shows a sample query created to a stored procedure that returns only values. The purpose of the procedure is to first, calculate the total current payroll, and second, calculate the total payroll based on a projected percentage raise.



Address http://astatine/gayle/MedianSalaryOut.asp

Raise : 6

Button

PAYROLL BEFORE 896353.73

PAYROLL AFTER 902009.15

The published page is shown. The edit box takes the input parameter (an integer representing a percentage raise). The second text field displays the result of the procedure after stored procedure executes.

To create a page that uses the sample stored procedure which returns a value only:

1. Place a Command object in the Basement.
2. In the Attributes tab for the Command object, select the query to which the Command object is bound and rename it if you wish.
3. Place a text box on the page to hold the first output parameter (PayrollBefore) and another to hold the second output parameter (PayrollAfter). (Alternatively, you can display the results on a separate page. See “Displaying results of a stored procedure on a separate page” on page 315.)
4. In the Attributes tab for each text box, for Content choose Stored Procedure. In the Content dialog box, choose the Command object and the OutParam.
5. Place a text box on the page to hold the input parameter (an integer representing the percentage raise). Select the text box and the Command object. Right-click and choose Possible Interactions. Find and apply the interaction in the SQL Parameter category:

   ```
   Pass [Edit1] to [Command] using [SQLParameter]
   ```

   The tokens in brackets are replaced by the names of the elements on your page.
6. In the Parameters dialog box, choose the SQL parameter that is being passed. (All available parameters are shown.) Choose the enclosing token, if required.
7. Add a form button for submitting the form.
8. Finally, select the Command object in the Basement. Right-click and choose Possible Activations. Find and apply the interaction in the Stored Procedures category:

   ```
   Execute [Command]
   ```

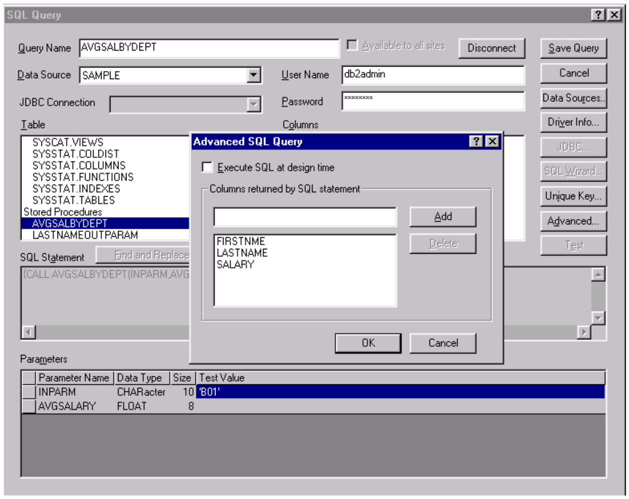
   The token in brackets will be replaced with the name of your Command object.

Stored procedures that return both a result set and output values

Some stored procedures may return both a result set and output values. With this type of stored procedure, you need to create a content table to use the part of the procedure that includes a result set. You must also use a recordset to display the results set and a Command object to obtain the output parameter value and link the two objects together.

When you create a query to a stored procedure that returns both a result set and a value, do the following:

- Since the output value cannot be displayed until the query is executed, when you create the query, click the Advanced button in the SQL Query dialog box and deselect Execute SQL at Design Time. Also, enter the columns that you want to be able to display on the layout.
- Create a content table from the query that displays the result set part of this query.
- The following illustration shows a query to a stored procedure that calculates the average payroll for the employees in a department. It returns the names of all people in a selected department and their salaries, and calculates the average salary for the department.



SQL Query

Query Name: AVGSALBYDEPT

☐ Available to all sites

Data Source: SAMPLE

User Name: db2admin

JDBC Connection:

Password: ********

Table

SYSCAT.VIEWS
SYSSTAT.COLDIST
SYSSTAT.COLUMNS
SYSSTAT.FUNCTIONS
SYSSTAT.INDEXES
SYSSTAT.TABLES
Stored Procedures
AVGSALBYDEPT
LASTNAMEOUTPARAM

Columns

SQL Statement

{CALL AVGSALBYDEPT(INPARM,AVG

Advanced SQL Query

☐ Execute SQL at design time

Columns returned by SQL statement

FIRSTNME
LASTNAME
SALARY

Parameters

Parameter Name	Data Type	Size	Test Value
INPARM	CHARacter	10	'B01'
AVGSALARY	FLOAT	8	

To create a page using this sample stored procedure, which returns both a value and a result set:

1 Place a Command object in the Basement.

2 In the Attributes tab for the Command object, select the query to which the Command object is bound. Select Returns Result Set.

3 Place a recordset element in the Basement. In the Attributes tab, choose the content table you created from the stored procedure.

4 Place an AutoTable on the page to hold the results. (Alternatively, you can display the results on a separate page. See "Displaying results of a stored procedure on a separate page" on page 315.)

5 In the Attributes tab for the AutoTable, for Content choose Recordset. In the Content dialog box, choose the columns you want to display in the AutoTable.

6 Select the recordset and the Command object in the basement. Right-click and choose Possible Interactions. Apply the interaction in the Stored Procedure category:

   Link [Recordset] to [Stored Procedure]

   The names of your recordset and stored procedure will be substituted for the tokens in brackets.

Now you need to place the form element to take the input parameter. In this case, the example uses a pop-up menu from which users can choose the department name. The dropdown list is populated from a separate query to the database.

To create the dropdown list:

1 Place a pop-up menu on the page to take the in put parameter. The content of this dropdown list will come from another recordset representing the departments, and the value it will pass is the department code. See “Retrieving an option list from a related table” on page 362.

2 In the Attributes tab for the dropdown list, set the Content option to Recordset and select the Depts recordset and the DeptName column.

3 Set the Value option to Recordset and select the Depts recordset and the DeptNo column.

4 Set the Selection to Server Expression and enter the Expression:

```
Request(“List1”)
```

Where List1 is replaced with the name of the listbox.

5 Select the list box and the Command object in the Basement. Right-click and choose Possible Interactions. Select and apply the interaction in the SQL Parameters category:

```
Pass [Edit1] to [Command] using [SQLParameter]
```

6 In the Parameters dialog box, choose the SQL Parameter that is being passed. Choose the enclosing token, if required.

Now you can set up the element on the page that will display the value returned from the stored procedure. In this example, the value is the average salary for the department.

To display the value:

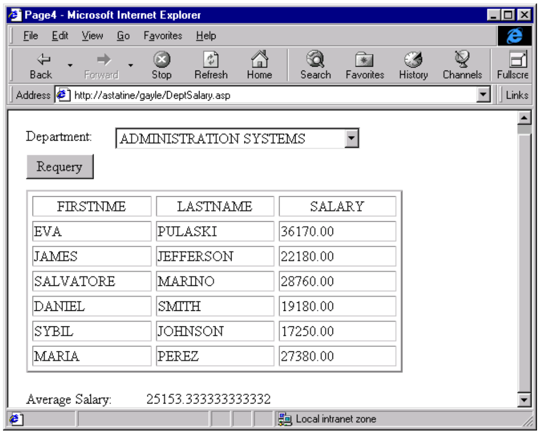
1 Place a text box on the page to hold the output parameter (AvgSalary).

2 In the Attributes tab for the text box, for Content choose Stored Procedure. In the Content dialog box choose the Command object and the output parameter (OutParam).

3 Add a form button for submitting the form.

4 Select the Command object in the Basement. Right-click and choose Possible Activations. Find and apply the interaction in the Stored Procedures category:

Execute Command



http://astatine/gayle/DeptSalary.asp

Department: ADMINISTRATION SYSTEMS

FIRSTNME	LASTNAME	SALARY
EVA	PULASKI	36170.00
JAMES	JEFFERSON	22180.00
SALVATORE	MARINO	28760.00
DANIEL	SMITH	19180.00
SYBIL	JOHNSON	17250.00
MARIA	PEREZ	27380.00

Average Salary: 25153.333333333332

The published page is shown. The edit box shows the input parameter. The AutoTable displays the results of the procedure. The text box below the table displays the value returned by the stored procedure after it executes.

The Command object needs to establish the database connection and then pass it to the recordset. Basement objects are created left to right, so you need to ensure that the Command object is the first one in the basement.

To move the Command object to the left of the recordset.

Select the Command object and drag it to the left-most position in the Basement.

You can now use interactions to set the parameters in the Command object if you want.

Displaying results of a stored procedure on a separate page

The foregoing examples all display the results of the stored procedure on the same page which contains the input box. You may prefer to display the results of the stored procedure on a separate page after entering your input parameters. (Otherwise, when the page loads, the default values are shown on the page.)

To display the results of a stored procedure on a separate page:

1 Set the parameters on the first page.

2 Execute the stored procedure and display results on the second page.

The first page must contain a form element. If you are using a form element for the input variable or a form button on the page (as in the previous examples) you will already have a form element in the Basement. If there is no form element on the page, you can add a hidden form element to the Basement, which will also create a form element.

To set up the first page:

1 Place the Command object in the Basement and in the Attributes tab, select the query.

2 Place the form element to contain the input variable on the layout.

3 Select the form element that contains the input variable and the Command object. Right-click and choose Possible Interactions. Find and apply the interaction in the SQL Parameters category:

   Pass [source] to [Command] using [SQL Parameter]

4 In the Parameters dialog box, choose the enclosing token, if required, and the SQL parameter.

5 Add a form button for submitting the form.

6 Select the Command object and the form button. Right-click and choose Possible Interactions. Find and apply the contract in the Database navigation category:

   Go to [pagename] maintaining state of Stored Procedure when [Button] is clicked

7 In the Parameters dialog box, click the Assign Page Name button and select the page from the site tree.

To set up the second page:

1 Place the Command object in the Basement and in the Attributes tab, select the query.

2 Place a recordset in the Basement and bind it to the content table created from the stored procedure.

3 Select the recordset and the Command object in the basement. Right-click and choose Possible Interactions. Apply the following interaction in the Stored Procedure category:

   Link [Recordset] to [Stored Procedure]

   The names of your recordset and stored procedure will be substituted for the tokens in brackets.

4 Place an AutoTable on the layout to display the results. In the Attributes tab for the AutoTable, for Content select Recordset.

5 In the Content dialog box, select the content table. Then choose the columns you want to display in the AutoTable. Be careful to preserve the order of the columns.

6 If the stored procedure is returning a value, place a text box on the layout to hold the value and set its Content to the Stored Procedure. In the Content dialog box, select the Stored Procedure and the ouput parameters.

7 Select the form element, right-click and choose Possible Activations. Find and apply the interaction in the Stored Procedures category: Execute Command.

CHAPTER 16
DataForm Wizard

The DataForm wizard is an extremely powerful tool that allows you to create database-driven pages from a wide range of data sources without coding and with all the dynamic features of active, server-driven web pages. The DataForm wizard easily and automatically generates Active Server Pages that perform all the most common database operations, including browsing master detail pages, accessing records, updating and inserting database information, and searching and navigating records.

This chapter explains how to set up for database publishing and use the Drumbeat DataForm wizard to create DataForms. Chapters 17 and18 go into more detail on the ways you can customize DataForm pages, use form elements, and use recordsets to create custom database operations.

Setting up for Active Server pages

Before you can use the DataForm wizard, you need to ensure that you are properly set up for database publishing. Building Active Server Pages requires a few preliminary steps.

Before beginning to build Active Server pages you need to:

- Configure your Web server to be ASP-compatible.
- Create a web publishing directory for your site.
- Set up publish settings for ASP in Drumbeat.

These steps are briefly described following. For more detailed information, see "Publishing Active Server Pages" on page 189.

Configuring your Web server

To publish Active Server Pages, an ASP-compatible web server is required.

Microsoft Windows NT Server can be configured for ASP by installing Microsoft Internet Information Server (IIS) 4.0 or 3.0. IIS is an add-on to NT Server and is included in Microsoft Windows NT 4.0 Option Pack, free for download from the Microsoft Web site.

To get Internet Information Server 4.0:

Go to: www.microsoft.com/products/prodref/534_ov.htm

Netscape Enterprise Server, Lotus Go, and a number of other Web servers can be configured for ASP by installing ASP interpretive software. Chili!ASP is one package designed for this purpose, which is included on the Drumbeat CD.

To publish and view ASP pages on your own machine or over a small network, you must install either IIS or Personal Web Server. Personal Web Server is a modified version of IIS that can be used on Windows NT Workstation or Windows 95/98. Personal Web Server is free from the Microsoft Web site.

For more information on Personal Web Server, see“Personal Web Server” on page 189.

Creating a Web directory

When you install Microsoft Internet Information Server (IIS) or Personal Web Server, a default directory is created for your Web documents at Inetpub\wwwroot. You can create subdirectories for each of your Web sites under this root directory. The name of the subdirectory becomes the alias for your site. Alternatively, you can create a virtual directory, if you want to post your site to a different location.

For more information on creating a Web directory, see “Creating a Web directory” on page 190.

For more information on creating a virtual directory, see “Creating a virtual directory” on page 191.

Configuring Drumbeat Publish Settings for ASP

Before you can configure the publish settings for ASP, you must set your server application support for the site to ASP. Setting server support for ASP also enables the DataForm wizard.

To set server support:

1. From the File menu, choose File > Preferences > Site.
2. In the Site Preferences dialog box, under Server Application Support, choose ASP, and click OK.

Choosing ASP server support activates the New DataForm button in Site Manager and the DataForm option on the Insert menu, which are otherwise disabled. Once server support is set, you will be able to access the DataForm wizard through the New DataForm button or by choosing Insert > DataForm.

Next, you need to configure the ASP publish settings in order to publish and view your Active Server Pages.

To configure Drumbeat publish settings:

1. From the Publish menu, choose Publish > Publish Settings.
2. In the General tab, enter the destination directory and server name.
3. In the ASP tab, enter the alias and default page for the site. If you have set up a virtual directory, enter the path to the virtual directory.

Note: The page you select as your default page is the page that will always be accessed first when you view your site. You can change this as often as you like as you work with your site to avoid having to "click through" to the page you want to preview.

DataForm operations

The DataForm wizard automatically creates six types of database interactions on five pages:

- Detail: Displays record details in browsable format, one record at a time.
- Search: Allows records to be searched based on specified criteria. Can be expanded to include searching on multiple fields and criteria.
- Result List: Creates a page to display matching results from a search in table format.
- Update Records: Creates a form to allow users to edit and update individual records.
- Insert Records: Creates a form to allow users to insert new records in the database.
- Delete Records: Allows users to delete records in the database. (When you choose this option, a Delete button is added to the detail page. No new page is created.)

A complete DataForm set contains five pages based on the same data source (the same ODBC content table created from the same query to the database). You can generate all five DataForm pages at once, or create only those pages you want. The detail page is always created as the basis for any DataForm you create.

Rather than creating all five pages in one DataForm, you may want to separate the display and maintenance functions of the site.

- For search and display functions on the detail, search, and result list pages, you may want to create a query that includes the appropriate database joins in order to display the information you want from different tables in the database.
- For update and insert pages, you may need to use a different query, since insert operations are frequently not allowed in table joins. However, you can still use related table information on the page—for instance, to make pop-up menus display understandable choices—by adding these functions to the page after you create them.

It is best to outline the types of data-driven pages you would like to have before you start the DataForm wizard. Some possible scenarios might be:

- You want to allow various people in your organization to view, update, insert, and delete records as necessary.
- You want to create a password system that allows only certain people, such as account executives and project managers, to update records, with the rest having view-only access. Password protection can be added to your pages after you have generated them.
- You have a data source that is maintained external to the web application, so creating editing functions will be unnecessary.
- You want customers on a public web site to be able to display and search records only.
- You want site visitors to be able to enter and update personal information into a database that can then be used for filtering displays or creating transactions.

Running DataForm wizard

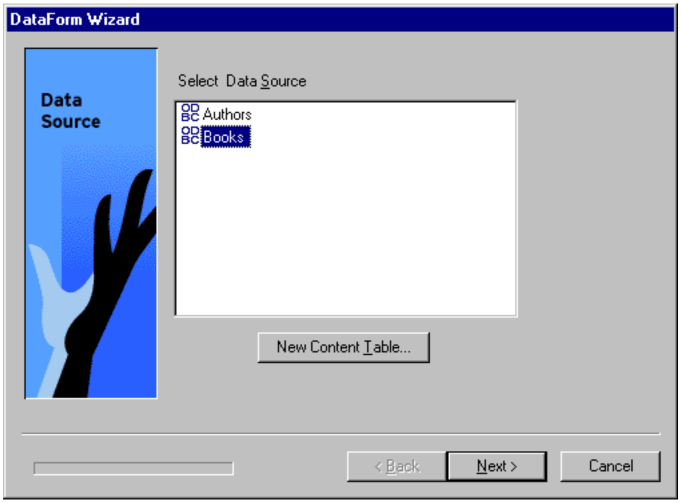
Before you can use the DataForm wizard, you must:

- Set up a web directory. See “Configuring your Web server” on page 318.
- Set up access to your database. See “Configuring Drumbeat Publish Settings for ASP” on page 319.
- Set server application support to ASP. See “Configuring Drumbeat Publish Settings for ASP” on page 319.
- Create a content table from an ODBC data source that contains the database content you want to use in your pages. See “Creating an ODBC content table” on page 298.

To activate DataForm wizard:

From the main menu, choose Insert > DataForm, or in Site Manager click the New DataForm button.

In the first dialog box, you must select the ODBC content table that you want to use in your pages. Only tables populated by an ODBC data source can be used with the DataForm wizard (not manual content tables or CSV content tables).



Choose an ODBC content table first when you enter the DataForm wizard.

If you have not already created an ODBC content table for the database query you want to use, click New Content Table to enter the Query Manager and create the ODBC content table now. When you finish, you will be returned to the DataForm wizard.

Existing recordsets will be displayed in the Data Source dialog box if 1) you have already created a DataForm in this site or 2) you have manually added a recordset to a page in the site and bound it to an ODBC content table.

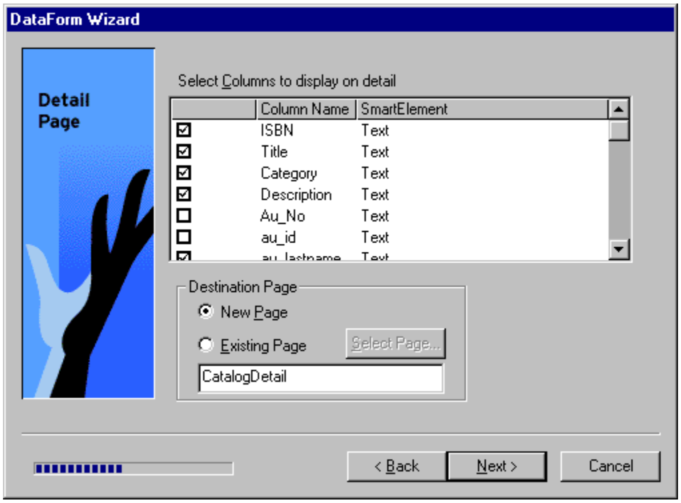
Instead of a content table, you can choose an existing recordset for your new DataForm. Multiple DataForms can be created from a single recordset, rather than creating a new recordset for each use.

DataForm detail page

Once you have chosen a data source for your DataForm, the DataForm wizard asks you to name the DataForm and choose the pages you want to create. A detail page will always be created, so this option is selected by default and grayed out. If you do not enter a new name for your DataForm set, the default name DataForm# (where # is a number) will be used.

In the Messages dialog box, you can choose the message you want to display when users reach the end of record navigation while browsing detail pages, or accept the defaults (No more records).

In the next dialog box, choose the columns in the content table that you want to include on the detail page by checking the column names.



Select Columns to display on detail

	Column Name	SmartElement
☑	ISBN	Text
☑	Title	Text
☑	Category	Text
☑	Description	Text
□	Au_No	Text
□	au_id	Text
☑	au_lastname	Text

Destination Page

● New Page

○ Existing Page

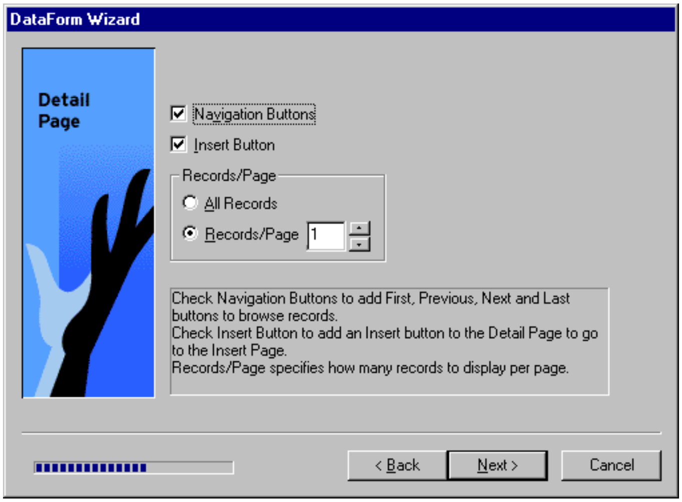
CatalogDetail

Choose the columns to display in the detail page.

By default, the DataForm wizard will create a new page for the detail page. You can edit the page name in the text box at the bottom of the screen to provide your own page name. However, you can use an existing page, if you prefer. Select Existing Page and click Select Page to select a page from your current site structure. If you choose an existing page, any elements already on that page will be retained and the detail record information will be added to it.

Other options for the detail page include:

- Navigation Buttons: First, previous, next, and last form buttons will be added to the detail page by default. Deselect the Navigation Buttons option if you do not need these buttons for browsing operations. If you want to use your own image buttons, you can replace the standard buttons later. See "Customizing form buttons" on page 343.
- Insert Button: You can create an insert form button linked to the insert page from the detail page. If you do not want users to be able to access the insert page when browsing records, leave this box option deselected.
- Records/Page: If you want to display more than one record per page, enter the number you want in this box. A data loop will automatically be created on the page. (The number of records per page can be changed later in the Attributes tab for the Looper element on the page.)



DataForm Wizard

Detail Page

☑ Navigation Buttons

☑ Insert Button

Records/Page

○ All Records

● Records/Page 1

Check Navigation Buttons to add First, Previous, Next and Last buttons to browse records.
Check Insert Button to add an Insert button to the Detail Page to go to the Insert Page.
Records/Page specifies how many records to display per page.

Options for the detail page include navigation buttons for browsing records and how many records to display per page.

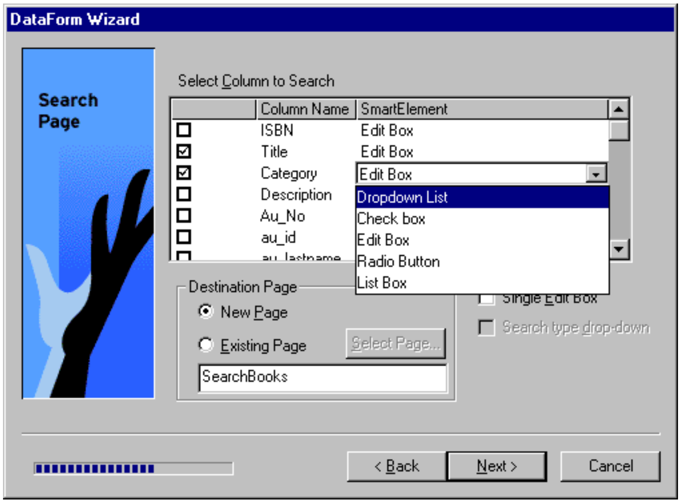
For more information on data loops, see "Using a data loop" on page 367.

DataForm search page

If you have selected the Search Page option in the main screen of the DataForm wizard, you can choose options for the search page in the DataForm wizard.

You can also add a search page to a DataForm set you have already generated. See “Adding and deleting DataForm pages” on page 334.

In the Search Page dialog box, select the column or columns in the database you want users to be able to search. You may select more than one column. Choose whether you want to create a new page (rename it as you like in the text box) or use an existing page for the output.



Select Column to Search

	Column Name	SmartElement
□	ISBN	Edit Box
☑	Title	Edit Box
☑	Category	Edit Box
□	Description	
□	Au_No	
□	au_id	
□	au_lastname	

Destination Page

● New Page

○ Existing Page

SearchBooks

□ Single Edit Box

□ Search type drop-down

Choose columns to search with the search function and the type of control you want for each.

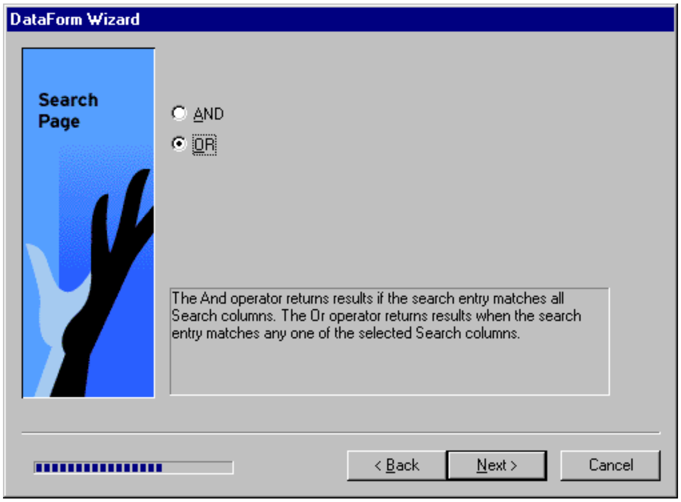
By default, the DataForm wizard creates a text box in which users may enter search text. However, you can choose the type of form element for each column by clicking the choice in the list and choosing the type from the pop-up menu. If you want the content of the pop-up menu to come from another source (such as a related table or a manual list) you can set this up after you generate your page. See “List boxes” on page 357.

For example, in the search page being created in the figures shown in this section, users will be able to search on book title and category, with the category being chosen from a pop-up menu (a dropdown list).

Other options you can choose for the search page are:

- Single Edit Box: Check this if you want an entry in a single edit box to search several columns (for instance, if you want a text string to be used to search two or more fields).
- Search type dropdown: Check this if you want users to be able to select search parameters that include Begins With, Contains, Ends With, and Equal To.

In the subsequent dialog box, you can choose the search criteria, if you've chosen more than one column for the search operation. The AND operator returns results if the search criteria matches all columns, while OR returns results where the search criteria is found in any column.



○ AND

● OR

The And operator returns results if the search entry matches all Search columns. The Or operator returns results when the search entry matches any one of the selected Search columns.

Choose AND/OR operators for the search function.

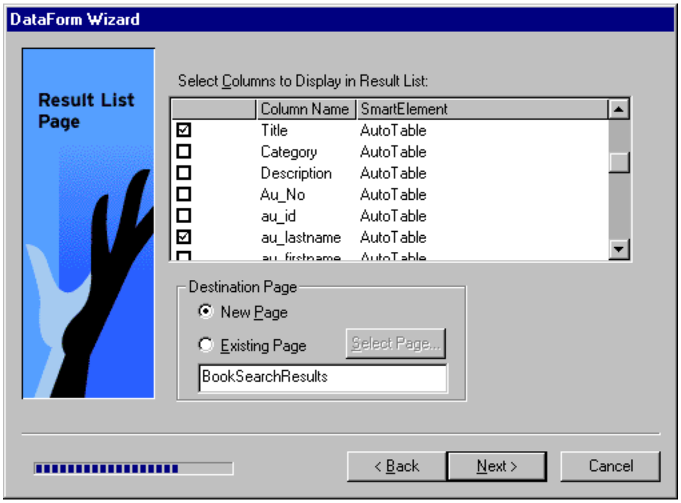
To learn how to build a search page manually without using the DataForm wizard, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

DataForm result list page

A result list page displays the results of a search in abbreviated form in a table, with links to the detail page for each matching record if you choose. A result list page can also be used to provide an abbreviated list of all records by default (when the page is accessed directly, instead of via a search), so you might choose to create a result list page to list your records even if you do not use the search function.

Create a result list page by selecting the Page option in the main screen of the DataForm wizard. You may also add a result list page to a DataForm set you have already generated. See “Adding and deleting DataForm pages” on page 334.

In the Result List dialog box, select the columns you want to be displayed in the result list page. Choose whether you want to create a new page (you can rename it in the text box) or use an existing page for the output.



	Column Name	SmartElement
☑	Title	AutoTable
□	Category	AutoTable
□	Description	AutoTable
□	Au_No	AutoTable
□	au_id	AutoTable
☑	au_lastname	AutoTable
□	au_firstname	AutoTable

Destination Page

● New Page

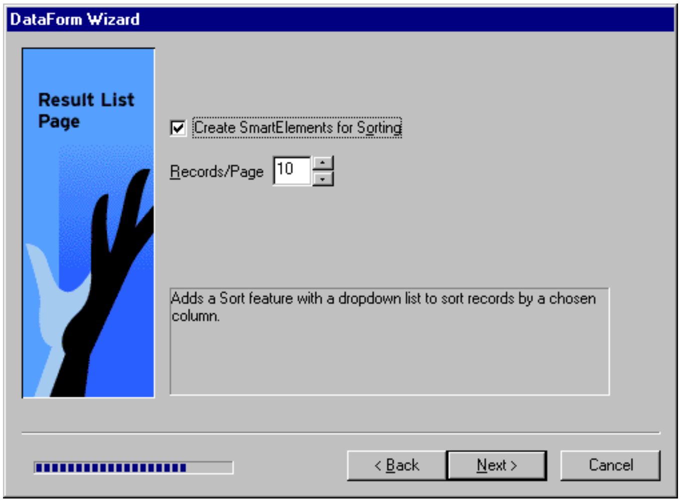
○ Existing Page

BookSearchResults

Choose the columns to display in the result list page.

Select Create SmartElements for Sorting if you want to be able to sort the results by different columns in the database. (For instance, if you choose to display both title and category, you will be able to sort the result list by title or category by choosing that option from a pop-up menu.)

The Max Records/Page option controls the number of records shown at one time in the table. This option can be changed in the Attributes tab for the AutoTable later. Navigational buttons (More and Back) will allow for accessing more records if the number of matching records exceeds the Max Records number.



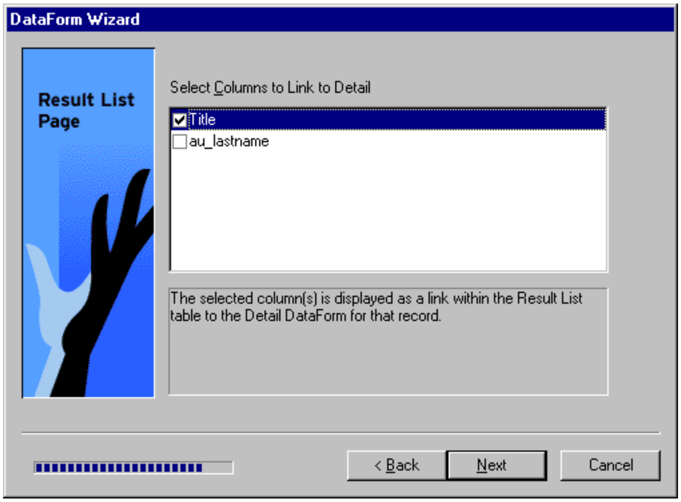
☑ Create SmartElements for Sorting

Records/Page 10

Adds a Sort feature with a dropdown list to sort records by a chosen column.

Options for the result list page include a feature to sort by different columns and the number of records to display per page.

You can make records in the result list link to their respective detail pages by choosing the column that you want to link. If you choose this option and you have not defined the unique key columns in the query, you will be prompted to choose the unique keys in a subsequent dialog box.



Select Columns to Link to Detail

☑ Title
□ au_lastname

The selected column(s) is displayed as a link within the Result List table to the Detail DataForm for that record.

Select columns from the result list to link to record detail pages.

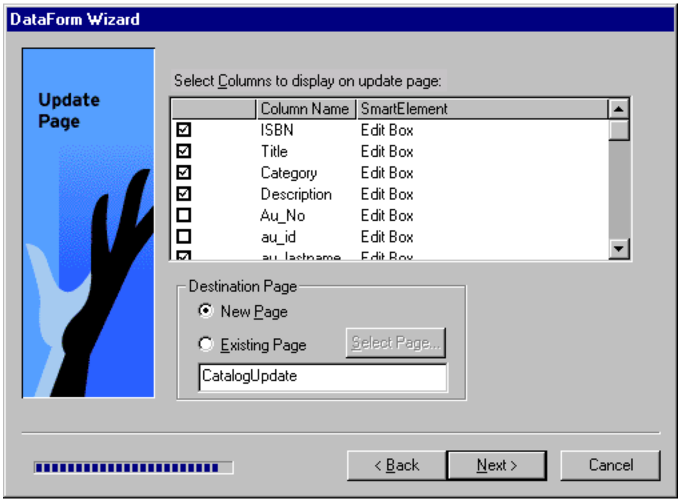
DataForm update page

Create an update page if you want to create a page for users to edit and update existing record entries from a Web page interface. You can create an update page by selecting the Page option in the main screen of the DataForm wizard.

You may also add an update page to a DataForm set you have already generated. See “Revising DataForms” on page 332.

Choose whether you want to create a new page (rename it as you like in the text box) or use an existing page for the output. Choose the columns that you want to be updateable. Update operations are restricted to only those columns you choose to display.

Note: If you have an automatic numbering operation (or database trigger) for your primary key in the main database application, be sure you deselect these fields for the update page.



Select Columns to display on update page:

	Column Name	SmartElement
☑	ISBN	Edit Box
☑	Title	Edit Box
☑	Category	Edit Box
☑	Description	Edit Box
□	Au_No	Edit Box
□	au_id	Edit Box
☑	au_lastname	Edit Box

Destination Page

● New Page

○ Existing Page   Select Page...

CatalogUpdate

Select columns that users can update from the update page.

An update button is generated on the detail page to allow for updating the current record when browsing. If you want to control access to the page by other means, you can delete this button from the detail page later. Password protection can also be added to the update page to restrict access. See “Adding password protection to a page” on page 382.

To learn how to create insert and update pages without using the DataForm wizard, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

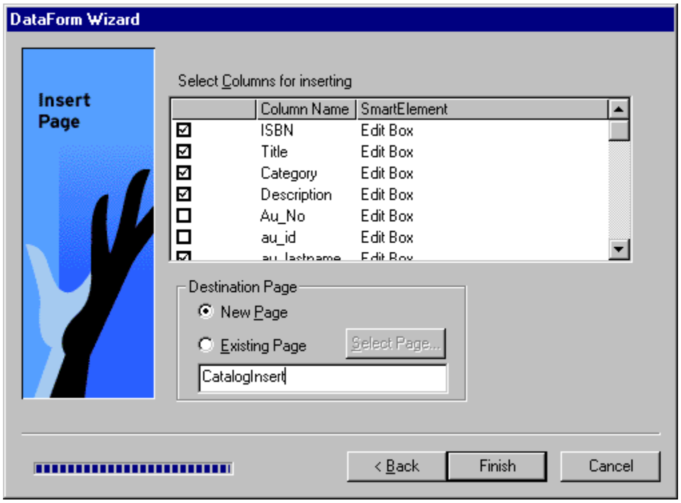
DataForm insert page

The insert page generates a blank form page that can be used to insert new records. An insert button will also be added to the details page if you selected that option in the detail page dialog box of the DataForm wizard.

You can create an insert page by selecting the Insert option in the main screen of the DataForm wizard. You may also add an insert page to a DataForm set you have already generated. See “Previewing or browsing DataForm pages” on page 332.

Choose whether you want to create a new page (rename it as you like in the text box) or use an existing page for the output. Choose the columns into which you want users to be allowed to input information. Insert operations are restricted only to the columns you choose.

If you have an automatic numbering operation for your primary key in the main database application, you should deselect this field for the insert page.



DataForm Wizard

Insert Page

Select Columns for inserting

	Column Name	SmartElement
☑	ISBN	Edit Box
☑	Title	Edit Box
☑	Category	Edit Box
☑	Description	Edit Box
□	Au_No	Edit Box
□	au_id	Edit Box
☑	au_lastname	Edit Box

Destination Page

● New Page

○ Existing Page

CatalogInsert

Choose the columns that are accessible for inserting new records in the insert page (deselect any auto-number fields).

DataForm Delete option

Unlike the other database options, choosing the Delete Records option in the DataForm wizard does not create a new page. It merely adds a button to the detail page so that users browsing the records can choose to delete the current record. To add this feature, select Delete Records in the first dialog box of the DataForm wizard or in the General tab of DataForm Properties dialog box, if you are rerunning the DataForm wizard.

Previewing or browsing DataForm pages

Once the DataForm wizard has completed generating your pages, you should first publish and preview your site to check how the DataForm pages work before beginning your design tasks. (Before you can view your data-driven pages, your Publish Settings must be properly set up. See "Publishing Active Server Pages" on page 189.)

You can view your pages either by using the Preview tab in Drumbeat, or in your browser by clicking the Browse button (choose a browser from the button if you wish). Browse behavior is different for an ASP site, depending on the option you use:

1. If you use the Preview tab to preview your page, Drumbeat will automatically publish the page if it is not current, and then bring up the ASP page that is currently in layout.
2. If you use the Browse button, the default page for the site will load, regardless of the page currently in the layout. Drumbeat does not republish the page if it is not current.

Revising DataForms

The output of the DataForm wizard can be revised later after you have generated your DataForm pages. Revisions may include:

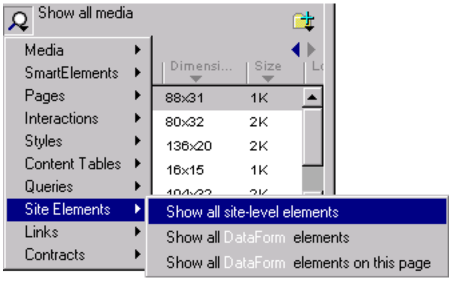
- Adding or deleting DataForm pages.
- Adding new fields from the underlying query.
- Revising the query and rerunning the DataForm wizard to regenerate the output.
- Changing the type of control for any field.

Rerunning the DataForm wizard

The DataForm wizard can be rerun anytime you want to change options on a DataForm page or add new pages to the existing DataForm. If you modify the query the DataForm uses, you must rerun the DataForm wizard to update the pages.

To rerun the DataForm wizard:

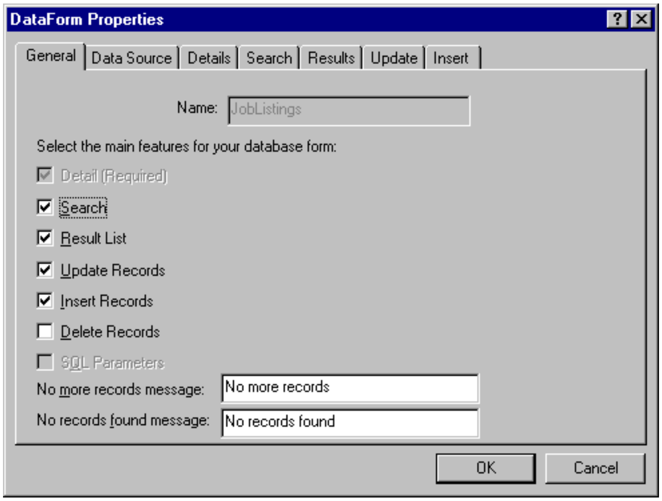
1 In the Asset Center, click the Locate Assets button and choose Site Elements > Show All DataForm Elements.



2 Double-click the DataForm you want to edit, or right-click and choose Properties.

3 If you want to add a new DataForm page, select the page you want to add in the General tab.

4 Select the tab for the page you want to edit or create and choose the options for the page.



Name: JobListings

Select the main features for your database form:

☑ Detail (Required)
☑ Search
☑ Result List
☑ Update Records
☑ Insert Records
□ Delete Records
□ SQL Parameters

No more records message: No more records

No records found message: No records found

Any elements that you may have deleted from your DataForm that are still selected in the DataForm Properties tabs will be regenerated on those pages when you rerun the DataForm wizard. This includes field labels that you may have deleted, as well as form elements that you may have replaced with other form elements. You can simply delete these extra elements from the page.

Adding and deleting DataForm pages

Once you have created a recordset, you can add new database-driven pages and edit existing ones by rerunning the DataForm wizard.

To add a new DataForm page to an existing DataForm:

1 In the Asset Center, click the Locate Assets button and choose Site Elements > Show All DataForm Elements.

2 Double-click the DataForm you want to edit, or right-click and select Properties.

3 Check the page you want to add in the General tab.

4 Select the tab for the page to choose the options for the page.

The options in the DataForm Properties tabs are the same as those in the DataForm wizard. When you click OK, the DataForm pages will be re-created and any new pages you have chosen generated. As you add new DataForm pages to the set, new tabs will be added for each page with the options that can be edited.

When you delete a DataForm page or individual elements from a page manually, the deleted page or elements will be re-created again, unless you first deselect those options in the DataForm Properties dialog boxes.

To delete elements or pages from a DataForm completely:

1 In the Asset Center, click the Locate Assets button and choose Site Elements > Show All DataForm Elements.

2 Double-click the DataForm you want to edit, or right-click and choose Properties.

3 Click the tab for the DataForm page you want to edit and change your options. If you want to delete all the DataForm elements from the page, deselect the page in the General tab.

   When you click OK, the DataForm pages will be re-created and any pages you have deselected will be deleted from the site.

When deleting DataForm pages, you should use the DataForm Properties dialog box, rather than manually deleting pages in Site Manager. That way all elements will be properly removed. If you delete all the pages belonging to a recordset, you should also delete the matching DataForm element in the Asset Center.

To delete the DataForm element from the site:

1 Click the Locate Assets button and choose Site Elements > Show All DataForm Elements.

2 Select the DataForm element, right-click, and choose Delete.

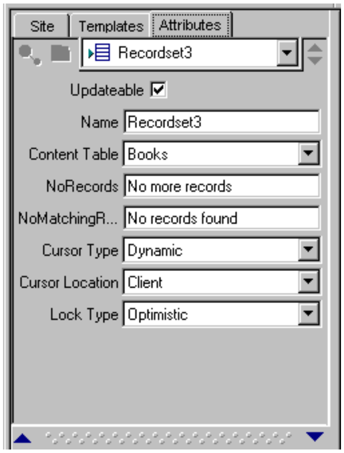
Using recordsets

Each DataForm page contains the recordset element in the Basement, which represents the data source for that page. A recordset element is created for each database query that is used on a page. The recordset itself contains a number of attributes that can be accessed through its Attributes tab.

To access the attributes of the recordset:

Select the recordset and click the Attributes tab, or right-click on the recordset and choose Attributes.

Note: If the recordset is generated by a stored procedure that returns values, then the Basement element will be a Command Object instead of a recordset element. See "The Command object" on page 303 for more information.



Updateable ☑

Name Recordset3

Content Table Books

NoRecords No more records

NoMatchingR... No records found

Cursor Type Dynamic

Cursor Location Client

Lock Type Optimistic

Check the Attributes of the recordset *on a page through the Attributes tab.*

The content of the recordset is derived from the ODBC content table assigned to it. The content table is a visual representation of your SQL query. You can rename the recordset to something more meaningful to suit your naming conventions and make applying interactions and scripting to a recordset easier. (This is especially useful if you are using multiple recordsets on a page.)

Note: If you want to reference any database content on a page, you must first add a recordset element to the Basement of the layout, then bind the recordset to the appropriate content table.

To add a new recordset to a page:

1 Select the recordset SmartElement on the SmartElement toolbar.

2 In the Attributes tab for the recordset, for Content select Content Table from the pop-up menu.

If an existing recordset already contains the data you want to use on a page, you can use that recordset, rather than create an identical, new recordset.

To use an existing recordset on a page:

1 Click the Locate Assets button in the Asset Center and select Show All Site-Level Elements.

2 In the Asset Center list, find the recordset you want to use and drag it to the page.

In this case, you do not need to set up the content in the Attributes tab, since it is already associated with the proper data source.

The attributes of the recordset are described in the following table.

Attribute	Description
Updateable	Specifies whether users are permitted to update the recordset object after it is first created. This includes adding filter strings to recordsets on a page.
Name	Name of the recordset. This can be changed anytime during the design process. All associations on the page will be updated automatically.
Content Table	Specifies the content table, which can also be considered your SQL query results, to be displayed in the site.
NoRecords	Message displayed when user reaches the end of records while browsing.
NoMatchingRecords	Message displayed when user reaches the end of records that match a search criteria while browsing.
Cursor Type	Defines the type of connection made to the database. Options are Dynamic, Forward Only, Keyset, and Static. Default is Dynamic. Dynamic cursors allow users to modify the database records and to navigate forward and backward through the database. Forward-only cursors limit navigation through the database to forward only, which improves performance when you only need to make a single pass through a recordset. Keyset cursors limit the display to a fixed set of records, reflecting updates and deletions to these records but not additions of other records. Static cursors limit the display to a static copy of records that do not reflect additions, changes, or deletions by other users.
Cursor Location	Client or Server: The cursor location attribute determines if the Web server or the database server manages the database cursor. Drumbeat defaults cursor management to Client.
Lock Type	The lock type attribute determines what level or record locking you want to use on your database connection. Options are Optimistic, Pessimistic, or Read-Only. Default is Optimistic. Optimistic locking is available for client or server cursors. The provider will lock the record only when update is called. Pessimistic locking is only a valid option when cursor location is set to Server. The provider will lock the record when any of the values are edited and keep it locked until an update is committed to the database. This guarantees that no one else is updating the same records at the same time. Read-Only locking is available for client or server cursors. If a cursor is Read-Only, updates and inserts will not be supported.

CHAPTER 17
DataForm Layouts and Form Elements

This chapter explains how to customize DataForm layouts that have been generated by the DataForm wizard. It also explains how to use form elements when you want to customize the output and functions of DataForm pages or create your own database interactions using form elements.

The DataForm wizard allows you to assign any valid SmartElement, such as a dropdown list, checkbox, and radio button, to fields on the DataForm pages. However, the output of the wizard may only approximate your final result and you will need to set up the functions properly after the pages have been generated. Other times you may want to add new form elements to DataForm pages manually without rerunning the DataForm wizard for the whole set of pages, particularly if you have already done a lot of design work on your pages that you do not want to risk disturbing. You may also want to create form pages manually to include special functions not possible in the wizard.

Working with DataForm layouts

When you finish running DataForm wizard your pages are ready to be designed as you wish. You can move elements, delete them, add graphics and styles, and design as you do with any other page.

Some quick initial fixes you may want to make to your DataForm page layouts are explained in the following sections.

Use SmartSpacer to allow for variable-size fields

If the database you are using to generate your DataForm pages includes long fields, such as description fields, elements may overlap one another as you browse through the records. You can use SmartSpacer to fix this quickly and easily and allow for variable widths on different record pages.

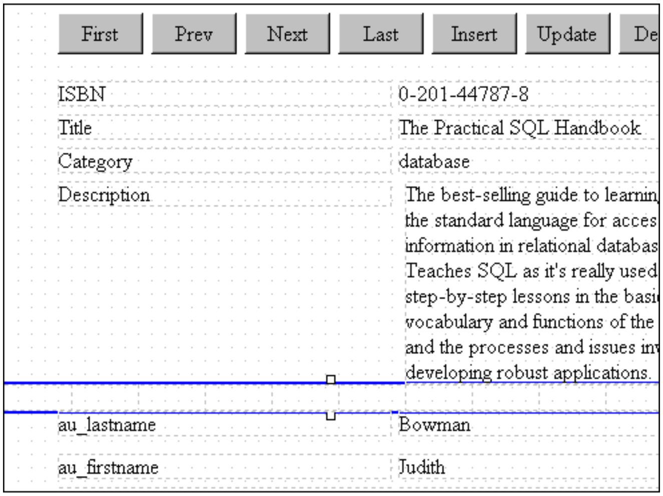
Note the following when using SmartSpacer:

- SmartSpacer can only be used on sites targeting browsers versions 4.0 and later.
- On pages targeting any 4.0 browsers, SmartSpacer cannot be used between form elements on a page. (Form elements include submit buttons, radio buttons, checkboxes, list boxes, and edit boxes.) If you want to use SmartSpacer on a page with several form elements, rearrange your page in such a way that all of the form elements are either above or below the SmartSpacer. (This restriction does not apply to pages designed exclusively for Internet Explorer 4.0.)

SmartSpacer cannot be used within a data loop. You may use SmartSpacer before or after the looper element on the page.

To use SmartSpacer to allow for variable-size fields:

Drag the SmartSpacer element from the toolbar onto your page and drop it at any point within the horizontal area of the display of the field.



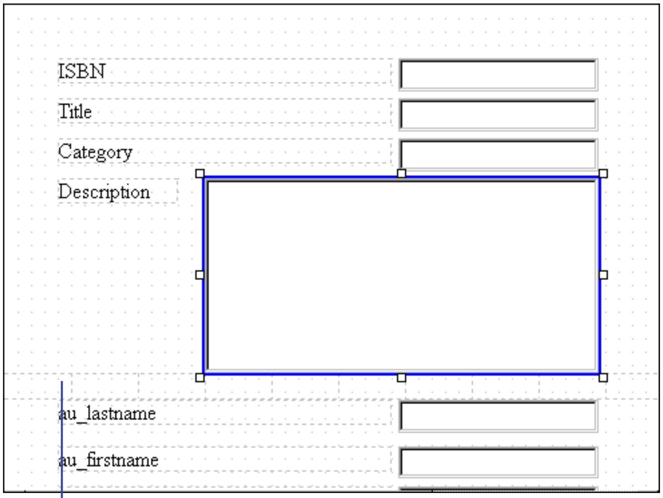
ISBN	0-201-44787-8
Title	The Practical SQL Handbook
Category	database
Description	The best-selling guide to learnin the standard language for acces information in relational databas Teaches SQL as it's really used step-by-step lessons in the basi vocabulary and functions of the and the processes and issues inv developing robust applications.
au_lastname	Bowman
au_firstname	Judith

Create a multi-line edit box

Update and insert pages commonly contain edit boxes for entering text. You can expand an edit box to allow for multiple line input.

To create multi-line edit boxes:

1. Select the edit box on the layout and click the Attributes tab. In the Attributes tab check MultiLine.
2. Drag the SmartSpacer element below the edit box. The SmartSpacer will be added only temporarily to make it easy to automatically position elements below it on the page. (If the target browser is any 4.0 browser, the SmartSpacer will appear red, with a warning that you cannot use it between form elements, but because you are only using it as a temporary spacing device that will be deleted later, don't worry about it.)
3. Use the handles on the edit box to scale it. The fields below the SmartSpacer will automatically adjust to allow for the new size.
4. Delete the SmartSpacer, which is no longer necessary.



SmartSpacer

Using media files from a database

Media files, including images, audio and video files may be associated with database records and be referenced by URL in a database. To use media files from a database, the database column must contain the path to the file. This may be either the full path or the relative path from the root publish directory.

If you use the DataForm wizard to generate the page, you can choose the appropriate SmartElement for the column. If you are referencing an image, for example, choose the image element. If you are adding the media file to a page you have already created or to a page you are creating manually, add the appropriate SmartElement and then bind the content of the SmartElement to the recordset and the URL column in the database in the Attributes tab.

To use a media file referenced by URL on a DataForm page:

1 Drop the appropriate SmartElement on the page.

2 In the Attributes tab, for Content, select Recordset.

3 In the Content dialog box, choose the content table and the appropriate column that includes the image path.

4 Set any other attributes for the SmartElement in the Attributes tab.

When you publish the site, you must post the media files to the appropriate directory on the server yourself, as Drumbeat cannot publish content that is not native to Drumbeat (in this case, referenced from an external database file).

Setting tab order

On form pages it is often desirable for users to be able to tab from field to field while entering data. Tab order can be set only for browsing in Internet Explorer 4, as Netscape Navigator and most other browsers will only tab fields left-to-right, bottom-to top. However, you can set the tab order on pages targeting any browser and users with Internet Explorer 4 can take advantage of this feature.

To set the tab order of the form fields:

1 Right-click an empty area in the layout of the page and select Set Tab Order from the pop-up menu, or select Arrange > Set Tab Order. Numbers will appear next to each element on the page.

2 Click on the element you want to be first in the tab order. Then click on successive elements to assign tab order in the order you click them. Continue until you have gone through all the elements for which you want to set tab order. You can increment the number of any item by one by clicking it again.

3 Once you have set the tab order of all the elements, click on the background to return the page to normal layout mode.

Linking with recordsets

Database fields can be linked to an associated URL if you set up the URL column in your database and provide the URL for each record.

To create a link based on a URL in a database:

1 Select the anchor element on the layout that you want to hyperlink.

2 In the Attributes tab, click the Link Assign button. Or right-click the selected element and choose Assign Link.

3 In the Link dialog box, select the recordset in the left pane (Link Targets). In the right pane (Column Names), select the column that contains the URL.

For a demonstration of how to use the recordset link type and other methods for creating dynamic links, see the Drumbeat Support Center (macromedia.com/ support/drumbeat).

Customizing form buttons

The standard form buttons that DataForm wizard creates can be customized in the following ways:

- Add a style to the button that may include font choices, background and foreground text colors and/or borders (4.0 browsers only).
- Replace the form buttons with image buttons.
- Make smart navigation buttons that do not appear when the indicated action is unnecessary or not an option.

Using styles with form buttons

An easy way to customize form buttons is to apply a style that includes background or border colors and custom fonts. You can create a style especially for your form buttons in the Style Builder dialog box.

To create a custom button style:

1 Click the Add Assets button in the Asset Center and select Styles or select Tools > Styles.

2 Click New to create a new style.

3 Click the Background tab. Select Fixed, and then click the Color bar and choose a color for your buttons.

4 Click the Text tab, select the font to use on the buttons and a color that will stand out against your background color.

5 To add a border to the buttons, click the Box tab. Select Fixed for Box Color and click the color bar to choose a color for the border. Choose a style from the pop-up menu (Solid). Set the width of the border in the Border box.

6 Click OK, and then name and save your new style.

7 Select the form buttons on the layout and apply the new style from the Style pop-up menu in the Attributes tab.

Replacing form buttons with images

All of the form buttons that the DataForm wizard creates can be replaced with image buttons if you wish. After substituting the buttons, you must set up the appropriate interactions with the database. These are all standard point-and-click interactions that you can find in Database Navigation category.

Note: The interactions used by the DataForm wizard are server-side, which means that they are supported in all browsers. Image buttons use client-side JavaScript, so will be supported in JavaScript-compatible browsers only. This means they will not work in pages targeted at Generic browsers.

Before replacing your form buttons, you might want to check the navigation interaction that is currently assigned to that button, as a guide to duplicating it on your image buttons.

To see the current interactions on a form button:

Select the form button you want to replace. Right-click and choose Assigned Interactions (ordered).

The current interaction assigned to that button will be displayed in the Interactions Center. You should apply the same interaction to the new image button.

To replace a form button with an image button:

1 Drag the Image Button SmartElement from the SmartElements toolbar to the layout.

2 Drag an image from the asset center or from a content table cell onto the image button on the layout. (To use multiple rollover images, you can drag a second image onto the first. for more information on this, see “Creating a rollover image” on page 161.)

3 Select the recordset in the Basement, the new image button, and any other participants in the interaction. If you are replacing the back or more button on the result list page, the AutoTable is also be a participant. On a page with a data loop, the looper is a participant. Right-click one of the selected elements and select Possible Interactions.

4 In the Interactions Center, find the appropriate interaction in the Database Navigation category. Double-click the interaction, or right-click and choose Apply. Set the required parameters, if any, in the Parameters dialog box.

   After you have replaced the buttons on a form page with images, you need to set up the submit action between the form and each button, so the form will be submitted when the button is clicked. Standard form buttons are automatically of the Submit type so do not need this interaction. However, image buttons cannot be type Submit so you need to apply the interaction.

5 Drag a hidden form element from the SmartElements toolbar into the Basement. (Both a hidden form element and the new form element will now appear in the Basement.)

6 Select the form element in the Basement and the image button. Right-click and choose Possible Interactions. In the Interactions Center, apply the following interaction from the Forms Submit category:

   Submit [Form] when [ImageButton] is clicked.

7 In the Parameters dialog box, click OK to accept the default action.

The Form Element will always appear in the Basement if there are any form elements on the page. If you are using a recordset on the page but do not have any form elements (for instance, if you are using a recordset to populate a text box), you need to place a hidden form element on the page to make a form element appear. You can then set up a submit action as above.

Note: The method on the form element in the Basement must be set to Post, which is the default method.

For a working example of how to use images as submit buttons, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

The following table shows all the default interactions for the form buttons generated by the DataForm wizard. Documentation for the Database Navigation contracts can be found in the Contracts folder of the Drumbeat application folder.

Form Button	DataForm Page	Participants	Interaction
First	Detail	Button + Recordset	Go to the first record in [target] when source is clicked.
Previous	Detail	Button + Recordset	Go to the previous record in [target] when source is clicked.
Next	Detail	Button + Recordset	Go to the next record in [target] when source is clicked.
Last	Detail	Button + Recordset	Go to the last record in [target] when source is clicked.
Insert	Detail	Button + Recordset	Go to pagename when [source] is clicked, maintaining state of [target].
Update	Detail	Button + Recordset	Go to pagename when [source] is clicked, maintaining state of [target].
Find	Search	Button + Edit Box + Recordset	Filter [target] using [columnValue] with comparisonOp when [source] is clicked.
Sort	Results List	Button + Recordset	Sort [target] on [columnName] when [source] is clicked.
Back	Results List	Button + Recordset	Set/move [target] direction backward when [source] is clicked.
More	Results List	Button + Recordset	Set/move [target] direction forward when [source] is clicked.
Update	Update	Button + Recordset	Update active record when [source] is clicked.
Back	Update	Button + Recordset	Go to previous pagename when [source] is clicked, maintaining state of [recordset].
Insert	Insert	Button + Recordset	Insert record into [target] when [source] is clicked.
Back	Insert	Button + Recordset	Go to previous pagename when [source] is clicked, maintaining state of [recordset].

Making smart navigation buttons

If the user reaches the end of the records, normally a message is displayed saying that there are no more records to browse. This message may be customized in the DataForm wizard or in the Attributes tab for the recordset (see "Using recordsets" on page 335). However, another option would be not to have the previous or next button display if the current record is the first or last record in the recordset.

You can make the navigation buttons smart by applying an interaction that detects the position in the recordset and conceals the appropriate button. Several interactions are available for this purpose in the Database Miscellaneous category, as shown in the following table. Full documentation for these contracts can be found in the Contracts folder of the Drumbeat application folder.

Interactions to conceal form buttons on DataForm pages in the Database Miscellaneous category.

Form Buttons	DataForm Page	Interaction
First or Previous	Detail	Do not create [source] if [target] is on the first record.
Last or Next	Detail	Do not create [source] if [target] is on the last record.
Next	Detail with Looper	Do not create [source] if [target] is on first set of records in the looper.
Previous	Detail with Looper	Do not create [source] if [target] is on last set of records in the looper.
Back	Result List (AutoTable)	Do not create [source] if [table] is on the first page of [target] records.
More	Result List (AutoTable)	Do not create [source] if [table] is on the last page of [target] records.

To conceal the next navigation button for the last record in the set:

1 Select the next navigation button in the layout and the recordset in the basement. Right-click and choose Possible Interactions.

2 In the Interactions Center, find the following interaction in the Database Miscellaneous category: Do Not Create [Recordset1_Next] if [Recordset1] Is on the Last Record. Double-click the interaction to apply it, or right-click and choose Apply.

To conceal the previous navigation button for the first record in the set:

1. Select the previous navigation button in the layout and the recordset in the Basement. Right-click and choose Possible Interactions.
2. In the Interactions Center, find the following interaction in the Database Miscellaneous category:

   Do Not Create [Recordset1_Previous] if [Recordset1] Is on the First Record. Double-click the interaction to apply it, or right-click and choose Apply.

When a data loop is involved, you must use the alternate contract: Do Not Create [Recordset1_Previous] if [Recordset1] Is on the First Set of Records in the Looper.

In the Parameters dialog box, for PageSize select the number of records displayed in the looper (the number of records per page), and select the button position on the page (above or below the looper).

To enforce even more stringent control, you can apply the contract Do Not Allow Navigation Past the End of the Recordset to the Recordset.

The navigation buttons for back and more that appear on the result list page and apply to the AutoTable display require a different contract. These buttons typically retrieve a group of results, which is limited to the number of records specified in the page size in the Attributes tab for the AutoTable.

To conceal the back or more button on the result list page when there are no more records:

1. Select the more button, the AutoTable and the recordset. Right-click and select Possible Interactions.
2. In the Interactions Center, find and apply one of the following interactions in the Database Miscellaneous category:

   Do Not Create [Source] if [Table] Is on the Last Page Of [Target] Records, or Do Not Create [Source] if [Table] Is on the First Page of [Target] Records.
3. In the Parameters dialog box, select the number of records displayed at any one time (this should match the Page Size attributes in the Attributes tab for the AutoTable).

For a working example of how to show and hide navigation buttons on a database page, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

Using form elements

All of the form elements that the DataForm wizard uses in DataForm pages are, by default, available on the SmartElements toolbar. You can add or substitute form elements on a DataForm page, or build form pages from scratch if you want custom features. Most of the time, even if you want to do extensive customization, you will find that working with an initial output from the DataForm wizard will be the most efficient way to accomplish what you want. But you may find that you want to add or change elements after a fair amount of design work has been done on the page. In that case, you will want to add form elements manually.

Assigning the content of a form element

When you add a new form element to a page, you must choose the content of that form element in the Attributes tab. The content of a form element can come from four possible sources:

- Recordset delivers content from a database query. In the Content dialog box, choose the recordset and the column. If the form element is participating in an insert or search page function, or is not otherwise connected with a current record, select Submit Only.
- Static Content delivers the value from a cell in a non-ODBC content table. In the Content dialog box, choose the content table, the row, and column.
- Text assigns a text string as content for the box. Click the Content (ellipses) button and type the text in the Text editor. If you want the text to be inserted into a database field as part of the submit action, you can set up an insert/update interaction with the text box, the submit button, and the recordset.
- Server Expression uses a value passed from the server. In the Content dialog box, enter the server expression that retrieves the content from the server. Since the content is only rendered at browse time when the server is accessed, you can also enter design text, which will appear on your layout for design purposes.

Using recordset content in form elements

When building ASP applications, the content for form elements most often will come from a database query. Each query has its associated ODBC content table, which gives you a visual representation of the data that you can refer to in your design. The recordset element is then used to represent this database content on the page. Each recordset must be associated with an ODBC content table.

It is important to keep in mind the order of the relationship of the SQL query, ODBC content table, and the recordset.

When you run the DataForm wizard, you choose the ODBC content table (or an existing recordset already associated with a content table) to use for the DataForm pages that the wizard generates. The wizard creates a recordset that is then associated with the form elements on your pages. If you want to use content from a different recordset on your DataForm pages, you must first add the second recordset to the page. Likewise, if you want to build your own database pages manually, you must first add a recordset to the page and bind it to the appropriate ODBC content table. You can use an existing recordset or create a new one.

Using an existing recordset that already references the appropriate data source is more efficient than creating multiple recordsets that reference the same query and content table. (Although you can generate more than one content table from the same query, and you can create more than one recordset for the same content table, it is not very efficient to do so.)

To add a new recordset to a page:

1 Drag the recordset SmartElement from the SmartElements toolbar to the Basement (or drag it to the page and it will automatically appear in the Basement).

2 In the Attributes tab for the recordset, for Content choose the content table from the list.

To use an existing recordset on a page:

1 Click the Locate Assets button in the Asset Center and choose Show All Site-level Elements.

2 In the Asset Center list, find the recordset you want to use and drag it to the Basement (or drag it to the page and it will appear in the basement).

In this case, you do not need to set up the content in the Attributes tab, since it is already associated with the data source.

Edit boxes

New edit boxes may be added to an existing form to represent new fields you may want to add to the form. Use new edit boxes for these common reasons:

- If you have edited a query to include new fields and do not want to re-run DataForm wizard to add it to the page.
- If you want to add a column from a different recordset.
- If you are building an insert or update page manually.

To add an edit box whose content will be a column from a recordset:

1. Drag the Edit Box SmartElement from the SmartElements toolbar and to the layout.
2. In the Attributes tab for the edit box, for Content, select Recordset.
3. In the Content dialog box, select the recordset and the column that contains the data you want.
4. If the edit box is participating in an insert or search page function, or is not otherwise connected with a current record, select Submit Only. This will bind the element to the proper column in the database, but will not display a field value when the page is loaded.

Setting up the update/insert interaction for edit boxes

If the content is coming from a recordset, the recordset must be one already displayed in the Basement. If not, you will have to add it. See “Using recordset content in form elements” on page 350.

If you are adding the edit box to an existing insert or update page generated by the DataForm wizard, all the necessary submit actions will be set up for you once the content is bound to the proper column in the recordset.

If you are building an insert or update page by hand, you need to set up the insert/update interaction between the edit box, the submit button, and the recordset to handle what happens when the form is submitted. This is also necessary if the content of the edit box is coming from a different recordset or the content is other than a recordset. Ssee “Assigning the content of a form element” on page 349.

To set up the insert/update interaction:

1 Select the edit box, the submit button and the recordset. Right-click and choose Possible Interactions.

2 In the Interactions Center, find the following interaction in the Database Edit category:

   Use [Edit1] Data to Update/Insert in [Recordset] When [Button] is Clicked.

   (The bracketed names will reflect the names of the elements on your page.) Double-click the interaction to apply it, or right-click and choose Apply.

3 In the Parameters dialog box, the column in the database into which the value should be inserted.

   Make sure that the insert/update interaction is the last interaction to occur with the recordset. See “Ordering elements and interactions” on page 373.

4 Select the recordset. Right-click and choose Order Interactions.

5 In the Interactions Center, find one of the following interactions in the Database Edit category:

   Insert Record into [Recordset1] When [Insert_Button] is Clicked, or Update Active Record When [Update_Button] is Clicked.

6 Select the interaction and drag it to the last item in the list.

Checkboxes

Checkboxes can be used as form elements to facilitate entry into a column that is a boolean field (Yes/No or True/False) in a database. They can also be used to pass these values to a script. When you use a checkbox for a field you must be sure that the corresponding database column is the appropriate field type.

To add a checkbox whose content will refer to a recordset to a DataForm page:

☑

1. Drag the Checkbox SmartElement from the SmartElements toolbar to the layout.
2. In the Attributes tab for the checkbox, for Content choose Recordset.
3. In the Content dialog box, choose the recordset and the column that is the true/false field from which you want it to obtain its value.
4. If the checkbox is participating in an insert or search page function, or is not otherwise connected with a current record, check Submit Only. This will bind the element to the proper column in the database, but will not display the field value when the page is loaded, so it will appear as a blank checkbox in the browser.

Setting up the insert/update interaction for checkboxes

If the content is coming from a recordset, the recordset must be one already displayed in the Basement. If not, you will have to add it. See “Using recordset content in form elements” on page 350.

If you are adding the checkbox to an existing insert or update page generated by the DataForm wizard, all the necessary submit actions will be set up for you once the content is bound to the proper column in the recordset.

If you are building an insert or update page by hand, you need to set up the insert/update interaction between the checkbox, the submit button, and the recordset to handle what happens when the form is submitted. This is also necessary if the content of the edit box is coming from a different recordset or the content is other than a recordset (see “Assigning the content of a form element” on page 349).

To set up the insert/update interaction:

1. Select the checkbox, the submit button and the recordset. Right-click and choose Possible Interactions.
2. In the Interactions Center, find the following interaction in the Database Edit category:

   Use [Check1] Data to Update/Insert in [Recordset] When [Button] is Clicked.

   (The bracketed names will reflect the names of the elements on your page.) Double-click the interaction to apply it, or right-click and choose Apply.
3. In the Parameters dialog box, choose the column in the database into which the value should be inserted.

   Make sure that the insert/update interaction is the last interaction to occur with the recordset. See “Ordering elements and interactions” on page 373.
4. Select the recordset. Right-click and choose Order Interactions.
5. In the Interactions Center, find one of the following interactions in the Database Edit category:

   Insert Record into [Recordset1] When [Insert_Button] is Clicked, or Update Active Record When [Update_Button] is Clicked.
6. Select the interaction and drag it to the last item in the list.

Radio buttons

Radio buttons can be used to represent a limited number of choices for a database field. They can be used to insert the selected value into the database or to pass a value to a script.

When you select a radio button for a field in the DataForm wizard only one radio button is generated on the DataForm page for the field. You will have to add as many additional radio buttons as you need for the number of selections you want. Alternatively, you can replace an edit box generated by the wizard with a set of radio buttons.

Radio buttons that you add to the page will automatically have the default group name Group1. You can change the group name in the Attributes tab and assign buttons to different groups if necessary—which you will need to do if you have more than one set of radio buttons. All radio buttons belong to a group and only one radio button in a group can be selected at any one time. The selected radio button is the one whose value either comes from or is passed to the database or script. One button in the group should be selected by default.

To add a set of radio buttons whose content will refer to a recordset column to a DataForm:

1. Select the Lock tool on the SmartElement toolbar or double-click the Radio Button SmartElement to put it in lock mode. Click the layout as many times as you need to create the number of radio buttons you want on the layout. Click the Lock tool again to unlock.
2. In the Attributes tab for each radio button, enter the value that will be passed to the database or script if this button is selected. This may be a numerical value or a text string.
3. For Selection, choose Recordset, if you want the selection to come from the database.
4. Specify the recordset and column in the Content dialog box.
5. If the radio button is participating in an insert or search page function, or is not otherwise connected with a current record, check Submit Only. This will bind the element to the proper column in the database. It will appear unselected in the browser page, unless the button has been designated the default selected button.
6. For Group Name in the Attributes tab, enter the name of the group to which the radio button belongs.
7. For Value, enter the value to insert into the database if the radio button is selected.

To specify which radio button in a group is selected by default:

In the Attributes tab for the default selected radio button, choose Selected.

Setting up the insert/update interaction for radio buttons

If the content is coming from a recordset, the recordset must be one already displayed in the Basement. If not, you will have to add it. See “Using recordset content in form elements” on page 350.

If you are adding the radio buttons to an existing insert or update page generated by the DataForm wizard, all the necessary submit actions will be set up for you once the content is bound to the proper column in the recordset.

If you are building an insert or update page by hand, you need to set up the insert/update interaction between the radio buttons, the submit button, and the recordset to handle what happens when the form is submitted. This is also necessary if the content of the edit box is coming from a different recordset or the content is other than a recordset. See “Assigning the content of a form element” on page 349.

To set up the insert/update interaction:

1 Select one radio button, the submit button and the recordset in the Basement. Right-click and choose Possible Interactions.

2 In the Interactions Center, find the following interaction in the Database Edit category:

   Use [Radio1] Data to Update/Insert in [Recordset1] When [Recordset1_Update] is Clicked.

   (The bracketed names will reflect the names of the elements on your page.) Double-click the interaction or right-click and choose Apply.

3 In the Parameters dialog box, enter the radio button’s group name (which should match the group name you entered in the Attributes tab) and choose the column in the database into which the data should be inserted.

   Make sure that the submit interaction is the last interaction to occur with the recordset. See “Ordering elements and interactions” on page 373.

4 Select the recordset. Right-click and choose Ordered Interactions.

5 In the Interactions Center, find one of the following interaction in the Database Edit category:

   Insert Record into [Recordset1] When [Insert_Button] is Clicked, or Update Active Record When [Update_Button] is Clicked.

6 Select the interaction and drag it to the last item in the list.

List boxes

List boxes can be of two types in Drumbeat: a dropdown list or an open list box. Both types of SmartElements for these form elements can be found on the SmartElement toolbar, by default. The list box SmartElement allows multiple selections and can be expanded to any size. A dropdown list SmartElement cannot be expanded and allows for only one selection to be visible. Otherwise the two elements act the same.

When you choose a list element for a field in your DataForm pages in the DataForm wizard, the resulting list often does not contain the actual content you envision for the list box. The options you want in your list box may be in another table in the database, or you may want to type in your own list of options. You can change the options for a list box by accessing the attributes of the list box in the Attributes tab.

When you add a list box to the page manually, you need to set all of the attributes for the list box yourself.

To add a list box to the page:

1 Drag the List Box element or the Dropdown List element from the SmartElements toolbar to the layout.

2 In the Attributes tab, choose the content, selection, and value and set any other attributes. See the following sections for specific instructions on using content from different sources.

- Content is the actual, visible content of the dropdown list, from which the user can choose an option.
- Selection is the default selected option when the page loads (should correspond to the record currently being displayed).
- Value is the value of the user's selection. This value is used when the form is submitted. The value is then inserted into the database or is the value passed to a scripted action.

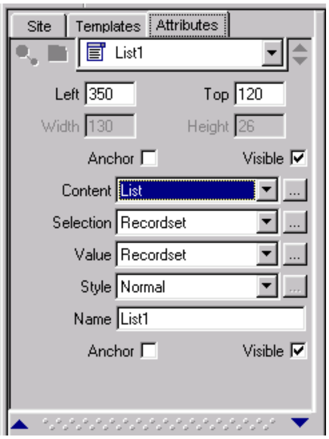
Creating a Manual List

The content for your dropdown list can come from a list that you enter by hand. These list selection can be inserted directly into the database, or a corresponding value for each list item may be inserted. This helps eliminate random entries or spelling errors that might make search operations on the field difficult.

To create list options manually for a dropdown list or list box:

1 Select the list box on the layout and click the Attributes tab.

2 In the Attributes tab, for Content select List.



Left 350 Top 120
Width 130 Height 26
Anchor □ Visible ☑
Content List
Selection Recordset
Value Recordset
Style Normal
Name List1
Anchor □ Visible ☑

3 In the List Content dialog box, select Add New Item and type the first option. Press Enter to enter a second item. Continue until you have made your list, and then click OK.

4 If the list box is participating in a database interaction, and the values you want passed to the database are different than the items in the content list, you must also provide these values. In the Attributes tab, for Value, select List and enter the corresponding values for each item in the content list (in the same order). The value attribute will interpret the user's selection and enter it into the database.

Note: If you do not enter anything in the Value content box, the values will default to the same as those in the Content list. The list box will appear empty in the layout, but will be populated with the correct content at browse time. If you want the content to appear on the layout, enter the same list items for Values as for Content.

5 For Selection, choose Recordset. In the Content dialog box, select the recordset and the column that displays the selected option. When the page loads in the browser, the value in the recordset matching the list value will be selected by default.

6 If the list box is participating in an Insert or Search page function, or is not otherwise connected with a current record, check Submit Only (don't get content from Recordset). If the list box is on an update page, leave this unchecked, as you want the page to display the value for the current record from the recordset.

You can edit the list of options by clicking the ellipses button next to the Content option in the Attributes tab. Remember to duplicate any edits to the content list in the value list.

Setting up the insert/update interaction for list boxes

If the selection is set to the recordset you intend to insert into, you do not need to set up an insert/update interaction. All the necessary interactions will be set up for you once the content is bound to the proper column in the recordset.

However, if the manual list box is not bound to a recordset, or you are building a page manually, you need to set up the insert/update interaction that will tell the recordset to use this field's value when inserting or updating the recordset. The interaction will be between the list box, the recordset, and the submit button.

To set up the insert/update interaction:

1 Select the list box, the submit button and the recordset in the Basement (the one you are submitting into). Right-click and choose Possible Interactions.

2 In the Interactions Center, find the following Interaction in the Database Edit category:

   Use [List1] Data to Update/Insert in [Recordset] When [Button] is Clicked.

   (The bracketed names will reflect the names of the elements on your page.) Double-click the interaction or right-click and choose Apply.

3 In the Parameters dialog box, choose the column that you want to update/ insert.

   Make sure that the insert/update interaction is the last interaction to occur with the recordset. See "Ordering elements and interactions" on page 373.

4 Select the Recordset. Right-click and choose Ordered Interactions.

5 In the Interactions Center, find the one of the following interactions in the Database Edit category:

   Insert Record into [Recordset1] When [Insert_Button] is Clicked, or Update Active Record When [Update_Button] is Clicked.

6 Select the interaction and drag it to the last item in the list.

Retrieving an option list from a separate (non-related) table

The content for your list can come from a column in another recordset that is not directly related to the recordset used in the DataForm wizard to generate the page. For example, the recordset you used to create your DataForm may have a column named ReferralType. You want users to be able to select options from a list and you have another database table which contains these options. You can use this column for your list options, even though the two recordsets are not related by a common field.

To create an option list for a dropdown list from a different recordset:

1 Add to the page the recordset that contains the content you want to display in the list. See "Using recordset content in form elements" on page 350.

2 Select the list box on the layout. In the Attributes tab for the list box, for Content select recordset2 (or the name of your second recordset) and the column that contains the items you want to display in the list.

3 Define where the list gets its value with the Value attribute. Select Recordset from the dropdown list and in the Content dialog box, select recordset2 and the column that contains the items you want for the list. (In this case, the value will be the same as the content, so if you did not define the value it would default to the same as the content.)

4 For Selection, choose Recordset. In the Content dialog box, select recordset1 (the original recordset) and the column that displays the selected option. This will bind the list to the original recordset, allowing automatic filtering, inserting, or updating.

5 If the list box is participating in an insert or search page function, or is not otherwise connected with a current record, check Submit Only. If the list box is on an update page, leave this unchecked, as you want the page to display the value for the current record from the recordset.

If the selection is set to the recordset you intend to insert into, you do not need to set up an insert/update interaction. All the interactions will be set up for you once the content is bound to the proper column in the recordset.

If the list box selection is not bound to a recordset, or you are building a page manually, you need to set up an interaction between the list box, the recordset, and the submit button.

To set up the inert/update interaction:

1 Select the list box, the submit button, and the recordset in the basement (the one you are submitting into). Right-click and choose Possible Interactions.

2 In the Interactions Center, find the following Interaction in the Database Edit category:

   Use [List1] Data to Update/Insert in [Recordset1] When [Recordset1_Update] is Clicked.

   (The bracketed names will reflect the names of the elements on your page.) Double-click the interaction or right-click and choose Apply.

3 In the Parameters dialog box, select the column that you want to update/insert.

   Make sure that the insert/update interaction is the last interaction to occur with the recordset. See “Ordering elements and interactions” on page 373.

4 Select the Recordset. Right-click and choose Ordered Interactions.

5 In the Interactions Center, find one of the following interactions in the Database Edit category:

   Insert Record into [Recordset1] When [Insert_Button] is Clicked, or Update Active Record When [Update_Button] is Clicked.

6 Select the interaction and drag it to the last item in the list.

For an example of how to create a lookup list from a different recordset to filter an existing recordset, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

Retrieving an option list from a related table

The content for your list can be a related field that is referenced numerically or by code from a field in the initial recordset used to generate the page in the DataForm wizard. For example, the main recordset in your DataForm may have a column named Dept. The content of this field may be a number that relates to a record in another table which contains both the DeptID and DeptName.

To create an option list from a related table:

1. Add to the page the recordset that contains the content you want to display in the list. See "Using recordset content in form elements" on page 350.
2. Select the list box on the layout. In the Attributes tab for the list box, for Content select recordset2 (or the name of the second recordset). Click the Content ellipses button and in the Content dialog box, choose recordset2 and the Column that has the text values you want for the list.
3. Define where the record gets its value with the Value attribute. For Value choose Recordset. In the Content dialog box, select recordset2 and the column that relates to the second recordset.
4. For Selection, choose Recordset. In the Content dialog box, select recordset1 (or the name of the original recordset) and the numerical or code column.
5. If the list box is participating in an insert or search page function, or is not otherwise connected with a current record, check Submit Only. If the list box is on an update page, leave this deselected, as you want the page to display the value for the current record from the recordset.

If the Selection is set to the recordset you intend to insert into, you do not need to set up an insert/update interaction. All the necessary interactions will be set up for you once the content is bound to the proper column in the recordset.

If the list box is not bound to a recordset, or you are building a page manually, you need to set up an insert/update interaction between the list box, the recordset, and the submit button.

To set up the insert/update interaction:

1 Select the list box, the submit button and the recordset in the Basement. Right-click and choose Possible Interactions.

2 In the Interactions Center, find the following Interaction in the Database Edit category:

   Use [List1] Data to Update/Insert in [Recordset1] When [Recordset1_Update] is Clicked.

   (The bracketed names will reflect the names of the elements on your page.) Double-click the interaction or right-click and choose Apply.

3 In the Parameters dialog box, select the column that you want to update/insert. Make sure that the insert/update interaction is the last interaction to occur with the recordset. See "Ordering elements and interactions" on page 373.

4 Select the recordset. Right-click and choose Ordered Interactions.

5 In the Interactions Center, find one of the following interactions in the Database Edit category:

   Insert Record into [Recordset1] When [Insert_Button] is Clicked, or Update Active Record When [Update_Button] is Clicked.

6 Select the interaction and drag it to the last item in the list.

For an example of how to add a static prompt or blank value option to a dropdown list populated by a recordset, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

Database editing interactions

These interactions allow you to insert, update, and delete records from a recordset. In addition, there are interactions that allow insertion of values from selection form-elements, such as radio buttons, checkboxes, and list elements.

Note: The use of these Interactions ultimately change the recordset's underlying database. It is advisable to use a backup copy of your database when designing and testing pages that use these contracts.

For more detailed information on individual contracts, see the individual contract descriptions, which are found on the Description tab of the Contract Properties dialog box Tools > Contracts. Documentation for the contracts can be found in the Contracts folder of the Drumbeat application folder.

Form Participants	DataForm Page	Interaction
Submit Button Recordset	Update	Delete active record when [source] is clicked. Deletes the active record.
Submit button Recordset	Update	Delete active record when [source] is clicked with confirmation. Deletes the active record after the user confirms the delete.
Submit Button Recordset	Insert	Insert record into [target] when [source] is clicked. Inserts a new record into the recordset. All form elements bound to the recordset will be inserted.
Submit Button Recordset	Insert	Insert record into [target] when [source] is clicked with maintain state option. Inserts a new record into the recordset. All form elements bound to the recordset will be inserted. The Parameters dialog box allows you to maintain the original position of the recordset before the record was inserted. If you deselect the Maintain Original State option, the recordset position will be the record that was last inserted.
Submit Button Recordset	Update	Update active record when [source] is clicked. Updates the recordset's active record. All form elements bound to the recordset will be inserted.

Form Participants	DataForm Page	Interaction
Submit Button Recordset List box, dropdown list	Insert Update	Use [ListElement] data to update/insert in [target] when [source] is clicked Inserts/updates a recordset with a list element's data. This Interaction inserts the list element's value attribute. Used when the list element is not bound to the target recordset (as when the list gets lookup values from another recordset). Values will not be reflected in the recordset unless an accompanying insert or update contract is applied between the button and recordset.
Submit Button Recordset Radio Button	Insert Update	Use [Radiobutton] data to update/insert in [target] when [source] is clicked. Inserts/updates a recordset with radio button data. Used when the radio buttons are not bound to the target recordset. Note that this contract should only be applied to one member of the radio button group. The underlying HTML references radio buttons according to their group, but Drumbeat does not have a group element. Since the contract needs participant requirements, the participant must be a radio button, but applying the contract to multiple radio button of a given group will cause an error. Values will not be reflected in the recordset unless an accompanying insert or update contract is applied between the button and recordset.
Submit Button Recordset Check Box	Insert Update	Use [Checkbox] data to update/insert in [target] when [source] is clicked. Inserts/updates a recordset with check box values. Used when the check boxes are not bound to the target recordset. Values will not be reflected in the recordset unless an accompanying insert or update contract is applied between the button and recordset. Note that this contract is intended for inserting the value of the checkbox as opposed to the state. The check box value is set in the content attribute. To insert the checkbox state in a true/false data field, don't use this contract; rather, bind the checkbox to the recordset directly.

Form Participants	DataForm Page	Interaction
Submit Button Recordset Edit box	Insert Update	Use [EditBox] data to update/insert in [target] when [source] is clicked. Inserts/updates a recordset with a edit box data. Used when the edit box is not bound to the target recordset (as when the list gets a value from another source such as a server expression). Values will not be reflected in the recordset unless an accompanying insert or update contract is applied between the button and recordset.
Recordset JSPServer - Session or Application basement SmartElement button - any clickable object	Insert	Insert record in [Recordset] and store AutoNumberID in [JSPSession] variable. Allows you to obtain an Autonumber ID from the database when a new record is inserted. The Autonumber IS is placed in an JSPServer object (typically a Session variable). Note that the method for retrieving the Autonumber ID requires a field in your database for capturing a timestamp.

Note: Many of these Interactions are used by the DataForm wizard and should not be modified.

Using a data loop

When you run the DataForm wizard you can choose to display more than one record per page by choosing this option for the Detail page. This creates a data loop that displays as many records on the page as you choose.

You can create a data loop on a page manually with the looper SmartElement. The looper SmartElement displays multiple records from a recordset on a single page. The number of records used in the loop is determined by the Display Number attribute in the Attributes tab for the looper.

To create a data loop on a page manually using a recordset:

1. In the Asset Center, click the Locate Assets button and select SmartElements.
2. Drag the Looper from the SmartElement toolbar and drag it to the layout. Two horizontal bars will appear on the layout to define the looper area.
3. In the Attributes tab for the looper, define the content. Choose Recordset and in the Content dialog box, select the recordset.

   If you are using a recordset, the recordset you want to use must already be on the page. If not, you will have to add it manually. See “Using recordset content in form elements” on page 350.
4. In the Attributes tab, choose the number of records you want to display. If you want to display all the records, enter 0.
5. Place any form elements you want to generate within the confines of the Looper and bind them to the appropriate recordset column.

To create a data loop on a page using a server expression:

1. Place the looper on the layout. With the looper selected, click the Attributes tab or right-click and choose Attributes.
2. For Content, select Server Expression. In the Content dialog box, enter the expression. Enter any design text you want to be used on the layout for design purposes (since the actual content will not appear until you browse the page).
3. In the Attributes tab, choose the number of records you want to display. If you want to display all the records, enter 0.
4. Place any form elements you want to generate within the confines of the looper and bind them to the appropriate recordset column.

For example, if you want to display the output of a simple counter via a server expression, first, declare a server-side global variable counter.

To declare the server-side global variable counter:

1 Open the script view.

2 Expand the global variables node.

3 Double click the server object.

4 Enter the text: var counter = 1.

5 Set the looper content to Server Expression.

6 In the Server Expression dialog box expression field enter:

```
counter++
```

7 This increments the counter every time the looper loops.

To display the value of the counter within the looper, you can add a text container inside which has content server expression.

To set the server expression on the text container:

1 Choose Content Server Expression.

2 Enter the following text in the Expression field:

```
counter
```

This will print the counter value to the screen in each loop.

Whatever you put inside the server expression associated with a looper will be executed in each iteration of the looper. You may place other elements besides recordset elements in the looper area. Only one representative item is displayed on the layout within the looper. However, when you publish, all items within the looper will repeat as many times as the Display Number attribute specifies.

To bring up the Attributes tab for the Looper:

Click one of the two broken lines on the page that define the looper area and click the Attributes tab. Or, with the page in layout, select the looper by name in the list in the Attributes tab.

Using SmartSpacer with the looper

If your page design includes any design elements placed below the looper, you will need to place a SmartSpacer below the looper component to adjust their page positions appropriately.

SmartSpacer cannot be place inside the range of the Looper, as this functionality is not supported by the data loop component.

Using insert/update buttons with a data loop

To use insert/update buttons with records in a data loop you need to set up the button links with URL parameters that will position the database cursor on the correct record in the recordset.

The URL parameter option of the Link dialog box is used to position the cursor of the recordset in the correct position using a technique called a keyed value lookup of the recordset. The value of the action variable is set to Find. This action forces the recordset to find a particular record in the recordset. The find method requires two parameters: 1) the name of the column that you are going to use as your look up column and 2) the value to look up.

To set up a link on an update button in a data loop, while maintaining state of the record:

1. Place the button you want to link on the page (an image or image button will work equally well).
2. Right-click the button and choose Assign Link.
3. In the Link dialog box, select the update page for the link destination.
4. In the URL Parameters box, enter this code:

   ```
   Recordset_Action=Find('COLID','<%=Recordset.GetColumnValue("COLID")%>')
   ```

For Recordset substitute the name of your the recordset you are using and for COLID substitute the name of the unique key column.

The above example uses JavaScript syntax. If you have your server scripting language set to VBScript, you will have to use a different syntax, as follows:

```
Recordset_Action=Find('COLID','<%=Recordset.GetColumnValue(""COLID"")%>')
```

Note: This example works as long as there is no other server-side contract applied to the button. The URL parameter code must not contain the “=” sign if another server-side contract is applied to the button. For an example of this, see the Search Engine starting point in the Drumbeat starting points folder. This example uses an update button in a data loop, with the server-side Do Not Create interaction applied for security.

18

CHAPTER 18
Form and Database Interactions

Drumbeat offers a wide range of standard and custom interactions you can apply to DataForm pages or use in building custom database pages to improve searching and filtering operations and customize output. This chapter covers just a few of the most common or popular interactions you may want to use, including a complete list of the database filter interactions. Keep in mind the possibilities are infinite for extending database interactions with your own scripted interactions and custom components.

Form and database contracts

Contracts are pre-written JavaScript functions. The effects of contracts are known as activations when applied to one SmartElement and as interactions when applied to two or more SmartElements.

A number of contracts included in Drumbeat are written especially for the most common types of form and database interactions. You can filter for these contracts by both category and by target browser in the Contract Properties dialog box.

Full documentation for all of the contracts that ship with Drumbeat can be found in the Contracts folder of the Drumbeat application folder.

In addition to the built-in contracts, you can write new contracts or alter existing ones to create custom contracts you may need. For more information on writing and adapting contracts, see “Interactions Center” on page 397. The following sections introduce some common form and database interactions you may want to use.

Viewing and editing interactions

After you have assigned a number of interactions on a page, you can check the interactions that have been assigned on any one element or on the whole page.

To view all the interactions on the page, in order:

Right-click an empty area on the layout and choose Assigned Interactions (ordered).

To view all the interactions in which a single element participates, in order:

Select the element, right-click and choose Assigned Interactions (ordered). You can remove or edit the parameters of any interaction in the list.

To remove an interaction:

Right-click the interaction in the Interactions Center and choose Remove.

To edit the parameters of an interaction:

Double-click the interaction in the Interactions Center, or right-click and choose Edit.

Ordering elements and interactions

When customizing your DataForm pages, it is quite possible that you will end up with a number of interactions on a single page. Much of the time, the order in which the interactions occurs is of little consequence. However, sometimes it can make an important difference. In addition, the order in which elements appear in the Basement may make a difference when they are participating in specific interactions. These basement elements include recordsets, hidden form elements, cookies, and session objects.

Ordering interactions

Drumbeat orders interactions in the order in which you assign them. When dealing with the order of interactions, you should be aware that there are two kinds of interactions for an element:

- Interactions in which the element is a participant.
- Interactions in which the element actually contributes script when an event occurs.

When an element is merely a participant, but does not contribute script to an interaction, the order is unimportant. If an element contributes script to an interaction, it can be considered an active participant and the order in which the script occurs on the page can be significant. You can both view and change the order of the interactions in which an element is an active participant.

To view the order of the interactions for which an element is an active participant:

Select the element, right-click and choose Order Interactions.

To change the order of interactions on an element:

1 Select the element, right-click and choose Order Interactions.

2 Select the interaction you want to move in the Interactions Center and drag it to the desired position in the ordered list.

One of the most common uses of this is in dealing with submit functions, since it is important that all interactions must occur before the submit action takes place. Therefore, after you have applied your interactions on the page, you should check the order of the interactions on the submit button. (This does not apply to form buttons of type submit, but it does apply to a form button of type button or to an image button with the Submit Form When Clicked contract assigned to it).The Submit action should always be the last action to occur on the submit button. This is the one you will most often have to drag to the last position in the list or ordered interactions.

Ordering recordsets and Basement objects

When you start customizing your DataForms it is highly likely that you will want to add new recordsets to a page that already contains one or more recordsets, or new Basement elements such as hidden form elements and cookies. Much of the time, the order in which the recordsets or other Basement objects load in the browser is inconsequential. However, there are occasions when the order of Basement elements can be important—such as when loading a cookie that contains login information, retrieving a server expression stored in a hidden form element, filtering one recordset by the value in another recordset, or when calling certain types of stored procedures.

The order in which recordsets are loaded is the order in which they are displayed, left to right, in the Basement. The leftmost element will be the first to load, followed by each element in order.

To reorder recordsets or other Basement elements on a page:

Drag the element to the position you want.

Security and validations

Various types of security safeguards and form validations can be applied to DataForm pages. The standard interactions that ship with Drumbeat can be found in the Security category and the Forms Validation category in the Interactions center or in the Contract Properties dialog box.

Security contracts

This set of contracts allows you to set up site level access privileges and to create a new user. They can be divided into two sub-categories: redirect on access privilege failure and login. Redirect on access privilege failure contracts take a comma-separated privilege group list for the page. If the site visitor does not belong to one of the groups, access is denied for the current page and may be redirected to a different page if necessary. Login contracts can be used to create new users with unique id's, and to allow access to specific pages in the site based on the login details.

Form Participants	Interaction
The page	Place client-side password protection on this page. Forces a user to input a password as defined when the contract is created and redirects to another page if the password is incorrect (light security).
Recordset Image or form button Username edit box Password edit box Form in basement	Verify username from [userbox] and password from [passbox] and set access group. Allows session-based security where a user inputs a username and password, and they are compared against a database. If the user is registered in the database a session variable is created that is equal to the value of a database column that contains access privileges for that user. You can then restrict access to pages on your site based upon that Session variable.
The page	Redirect if user does not have permission based on access group. Redirects the user if the access group session variable does not match the one set up by the security logon page previously.
Recordset Image or form button Username edit box Password edit box Form in basement	Verify username from [userbox] and password from [passbox] and redirect (store additional variable in Session). Allows session-based security where a site visitor inputs a username and password, and they are compared against a database. If the site visitor is registered in the database a session variable is created that is equal to the value of a database column that contains access privileges for that user. You can then restrict access to pages on your site based upon that session variable.

Form Participants	Interaction
Recordset Image or form button	Update [username] and [password] into [source] if [username] is unique in [source].
Username edit box Password edit box Form in basement	Checks the database to make sure that username is unique, and updates the username and password into the active record. Otherwise it redirects to a failure page, and sets the failed request variable to false.
The page	Do not create [source] if user does not have permission based on access group.
	Redirects the user if the access group session variable does not match the one set up by the security logon page previously.
Recordset Image or form button	Insert [username] and [password] into [source] if [username] is unique in [source].
Username edit box Password edit box Form in basement	Checks the database to make sure that username is unique, and inserts the username and password into a new row in the recordset. Otherwise it redirects to a failure page, and sets the failed request variable to false.
Recordset Image or form button	Insert [username] and [password] into [source] if [username] is unique in [source] and define Access Group.
Username edit box Password edit box Form in basement	Checks the database to make sure that username is unique, and inserts the username and password into a new row in the recordset. Otherwise it redirects to a failure page, and sets the failed request variable to false.
Recordset Image or form button	Update [username] and [password] into [source] if [username] is unique in [source] and define Access Group.
Username edit box Password edit box Form in basement	Checks the database to make sure that username is unique, and updates the username and password into the active record. Otherwise it redirects to a failure page, and sets the failed request variable to false.

Adding form validations

A common requirement on a form page is to ensure that users enter the correct type of data for a field. A number of validation contracts are included in the Interactions Center for this purpose.

Note: A list of the validations contracts is at the end of this section.

To set up validations on a form field:

1. First apply an interaction to the field to be validated
2. Then apply the Validate on Submit action to the form.

To set up a date validation on a field:

1. Select the edit box on the layout that is a date field. Right-click and choose Possible Activations.
2. In the Interactions Center, find the Date Validation interaction. Double-click to apply it to the field or right-click and choose Apply.
3. In the Parameters dialog box, choose the type of date validation you want.

To set up a non-blank validation:

1. Select the field on the layout. Right-click and choose Possible Activations.
2. In the Interactions Center, find the interaction Non-Blank Validation on [Source]. Double-click to apply it to the field, or right-click and choose Apply.
3. In the Parameters dialog box, enter the error message to display if the field is left blank.

Setting validation when a form is submitted

After setting up the validation on the field, you must also set up the submit action to execute the validation. The simplest case is a form that has a single submit button. In a slightly more complex case, you may have multiple submit buttons on the page which perform different actions (such as an update page which contains both an update and delete button). In this case you only want the validation to be performed on one button only (the update button). In both cases, make sure that the submit action is the last action on the button. (The logical order of events is that the form must first validate, then submit.)

To set up the validation on submit, when using standard form buttons generated by the DataForm wizard or the Form Button SmartElement:

1. Select the form element in the Basement, right-click and choose Possible Activations.
2. In the Interactions Center, apply the interaction Validate When [Form] is Submitted.

To set up the validation on submit, when using an image button or text element to submit the form:

1. Select the image (or text) element and the form element. Apply the interaction Submit [Target] When [Source] is Clicked.
2. In the Parameters dialog box deselect Validate Form (as the form validation will already have been applied to the form element) or remove the validation on the form element and select this option. Only one validate action should be applied.

To set up the validation on submit on a form with more than one submit button:

1. Select the field to validate on the layout and the submit button (or image button) on which the validation should occur. Right-click and choose Possible Activations.
2. In the Interactions Center, apply the interaction Submit [Target] When [Source] is Clicked.
3. In the Parameters dialog box, select Validate Form.

To re-order the interactions so the submit action is last:

1 Select the submit button, right-click and choose Order Interactions.

2 In the Interactions Center select the interaction (Submit [Form] When [Button] is Clicked) and drag it to the last position.

All validations applied to the page will be performed when the form is submitted.

See also the Drumbeat Support Center (macromedia.com/support/drumbeat) for a series of articles on form validation methods.

Form validation contracts

The form validation contracts are used to validate data entered by a site visitor in an edit box. The contracts that specify how to validate a control don't actually do anything by themselves. They must be combined with one of the contracts that execute the validations that have been applied. Most typically this is the contract Validate When [Source] is Submitted applied to the form element in the Basement. When the form is submitted the applied validations will execute.

Form Participants	Interaction
Edit box Form in Basement	Show alert if [source] is identical to [target] when [form] is submitted. Allows you to validate the data by comparing it to the data in another field.
Edit box Edit Box Form in Basement	Show alert if [source] is not identical to [target] when [form] is submitted. Allows you to validate the data by comparing it to the data in another field.
Edit box	Credit Card Number Validation on [source]. Ignores spaces, dashes, periods, and carriage returns. Valid if all remaining characters are digits and the entire number passes a Mod 10 check (a mathematical formula where the value of the final digit is known when given the first part of the number).
Edit box	Custom Validation on [source]. This contract allows you to validate the data by comparing it to a format that you supply. (See Description tab in the Contract Properties dialog box for details.)

Form Participants	Interaction
Edit box	Date Validation on [source].
	Valid if the user enters a valid date as recognized by the JavaScript Date() function. This function allows a wide range of date entries. It also allows seemingly invalid dates such as 1/32/98. This date is interpreted as 2/1/98. Different browsers implement the JavaScript Date() function a little differently in terms of what it will accept as valid. In all browsers if the date is accepted by the Date() function it is reformatted and written back to the edit control.
Edit box	Entry Length Validation on [source].
	Allows you to put limitations on the number of characters entered by the user.
Edit box	E-mail Address Validation on [source].
	Valid if the value contains the @ character.
Edit box	Fixed Point Validation on [source].
	Limits the number of digits the user can enter after the decimal place. This contract is useful if you want to validate a money value by setting the Decimal Places parameter to 2
Edit box	Floating Point Validation on [source].
	Valid if the user enters a floating point value. A floating point value is a number with a decimal place such as 1.23 or 0.1. Any floating point number that is recognized by the JavaScript function parseFloat() is valid.
Edit box	Huge Integer Validation on [source].
	Valid if all characters are digits and an optional leading dash (minus sign). Spaces are removed. This is different than integer validation in that it accepts larger integers than the Integer Validation contract.
Edit box	Integer Validation on [source].
	Valid if all characters are digits. Leading and trailing spaces are removed and leading zeros are removed. The validated integer is written back to the control for submission to the server.
Edit box	International Phone Number Validation on [source].
	Ignores spaces, pluses, dashes, carriage returns, parentheses, slashes, underscores, and asterisks. Valid if all remaining characters are digits and there are at least 6 digits.

Form Participants	Interaction
Edit box	Non Blank Validation on [source]. Ignores spaces and carriage returns. Valid if there are any remaining characters.
Edit box	Social Security Number Validation on [source]. Ignores spaces, dashes and periods. Valid if the remaining characters are 9 digits. The value will be reformatted to ###-##-####.
Edit box	Time Validation on [source]. Accepts any time value that is accepted by the JavaScript time (and date) functions.
Edit box	URL Validation on [source]. Valid if a valid URL is entered.
Edit box	US Phone Number Validation on [source]. Ignores spaces, dashes, carriage returns, parentheses, slashes, underscores, and asterisks. Valid if all remaining characters are 7 digits, 10 digits, or 10 digits preceded by a 1. The value will be reformatted to one of the following: 1(###) ###-#### (###) ###-#### ###-####
Dropdown list	Validate a selection was made in [source]. Valid a selection other than the first entry was made in the list. This validation is used in dropdown lists where the first element is something like Make a Selection, or Choose One.
Edit box	Zip Code Validation on [source]. Ignores dashes, spaces, and carriage returns. Valid if remaining characters are 5 digits or 9 digits. The value will be reformatted to #####-#### or #####.
Form in basement	Validate when [source] is submitted. Validates all controls that have validation activations applied prior to submission. If any fail an error report is generated, displayed to the user, and the form is not submitted.

Form Participants	Interaction
Edit box with a validation activation applied	Validate [source] when it loses focus. Validate the individual control when it loses focus. An error message is generated if the control fails to validate. A message will not be generated if the edit box is empty.
Edit box	Validate [Source] text equals [Target] text. Ignores spaces, dashes, carriage returns, parentheses, slashes, underscores, and asterisks. Valid if all remaining characters are 7 digits, 10 digits, or 10 digits preceded by a 1. The value will be reformatted to one of the following: 1(###) ###-#### (###) ###-#### ###-####

Adding password protection to a page

A simple client-side password protection can be added to any page by applying the interaction for passwording a page. First, you may want to create a page to which users will be directed if they try to access the protected page without first logging on into the application. The page can contain an appropriate message or new login screen.

To apply password protection to a page:

1 Right-click an empty area on the layout of the page and choose Possible Activations.

2 In the Interactions Center find the following interaction in the Security category: Place Client-side Password Protection on This Page. Double-click the interaction, or right-click and choose Apply.

3 In the Parameters dialog box, enter the password for the page and select the name of the page to which you want to redirect users if the password fails.

A more elaborate password protection that checks for user groups and passwords against a database is illustrated and described in the User Login Registration starting point. See the documentation for this starting point in the Starting Points folder of the Drumbeat application folder.

Customizing searches

When you run the DataForm wizard, the options for the search page allow you to choose more than one field on which to base a search. You can choose to have separate edit boxes or dropdown lists for each search field or include multiple search fields in one edit box (as you might if you wanted to search several fields for a given keyword). However, there are a number of other ways you might want to expand the search capabilities, which you can do manually on your page.

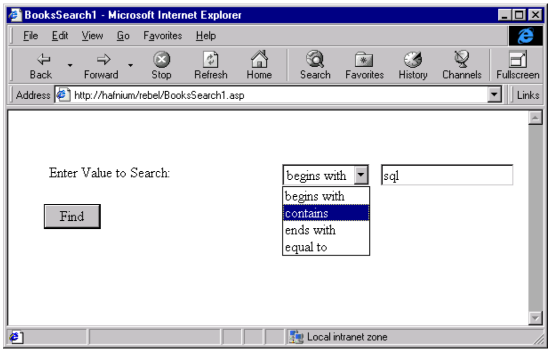
When you create a search page with the DataForm wizard, the page contains filter contracts specifically designed for the DataForm wizard. The DataForm wizard filter contracts allow you to only use one operator for the page. You might want to use different operators on each field, in which case, you must delete the DataForm wizard filter and apply the standard ones. This section discusses the search functions you can create in the DataForm wizard and then explains how to set up manual search functions.

Setting up a search page with the DataForm wizard

The DataForm wizard allows you to control the search method when searching a database field (or fields) for a matching string.

If you choose a single edit box for searching, you can also include a search type dropdown that gives added search flexibility. For example, you can choose a single edit box so that users can enter a keyword (or words) and search both the title and description fields of a book catalog.

If you did not choose this search option when you ran the DataForm wizard, you can select this option in the DataForm Properties dialog box and run the wizard again to create it (see “Rerunning the DataForm wizard” on page 332). The resulting search type drop-down created by the wizard is shown in the following illustration.


BooksSearch1 - Microsoft Internet Explorer
File Edit View Go Favorites Help
Back Forward Stop Refresh Home Search Favorites History Channels Fullscreen
Address http://hafnium/rebel/BooksSearch1.asp
Links
Enter Value to Search:
begins with
sql
begins with
contains
ends with
equal to
Find
Local intranet zone

Creating search functions with multiple search criteria

When you run the DataForm wizard, you can choose multiple fields on which to conduct a search. You can choose to have two or more fields searched with a single edit box, as in the previous example, or you can choose separate form elements for different fields. If you want to search on more than two fields, you should choose separate form elements.

The book catalog example in the previous section can be built manually to include expanded search options that allow users to search on a category, with a dropdown list of categories supplied from a related table.

To add a new search category to the search page:

1 Add a Listbox SmartElement to the page.

2 Place a recordsets onto the page and bind the recordset to the content table that contains the list content.

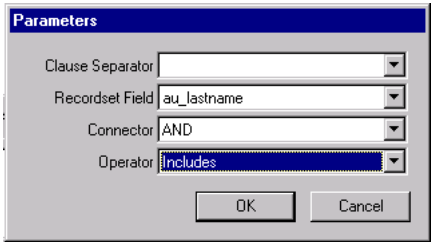
3 In the Attributes tab for the listbox, for Content select Recordset and the column that contains the text options for the list box (CategoryName).

4 For Value, select the column that contains the numerical field that is referenced in the first recordset (CategoryID).

5 Select the listbox, the submit button and the first recordsets (the one you want to search and filter). Right-click and choose Possible Interactions. Apply the following interaction in the Database Filter category:

Add [Value in] Dropdown List Value to Filter of [Recordset] When [Button] is Clicked

6 In the Parameters dialog box, enter the field you want to search, the connector (AND/OR search criteria) and the operator.



Parameters

Clause Separator

Recordset Field au_lastname

Connector AND

Operator Includes

7 Check the assigned interactions on the page (right-click an empty area in the layout and choose Order Interactions) and make sure that the last interaction is the one applied to the submit button: Filter [Target] When [Source] is Clicked.

For an example of how to create client-side drill-down lists for searching a database, to learn how to display the number of records returned by a search, and how to display record numbers during navigation, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

Filtering data

Filters are used to reduce the number of records displayed in the recordset to show only the records that meet certain criteria. Full documentation for these contracts can be found in the Contracts folder of the Drumbeat application folder.

Some of these contracts are in no way related to the DataForm wizard contracts and should not be used interchangeably with the database filter contracts that are automatically applied by the DataForm wizard. The DataForm wizard contracts are denoted by *(DFW)* following the interaction phrase.

Form Participants	Contract Filter Type	Interaction
Submit Button Recordset	DataFormwizard	Filter [target] when [source] is clicked (DFW). Creates a filter for the given recordset when the submit button is clicked. The filter is constructed based on all of the form elements that are bound to the recordset.
Submit button Recordset	DataForm wizard	Filter [target] using [columnValue] with [comparisionOp] when [source] is clicked (DFW). Creates a filter for a column in the given recordset when the submit button is clicked. This page is submitted to itself and then redirected to the masterpage.
Submit Button Recordset	DataForm wizard	Filter [target] using [columnValue] with comparisionOp when [source] is clicked (DFW). Creates a filter for a column in the given recordset when the submit button is clicked. This page is submitted to itself and then redirected to the masterpage.
The recordset that is filtered by the value of a related field in [target] The recordset that contains a field that is used to filter [source].	Manual search	Filter [target] with related field from [source]. Lets you filter one recordset based on the value of a certain field of another recordset on the same page. Note that the [source] must be to the left of the [target] in the basement.

Form Participants	Contract Filter Type	Interaction
Recordset Edit box Image or form button	Manual search	Add [value in] edit box to filter of [Recordset] when [Button] is clicked Used to build a filter on a recordset. This is a server-side contract (which means that site visitors with browsers that do not support JavaScript can still browse this page).
Recordset Dropdown list whose "value" will be submitted Image or form button	Manual search	Add [value in] dropdown list value to filter of [Recordset] when [Button] is clicked. Used to build a filter on a Recordset. This is a server-side contract.
Recordset Check box Image or form button	Manual search	Add [value in] check box to filter of [Recordset] when [Button] is clicked. Used to build a filter on a recordset. This is a server-side contract.
Recordset Submit button	Manual search	Filter [Recordset] when [source] button is clicked. Used to build a filter on a recordset. This is a server-side contract. Make sure that the action of the form on this page is either blank, or this page's name
Recordset	Manual search	Filter [Recordset] with filter string on server before page loads. Used to filter a recordset with the user-supplied filter string on the server before the page loads. Knowledge of how to create SQL WHERE clauses is useful when using this contract
Recordset Session or Application Object in the basement	Manual search	Remember Filter of [Recordset] in [JSPServer] variable. This interaction remembers the filter that has been placed on the recordset. It does this by using the Session or Application Object. If you have a link from your detail page to your result list page, the filter that was put on the result list will be reapplied when you follow that link. Without this contract, the filter would not be applied and the user would see all of the records in the recordset. This contract is meant to be applied to the recordsets on the result list page and the detail page.

Form Participants	Contract Filter Type	Interaction
Recordset List box Image or form button	Manual search	Filter [Recordset] with [value in] list value when [Button] is clicked (Multiple Values). Filters a recordset when a button or image is clicked.If you want to use an image as a button, you must use a contract to submit the form to this page itself (which is the default behavior), when it is clicked.
Recordset Image or form button Application or Session object in the basement	Manual search	Add value in [ASPServer] variable to the filter of [Recordset] when [Button] is clicked Used to build a filter on a recordset. This is a server-side contract.
Recordset	Manual search	Filter [Recordset] with [ASPServer] variable on server before page loads. Used to filter a recordset with the user-supplied filter string on the server before the page loads. Knowledge of how to create SQL WHERE clauses is useful when using this contract.

Redirecting when filter has no results

When a search filter returns no results, the results list page generated by the DataForm wizard displays an empty table. It is often desirable, instead, to redirect the user to another page with an appropriate message and a chance to do a new search.

To redirect users when a search has no results:

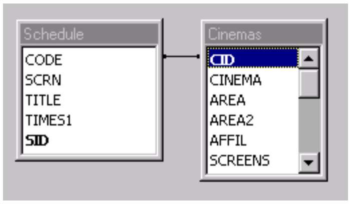
1. On the page intended to display search results, select the recordset from which the results come.
2. Right-click and select Possible Activations. In the Interactions Center, find and apply the following contract in the Database navigation category Redirect If Recordset Is Empty.
3. In the Parameters dialog box, enter the page to which you want to redirect users.

For a working example of how to set up a redirect page when a search has no results, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

Relating two recordsets

Sometimes you may want to filter the results from one set of data by the selection in another set of data. Cross-referencing data from two different DataForms is a fairly easy process, once you envision exactly what it is you want to accomplish.

The database tables shown in the following illustration can be used to create a simple example. One table holds a list of cinemas, while the other shows a list of movie schedules. The CID field in the Cinemas table is related to the CODE field in the Schedule table.



These two tables are related by the linking field shown.

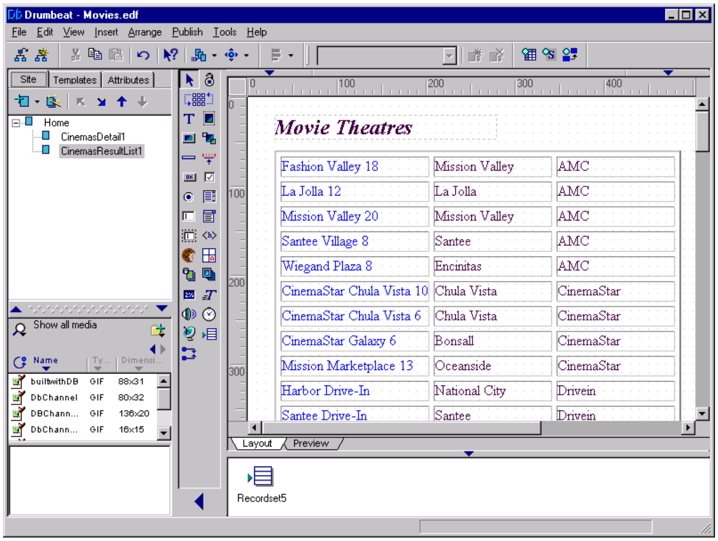
The two sets of data can be related to make one schedule output page from two sets of DataForms, with the second output filtered appropriately.

To relate the two recordsets in one DataForm:

1 Create a query for each table. Create content tables for each query.

2 Use the DataForm wizard to generate a detail page and a results list page for the first query.

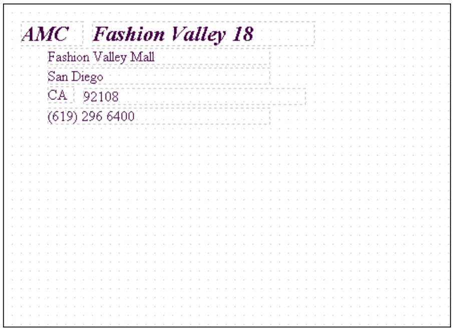
In this example, the CinemaResultsList1 contains just a list of cinemas, with the name linked to the detail page for each, as shown.



Movie Theatres

Fashion Valley 18	Mission Valley	AMC
La Jolla 12	La Jolla	AMC
Mission Valley 20	Mission Valley	AMC
Santee Village 8	Santee	AMC
Wiegand Plaza 8	Encinitas	AMC
CinemaStar Chula Vista 10	Chula Vista	CinemaStar
CinemaStar Chula Vista 6	Chula Vista	CinemaStar
CinemaStar Galaxy 6	Bonsall	CinemaStar
Mission Marketplace 13	Oceanside	CinemaStar
Harbor Drive-In	National City	Drivein
Santee Drive-In	Santee	Drivein

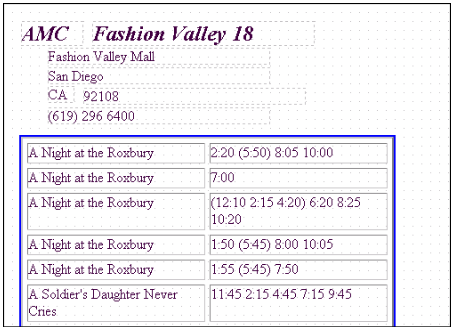
The CinemasDetail1 page contains the name and address for the each cinema, as shown.



AMC Fashion Valley 18
Fashion Valley Mall
San Diego
CA 92108
(619) 296 6400

3 Use the DataForm wizard to generate a detail and a results list page for the second query. For the results list page, select the Existing Page option and select the detail page of the first DataForm.

In this example, the results list for the movie schedule query is created on the detail page of the cinemas query. The results list contains just the title and times of the movies. (The schedule detail page is irrelevant to the operation, but is generated automatically.)



AMC Fashion Valley 18

Fashion Valley Mall
San Diego
CA 92108
(619) 296 6400

A Night at the Roxbury	2:20 (5:50) 8:05 10:00
A Night at the Roxbury	7:00
A Night at the Roxbury	(12:10 2:15 4:20) 6:20 8:25 10:20
A Night at the Roxbury	1:50 (5:45) 8:00 10:05
A Night at the Roxbury	1:55 (5:45) 7:50
A Soldier's Daughter Never Cries	11:45 2:15 4:45 7:15 9:45

At this point, there are detail pages for each cinema on which all of the movie schedules in the database are displayed (or up to the limit set for Records/Page for the results list AutoTable). This list has to be filtered so that it displays only the movies playing at the current cinema. This is done by setting up a filter interaction between the two recordsets.

To filter one recordset by another recordset on the page:

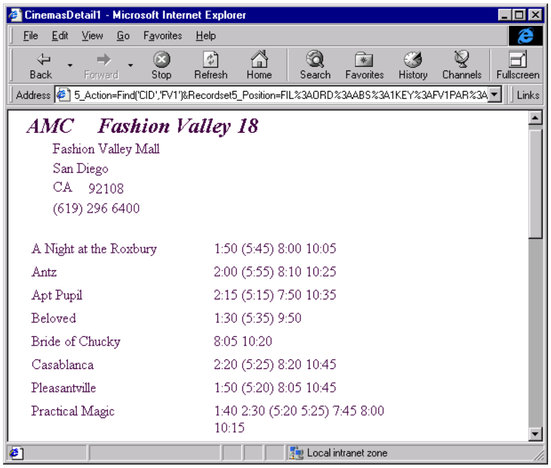
1 Select the two recordsets in the Basement. Right-click and choose Possible Interactions.

2 In the Interactions Center, find the following interaction:

   Filter [Recordset2] With Related Field From [Recordset1]. Double-click the interaction, or right-click and choose Apply.

3 In the Parameters dialog box, select the Filter Column from Recordset2 (CODE) and the Filter By column from Recordset1 (CID).

When you publish and browse the pages, the records on each cinema page are filtered to show movies playing only at that cinema.



5_Action=Find('CID','FV1')&Recordset5_Position=FIL%3AORD%3AABS%3A1KEY%3AFV1PAR%3A

AMC Fashion Valley 18

Fashion Valley Mall
San Diego
CA 92108
(619) 296 6400

A Night at the Roxbury	1:50 (5:45) 8:00 10:05
Antz	2:00 (5:55) 8:10 10:25
Apt Pupil	2:15 (5:15) 7:50 10:35
Beloved	1:30 (5:35) 9:50
Bride of Chucky	8:05 10:20
Casablanca	2:20 (5:25) 8:20 10:45
Pleasantville	1:50 (5:20) 8:05 10:45
Practical Magic	1:40 2:30 (5:20 5:25) 7:45 8:00 10:15

The order in which recordsets are applied to the page can be important when filtering one recordset by another. The order of the recordsets on the page is the order displayed in the Basement, starting from the left. See “Ordering recordsets and Basement objects” on page 374.

For another working example of filtering one recordset with another, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

Filtering data with SQL parameters

If you create queries with SQL parameters you can use them on a page to dynamically filter the data. Dynamic SQL queries can dramatically reduce access time over conventional filter operations when working with large databases, as filter operations can work on small subsets of data. See “Creating dynamic queries with SQL parameters” on page 302.

The following exercise is an example of using a SQL query on a page to create a search result list that can be dynamically refiltered. It uses the movie site example established in the previous section.

The first step is to create the query you want to use and add a parameter for the query. A parameter called @area has been added to the SQL statement that selects for cinemas. The SQL statement looks like this:

```
SELECT * FROM CINEMAS WHERE AREA = @area
```

To set up a page to use this query dynamically:

1 Drag a recordset element from the SmartElements toolbar to the Basement of the page. In the Attributes tab, name it FindCinsRS and for Content choose the FindCinemas content table.

2 Place an AutoTable SmartElement on the page. In the Attributes tab, for Content, choose Recordset, and select FindCinsRS. Choose the columns to display: CINEMA and AREA.

3 Place a dropdown list box on the page. For the content of the list use another small query that gives a list of areas of town.

4 Drag a new recordset element to the basement. In the Attributes tab, name it Area and assign its content to the Areas content table.

5 Select the dropdown list box and the FindCinsRS, right-click and choose Possible Interactions. Find and apply the following interaction in the SQL Parameters category: Pass AreaList to FindCinsRS Using [SQLParameter].

6 In the Parameters dialog box, for enclosing token, choose ' (the single quote). For SQL Parameter, choose the @area parameter.

To set up the necessary action to make the table contents change dynamically when a different selection is made from the list, create a script in the Script Center.

To create the script:

1 Right-click FindCinsRS in the basement and choose Edit Script > Server.

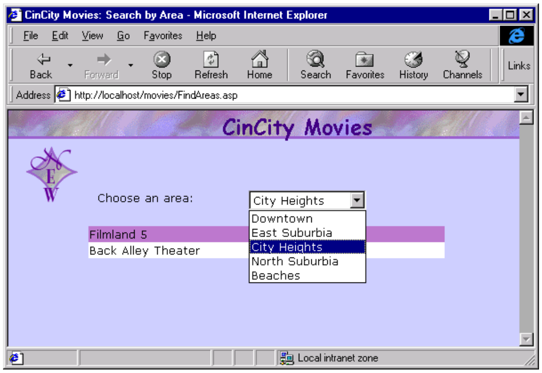
2 In the Script Center, click the plus sign next to the ASP node to open it. Then open the FindCinsRS node.

3 Scroll to the bottom of the list of methods and select MoveFirst(). Drag it to the scripting pane on the left. Close the Script Center.

4 To retain the list selection after it's selected, you need to add a server expression. In the Attributes tab for the dropdown list, change the Selection option to Server Expression. In the Content dialog box, for the server expression enter: Request("List1")

5 Select the dropdown list box and the form element in the Basement, right-click and choose Possible Interactions. Apply the interaction Submit Form1 When List1 Changes. Click OK.

The results are shown in the following figure. Users can choose a new area from the dropdown list and the database is automatically filtered to produce new results in the AutoTable.



CinCity Movies: Search by Area - Microsoft Internet Explorer

Address http://localhost/movies/FindAreas.asp

CinCity Movies

Choose an area: City Heights

Downtown
East Suburbia
City Heights
North Suburbia
Beaches

Filmland 5
Back Alley Theater

Other contracts are available for using with SQL parameters which can be used with specific form elements (checkboxes, radio buttons, edit boxes). You can also create your own contracts for these functions (see “Contracts” on page 417.) Full documentation for these contracts can be found in the Contracts folder of the Drumbeat application folder.

For two examples of working with SQL parameters, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

Maintaining state of recordsets

Maintaining state between pages when searching a database is a very common requirement, particularly when a filter has been applied to a recordset that you want to carry through to subsequent pages. To pass the value from one page to the next, both the original and the target page must contain identical recordsets (recordsets bound to the same content table). Once a recordset has been created it becomes available as a site element that can be reused on subsequent pages.

The value to be passed (known as a hidden form field in HTML), can be passed through a form submit action. When you use a form submit button, and set up the interaction to target a page containing the same recordset, the value will automatically be passed. If you want to use a text or image link, you must set up the submit action to maintain state of the recordset yourself.

To pass a filter value from one page to the next via a linked element:

1 On the first page, place a form element in the Basement. Assign a default action to the form element by clicking the Link Assign button in the Attributes tab and choosing a default page to link to.

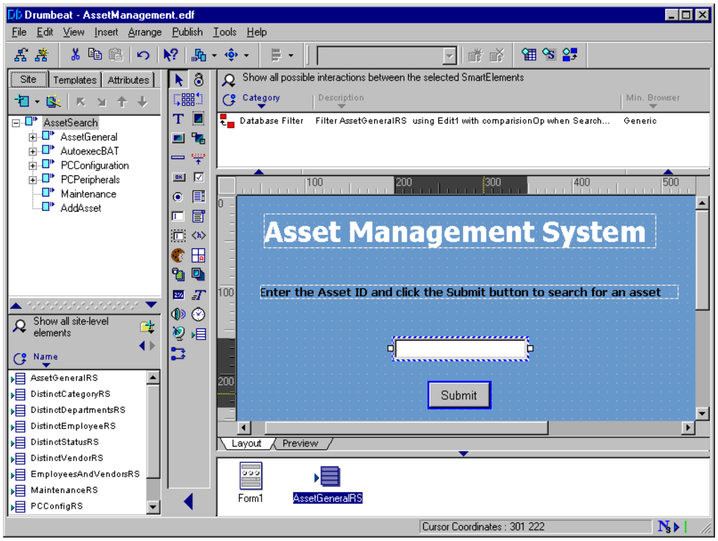
2 Select the form element and the anchor element to be hyperlinked. Right-click and choose Possible Interactions.

3 In the Interactions Center, locate and apply the following interaction: Submit Form When Clicked

4 Switch to the page you are linking to in the Site Manager.

5 In the Asset Center, query for site elements. From the Asset Center list, select the same recordset that was on the referring page and place it in the Basement of the receiving page (if the page does not already contain this recordsets).

The example in the following figure shows an asset management system that keeps track of the computer resources in a company. The initial screen allows searching for information about a specific asset by ID number. In this case, the search filter is applied as an interaction between the edit box, the submit button and the recordset.



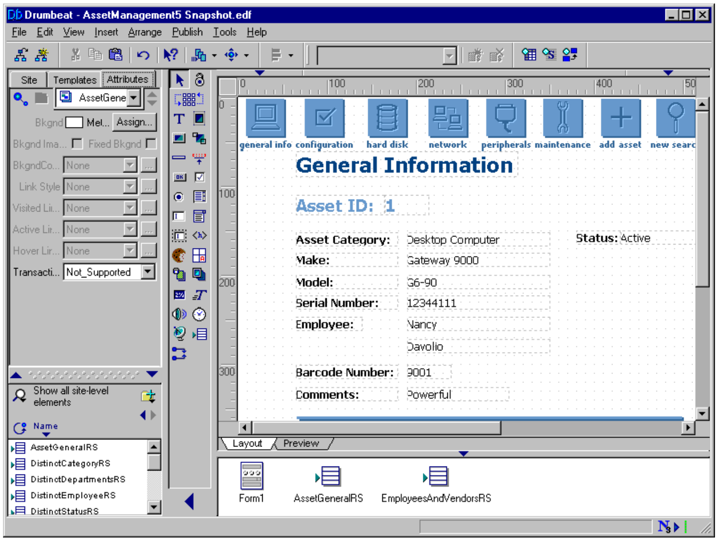
Category	Description	Min. Browser
Database Filter	Filter AssetGeneralRS using Edit1 with comparisionOp when Search...	Generic

Asset Management System

Enter the Asset ID and click the Submit button to search for an asset

Submit

The search result page displays the general information about the selected asset. The links on the top of the page allow you to display different types of information about that asset. Once the filter is applied, it is necessary to maintain the filter through the subsequent links, displayed across the top of the page.



BkgndCo... None
Link Style None
Visited Li... None
Active Li... None
Hover Li... None
Transacti... Not_Supported

General Information

Asset ID: 1

Asset Category: Desktop Computer
Status: Active
Make: Gateway 9000
Model: G6-90
Serial Number: 12344111
Employee: Nancy
Davolio
Barcode Number: 9001
Comments: Powerful

If the links on the image buttons across the top of the page were made as straight links, all of the state information (which asset you wanted to track) would not be passed on. To pass the value (the number of the asset) on to the next page, you must set up a form submit action between the image button and the form element in the Basement. When the interaction is applied, you can choose the page to browse as the action of the form in the Parameters dialog box.

CHAPTER 19
Interactions Center

It is worth remembering that HTML, the language of the Web, is inherently static. HTML is merely a way of marking up a document to express the formatting intentions of its author and has none of the features of a real computer language. It has no inherent logic, no in-built hierarchy of options, and—most important—almost no ability to interact with its readers.

In response to demand from advanced web authors, logic was tacked on to HTML in two ways—first, a different language called JavaScript was invented. JavaScript does have the logical features of a real computer language, and it can be incorporated in the exact same document as the basic HTML that describes a page (JavaScript may also be in a separate file, and in Drumbeat usage, it usually is). Second, JavaScript was extended into the fuller features of Dynamic HTML, or DHTML.

Drumbeat is quite capable of bringing the resources of both JavaScript and DHTML to your web site—and the really good news is that you can call upon these powerful tools without having to learn JavaScript or even having any but the most general idea about how it works.

Activations and interactions

JavaScript makes available extensive possibilities for reader interaction with a web page. It would be impossible for any set of pre-written JavaScript functions to encompass everything possible—but Drumbeat does offer a very wide range of functions, helpfully filtered according to which SmartElements are selected in the layout. These prewritten functions are known as contracts, and the effects invoked by these contracts are activations (when applied to a single SmartElement), or interactions (when applied to a set of SmartElements). From the user's point of view, these appear as choices in the Interactions Center when objects are selected in the layout.

An example of an activation might be an icon that opens a custom window when it is clicked. An example of an interaction might be that when the user moves the mouse over one image, another image or a text box pops up (or becomes visible). Interactions can also be used to validate form data, initiate media playback, and control DHTML elements. Each SmartElement can have one or more contracts attached to it, that control its activations and interactions.

An interaction invariably involves at least one SmartElement, and at least one—usually only one—event. An event is usually some action taken by the user of the page, such as a mouse movement, a mouse click, or a keyboard action (although an event can also occur without user intervention, such as a timer expiring). The event is captured by the contract and some desired logical process then happens. It is important to understand that events do not have to be associated with a specific object in the layout—they may relate to the document itself (document onLoad and onUnload being the prime examples).

Drumbeat automatically keeps track of which interactions you have applied on a page, and it embeds the appropriate JavaScript code and event handlers in the page. In fact, Drumbeat even understands which scripting code works in each browser and automatically makes appropriate scripts available when your page is viewed in Netscape Navigator or Microsoft Internet Explorer.

When a SmartElement participates in several interactions, the order in which they are invoked can be very important. This is of crucial importance when one of the interactions is a form submit event, which must be last in order if all interactions are to take proper effect. Drumbeat therefore offers a special display mode for ordered interactions, in which their order may be inspected and changed by dragging.

To impose an activation or interaction is:

1. Select the object or set of objects (SmartElements) you wish to activate.
2. Expose the Interactions Center. It will contain a list of only those interactions relevant to your selected object(s) and achievable with your target browser(s). The minimum browser level required for each interaction is listed.
3. Choose from the list by double-clicking or right-click and click Apply.
4. If appropriate, add parameters in a Parameters dialog box (for example, the text of an error message, a column to be searched in a database, or the speed and direction of a movement).

Using the Interactions Center

When open, the Interactions Center occupies the space directly above the layout, known as the Attic, which is shared with the Content Center and the Script Center. There are several ways of exposing the Interactions Center, depending on how you are working in Drumbeat at the moment.

To open the Interactions Center:

Do one of the following:

- Click the Interactions Center button on the toolbar.
- Select an element or elements and right-click and choose Possible Activations or Possible Interactions.
- Select View > Interactions Center from the main menu.

If you have dragged the split bar to open or close the Attic while using the Content Center or Script Center, it may be necessary to reopen the Attic by dragging the split bar down. You can open the Attic automatically by clicking one of the arrow keys on the split bar that divides the layout and the Attic.

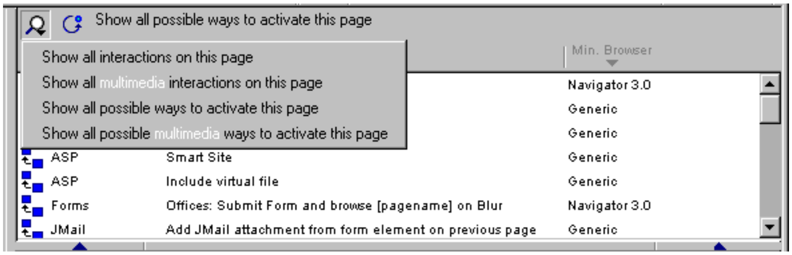
Querying for interactions

You can query for interactions and activations of various types within the Interactions Center.

To query for interactions:

Click the Locate Interactions button and choose an option from the pop-up menu.

The options are context-sensitive. This means that when you select a SmartElement, or select several elements or even the page itself, the list will automatically be filtered so as to reflect the possible activations or interactions that apply to the selected element(s).



Show all possible ways to activate this page

Show all interactions on this page
Show all multimedia interactions on this page
Show all possible ways to activate this page
Show all possible multimedia ways to activate this page

		Min. Browser
		Navigator 3.0
		Generic
		Generic
ASP	Smart Site	Generic
ASP	Include virtual file	Generic
Forms	Offices: Submit Form and browse [pagename] on Blur	Navigator 3.0
JMail	Add JMail attachment from form element on previous page	Generic

Query for different types of activations and interactions through the Locate Interactions button in the Interactions Center.

Sorting activations and interactions by category

All interactions belong to one of a set of categories. The category each interaction belongs to is listed in the far left column in the Interactions Center. The categories are such functional groupings as multimedia, edge transitions, database edit, and so forth.

There are two distinctly different ways to sort interactions by category.

To sort the entire list by category:

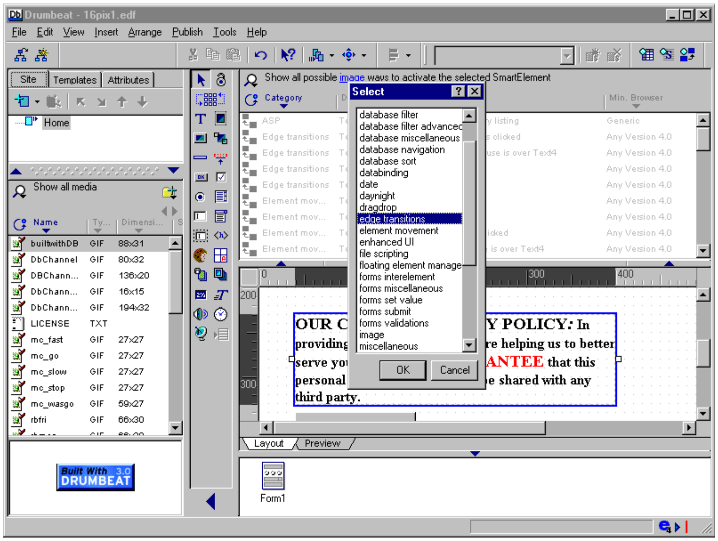
Click on the word Category in the list header. This sort toggles between alpha ascending and alpha descending order.

To sort by an interaction category:

1 Click the Locate Interactions button.

2 Instead of selecting All Interactions, select All [category] Interactions. Interactions in that category are promoted to the head of the list and all others are grayed out (although they are in fact still available).

To change the category focus, click on the blue underlined query phrase in the header bar. Select the category from the pop-up menu.



Changing the default category of activations.

Many times you will want to browse the entire list of possibilities. Presorting possible interactions focused on a category is most useful when the number of interactions is quite large and you expect to be working in that category for a while. You can then alpha-sort the list on the Description field, and no other categories will be included in the sort.

Single-element activations

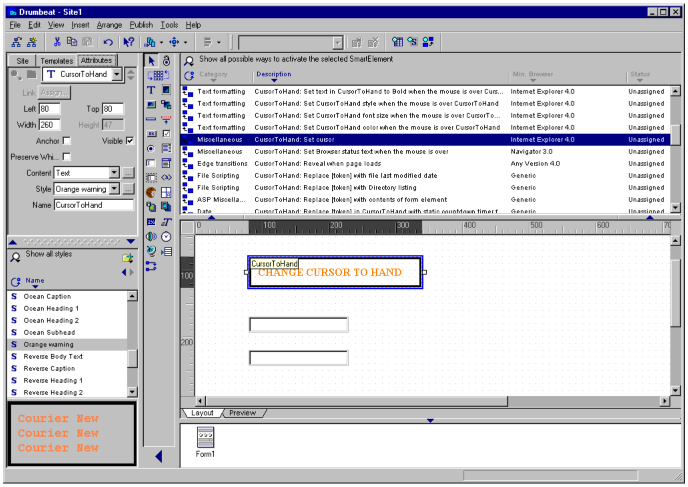
Activations are scripted events that can be applied to a single element, rather than to multiple elements on a page. These include such things as element movement, reveal/hide actions, edge transitions, and simple form validations (verify that an entry in an edit box meets the requirements). You can easily view all the possible activations that can be applied to any element.

To view the possible activations for an element:

Select the element, right-click, and choose Possible Activations.

Activating a single SmartElement

An example of a very simple interaction is sensitization of a line of text, shown in the following figure. Here the SmartElement is the text box named CursorToHand, and the required activation is a change of cursor shape. The event to be captured is the onClick event of the CursorToHand element.



CursorToHand

Left 80 Top 80

Width 260 Height 47

Anchor □ Visible ☑

Preserve Whi... □

Content Text

Style Orange warning

Name CursorToHand

Ocean Caption
Ocean Heading 1
Ocean Heading 2
Ocean Subhead
Orange warning
Reverse Body Text
Reverse Caption
Reverse Heading 1
Reverse Heading 2

Show all possible ways to activate the selected SmartElement

Category	Description	Min. Browser	Status
Text formatting	CursorToHand: Set text in CursorToHand to Bold when the mouse is over Curs...	Internet Explorer 4.0	Unassigned
Text formatting	CursorToHand: Set CursorToHand style when the mouse is over CursorToHand	Internet Explorer 4.0	Unassigned
Text formatting	CursorToHand: Set CursorToHand font size when the mouse is over CursorTo...	Internet Explorer 4.0	Unassigned
Text formatting	CursorToHand: Set CursorToHand color when the mouse is over CursorToHand	Internet Explorer 4.0	Unassigned
Miscellaneous	CursorToHand: Set cursor	Internet Explorer 4.0	Unassigned
Miscellaneous	CursorToHand: Set Browser status text when the mouse is over	Navigator 3.0	Unassigned
Edge transitions	CursorToHand: Reveal when page loads	Any Version 4.0	Unassigned
File Scripting	CursorToHand: Replace [token] with file last modified date	Generic	Unassigned
File Scripting	CursorToHand: Replace [token] with Directory listing	Generic	Unassigned
ASP Miscella...	CursorToHand: Replace [token] with contents of form element	Generic	Unassigned
Date	CursorToHand: Replace [token] in CursorToHand with static countdown timer f...	Generic	Unassigned

Assigning a simple activation to one SmartElement, in this case an edit box.

To activate a text box:

1. Select the text box in the layout, right-click, and select Possible Activations.
2. Scroll the list of possible activations until you come to the miscellaneous item CursorToHand: Set cursor. Double-click the activation or right-click and select Apply.
3. When the Parameters dialog box appears, use the handy pop-up menu to select the cursor, type Hand.

When you click OK in the dialog box, the icon to the left of the activation text turns red, indicating that this activation is in use on the page, and its status is listed as Assigned.

Once you have assigned an activation and or interaction(s) to an element, you can view the assigned activations and interactions.

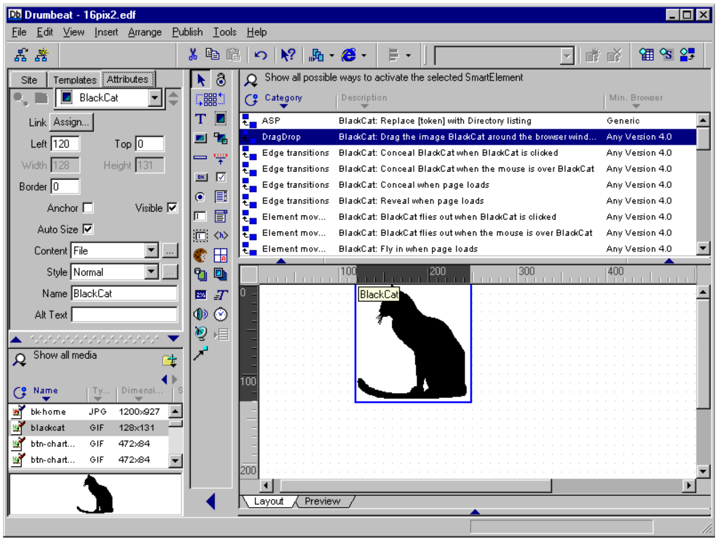
To view assigned interactions on an element:

Select the element, right-click, and select Assigned Interactions (ordered).

Note: Another common activation that is available for almost all SmartElements is `Set browser status text when mouse is over[shows]`The browser status text is, of course, the message that appears temporarily in the status bar at the bottom of the browser screen. Apply this activation to an image or even a text block, and the Parameters dialog box will let you enter the text you'd like to see down in the browser status bar.

Making page objects drag-and-drop

One of the most interesting Dynamic HTML special effects is one that allows users to drag and drop objects around on the page. Even though Microsoft and Netscape have completely different ways of achieving the same DHTML effect, Drumbeat has a single implementation. The following figure shows an image being activated so that users of the page can pick it up, drag it around, and drop it anywhere in the browser window.



BlackCat

Link Assign...

Left 120 Top 0

Width 128 Height 131

Border 0

Anchor □ Visible ☑

Auto Size ☑

Content File

Style Normal

Name BlackCat

Alt Text

Show all possible ways to activate the selected SmartElement

Category	Description	Min. Browser
ASP	BlackCat: Replace [token] with Directory listing	Generic
DragDrop	BlackCat: Drag the image BlackCat around the browser wind...	Any Version 4.0
Edge transitions	BlackCat: Conceal BlackCat when BlackCat is clicked	Any Version 4.0
Edge transitions	BlackCat: Conceal BlackCat when the mouse is over BlackCat	Any Version 4.0
Edge transitions	BlackCat: Conceal when page loads	Any Version 4.0
Edge transitions	BlackCat: Reveal when page loads	Any Version 4.0
Element mov...	BlackCat: BlackCat flies out when BlackCat is clicked	Any Version 4.0
Element mov...	BlackCat: BlackCat flies out when the mouse is over BlackCat	Any Version 4.0
Element mov...	BlackCat: Fly in when page loads	Any Version 4.0

Name	Ty...	Dimensi...
bk-home	JPG	1200x927
blackcat	GIF	128x131
btn-chart...	GIF	472x84
btn-chart...	GIF	472x84

Making an image draggable in Drumbeat.

A refinement on the simple drag activation is to add a Drop Zone. This SmartElement, with the appropriate interaction applied, forces the dragged object it interacts with to drop only within range of the Drop Zone. The Drop Zone SmartElement, valid for Internet Explorer 4.0 only, can be found in the Custom tab of the SmartElement library. You should not apply both the Drag activation and the Drop Zone interaction to the same object, or the Drop Zone will be ineffective.

Multiple complex activations

A single SmartElement can have several activations. For example, you can make a block of text change in two ways when moused over. This works with Internet Explorer 4.0 only, since Netscape's current document object model does not allow for changes to text once the page has been laid out.

To set two activations, italics and border, for the same object:

1 Right-click on the text and select Possible Activations from the pop-up menu.

2 Double-click on Text1: Set text in Text1 to Italic when the mouse is over Text1.

3 Double-click on Text1: Add border to Text1 when the mouse is over Text1.

4 Since there are many possible styles for a text border, you may want to specify your border in terms of its style, width, and color. A Parameters dialog box pops up to enable you to make these choices.

You can now verify that both activations are attached to your text by right-clicking on the text object and selecting Assigned Interactions from the pop-up menu. The two activations, italic and border, will appear in the Interactions Center.

Parameters of an activation or interaction may be edited. If, for example, you do not like the border you just specified, you can edit the parameter.

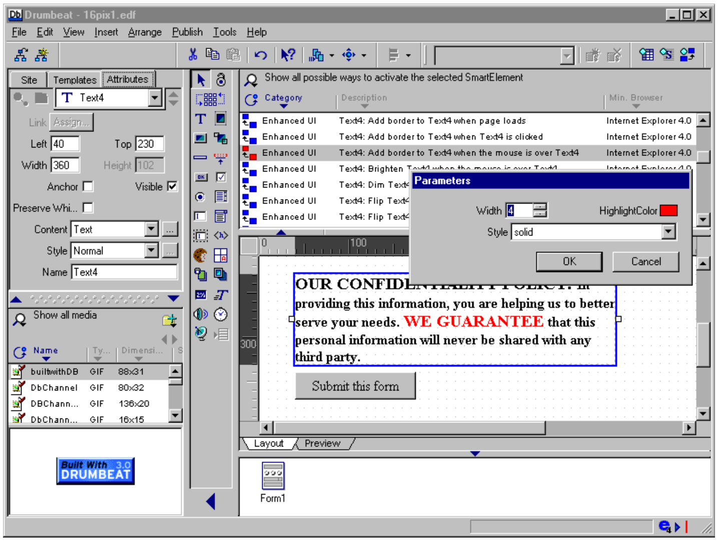
To edit an interaction and change parameters:

1 Right-click on the activation in the Interactions Center and select Edit.

2 In the Parameters dialog box, make whatever changes you want. Click OK to close the dialog box and save your changes.

To remove an activation or interaction:

Right-click on the interaction in the Interactions Center and select Remove.



Text4

Left 40 Top 230

Width 360 Height 102

Anchor □ Visible ☑

Preserve Whi... □

Content Text

Style Normal

Name Text4

Category	Description	Min. Browser
Enhanced UI	Text4: Add border to Text4 when page loads	Internet Explorer 4.0
Enhanced UI	Text4: Add border to Text4 when Text4 is clicked	Internet Explorer 4.0
Enhanced UI	Text4: Add border to Text4 when the mouse is over Text4	Internet Explorer 4.0
Enhanced UI	Text4: Brighten Text4 when the mouse is over Text4	Internet Explorer 4.0
Enhanced UI	Text4: Dim Text4	
Enhanced UI	Text4: Flip Text4	
Enhanced UI	Text4: Flip Text4	

Width 4 HighlightColor 🟥

Style solid

OUR CONFIDENTIALITY POLICY: In providing this information, you are helping us to better serve your needs. WE GUARANTEE that this personal information will never be shared with any third party.

Name	Ty...	Dimensi...
builtwithDB	GIF	88x31
DbChannel	GIF	80x32
DBChann...	GIF	136x20
DbChann...	GIF	16x15

Example of a parameters dialog box applying detail to an interaction.

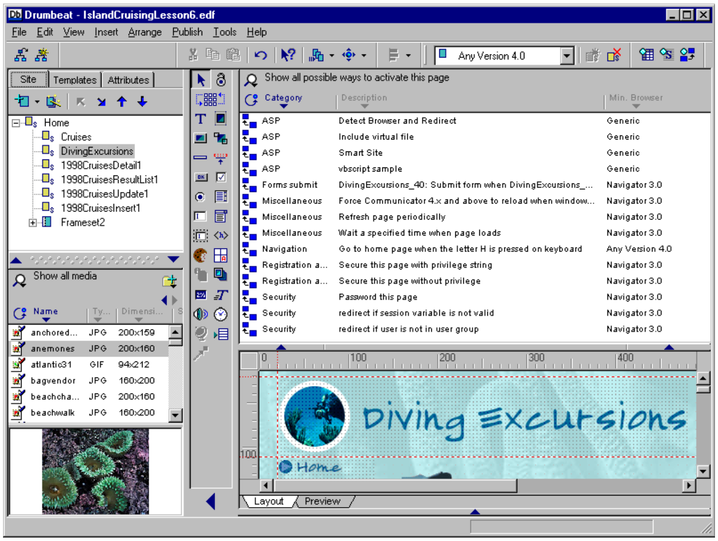
Page activations

Certain activations apply at the document level to the page itself, using the page as the object. You can view the list of possible page activations in the Interactions Center.

To view possible page activations:

Do one of the following:

- Right-click on any vacant space in the layout and select Possible Activations from the pop-up menu.
- In the Interactions Center, click the Locate Activations button and select Show All Possible Ways to Activate This Page.



Name	Ty...	Dimensi...
anchored...	JPG	200x159
anemones	JPG	200x160
atlantic31	GIF	94x212
bagvendor	JPG	160x200
beachcha...	JPG	200x160
beachwalk	JPG	160x200

Category	Description	Min. Browser
ASP	Detect Browser and Redirect	Generic
ASP	Include virtual file	Generic
ASP	Smart Site	Generic
ASP	vbscript sample	Generic
Forms submit	DivingExcursions_40: Submit form when DivingExcursions_...	Navigator 3.0
Miscellaneous	Force Communicator 4.x and above to reload when window...	Navigator 3.0
Miscellaneous	Refresh page periodically	Navigator 3.0
Miscellaneous	Wait a specified time when page loads	Navigator 3.0
Navigation	Go to home page when the letter H is pressed on keyboard	Any Version 4.0
Registration a...	Secure this page with privilege string	Navigator 3.0
Registration a...	Secure this page without privilege	Navigator 3.0
Security	Password this page	Navigator 3.0
Security	redirect if session variable is not valid	Navigator 3.0
Security	redirect if user is not in user group	Navigator 3.0

Possible activations of the page itself.

To add a password to a page:

1. Query the Interactions Center for Show All Possible Activations on This Page.
2. Double-click the interaction Password This Page.
3. In the Parameters dialog box, provide the password and the page to redirect to on password failure.

To place a browser redirect activation on a page:

1 Query the Interactions Center for Show All Possible Activations on This Page.

2 Double-click the interaction Detect Browser and Redirect.

3 In the Parameters dialog box, click the Assign button for the browser you want to redirect and select the page from the tree in the next dialog box.

For browser redirection, the Parameters dialog box allows for four different pages to redirect to.

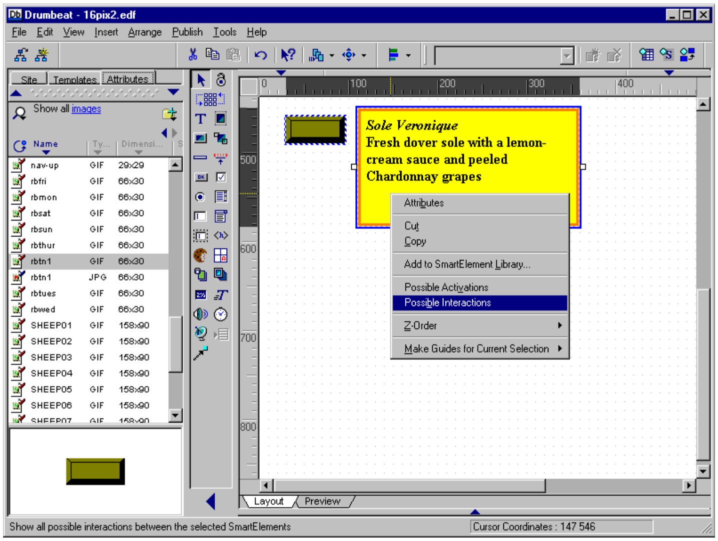
Note: Redirection to several page versions is handled differently when you convert a page to a SmartPage. This procedure has some advantages and is covered in `Chapter 12, SmartPages`

Interactions with multiple elements

The difference between activations and interactions is simply that interactions are scripted actions that are applied to more than one element. Interactions range from the very simple (click a button to change the visibility of another element) to complex interactions involving form elements and record sets, and even interactions between two record sets. Many database interactions are covered in Chapters 17 and 18.

Specifying simple interactions

For an example of a simple interaction, consider an interaction between an icon and a block of text. The following figure shows a single icon and a prewritten block of text representing a restaurant menu item. Both of these elements have been selected and the pop-up menu has been brought up by right-clicking on one of them.



Name	Ty...	Dimensi...
nav-up	GIF	29x29
rbfri	GIF	66x30
rbmon	GIF	66x30
rbsat	GIF	66x30
rbsun	GIF	66x30
rbthur	GIF	66x30
rbtn1	GIF	66x30
rbtn1	JPG	66x30
rbtues	GIF	66x30
rbwed	GIF	66x30
SHEEP01	GIF	158x90
SHEEP02	GIF	158x90
SHEEP03	GIF	158x90
SHEEP04	GIF	158x90
SHEEP05	GIF	158x90
SHEEP06	GIF	158x90
SHEEP07	GIF	158x90

Sole Veronique
Fresh dover sole with a lemon-cream sauce and peeled Chardonnay grapes

Show all possible interactions between the selected SmartElements

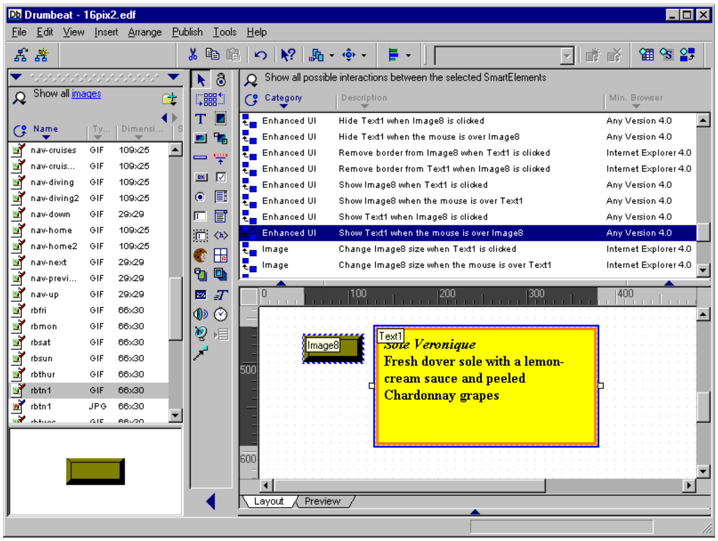
Cursor Coordinates : 147 546

Selecting Possible Interactions from the pop-up menu

Note that the menu offers both Possible Activations and Possible Interactions. You might want to use the Possible Activations option if you wanted both objects to be activated similarly—both of them to hide when clicked, for example. In this example, however, Possible Interactions is chosen and the Interactions Center is in the attic, and a list of possible interactions between the selected objects is displayed. Among the interactions is one that will make the text appear when the user mouses over the icon. The whole operation is done in three very easy steps.

To set up an interaction between an icon and a text block:

1. Select both elements on the layout.
2. Right-click either element to bring up the pop-up menu, and select Possible Interactions.
3. From the list in the Interactions Center, double-click Show [Text] When the Mouse is Over [Image].



Show all possible interactions between the selected SmartElements

Category	Description	Min. Browser
Enhanced UI	Hide Text1 when Image8 is clicked	Any Version 4.0
Enhanced UI	Hide Text1 when the mouse is over Image8	Any Version 4.0
Enhanced UI	Remove border from Image8 when Text1 is clicked	Internet Explorer 4.0
Enhanced UI	Remove border from Text1 when Image8 is clicked	Internet Explorer 4.0
Enhanced UI	Show Image8 when Text1 is clicked	Any Version 4.0
Enhanced UI	Show Image8 when the mouse is over Text1	Any Version 4.0
Enhanced UI	Show Text1 when Image8 is clicked	Any Version 4.0
Enhanced UI	Show Text1 when the mouse is over Image8	Any Version 4.0
Image	Change Image8 size when Text1 is clicked	Internet Explorer 4.0
Image	Change Image8 size when the mouse is over Text1	Internet Explorer 4.0

Name	Ty...	Dimensi...
nav-cruises	GIF	109x25
nav-cruis...	GIF	109x25
nav-diving	GIF	109x25
nav-diving2	GIF	109x25
nav-down	GIF	29x29
nav-home	GIF	109x25
nav-home2	GIF	109x25
nav-next	GIF	29x29
nav-previ...	GIF	29x29
nav-up	GIF	29x29
rbfri	GIF	66x30
rbmon	GIF	66x30
rbsat	GIF	66x30
rbsun	GIF	66x30
rbthur	GIF	66x30
rbtn1	GIF	66x30
rbtn1	JPG	66x30

Sole Veronique
Fresh dover sole with a lemon-cream sauce and peeled Chardonnay grapes

The Interactions Center exposed and the correct interaction selected

This exercise is, in fact, the basis of a very common practice in modern web design—revealing alternative blocks of text by mousing over a set of icons. It is easy to imagine that the scrumptious-sounding menu item might be the special of the day at a restaurant. Furthermore, icons with the names of the days of the week might easily be arranged in a column, and the mouse-over event captured by each of them might reveal the special of the day in the exact same screen space.

To set up interactions between a set of icons and a matching set of text objects:

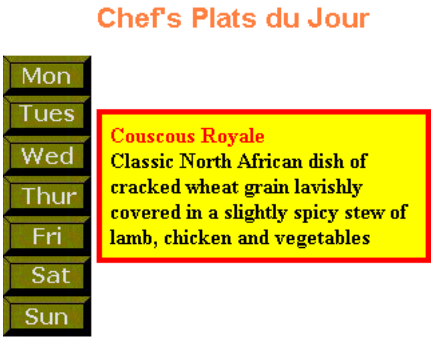
1 Arrange the set of icons in the layout.

2 Write the first text block and create the interaction between it and the first icon exactly as in the previous example.

3 Write the second text block in a vacant space on the layout, create its interaction with icon2, then drag it on top of the first text block.

4 In the Attributes tab, deselect the Visible option.

5 Continue until all text blocks have been stacked.



Couscous Royale
Classic North African dish of cracked wheat grain lavishly covered in a slightly spicy stew of lamb, chicken and vegetables

Preview of the completed set of interactions

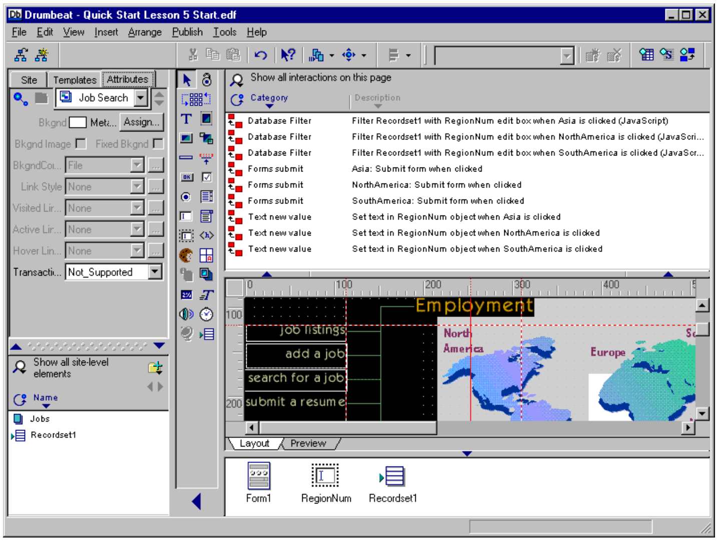
Note: If subsequent editing is needed on the overlayed blocks of text, it is possible to juggle the Z-Order until the needed block is on top. A better technique is to use the pop-up menu of page elements at the top of the Attributes tab. Select the text element you need to edit (this is much easier if you assign meaningful names to your text objects) and click the Content button (the ellipsis button to the right of the Content window). The text editor opens with the text available for editing.

Displaying all interactions on a page

The total activations and interactions in force on a page can become quite complex, particularly on ASP pages that interact with databases. However, it is easy to see at any one time the whole set of interactions that have been applied to elements on any page.

To view all the interactions on a page:

In the Interactions Center, click on the Locate Interactions button and select Show All Interactions on This Page.



Category	Description
Database Filter	Filter Recordset1 with RegionNum edit box when Asia is clicked (JavaScript)
Database Filter	Filter Recordset1 with RegionNum edit box when NorthAmerica is clicked (JavaScri...
Database Filter	Filter Recordset1 with RegionNum edit box when SouthAmerica is clicked (JavaScr...
Forms submit	Asia: Submit form when clicked
Forms submit	NorthAmerica: Submit form when clicked
Forms submit	SouthAmerica: Submit form when clicked
Text new value	Set text in RegionNum object when Asia is clicked
Text new value	Set text in RegionNum object when NorthAmerica is clicked
Text new value	Set text in RegionNum object when SouthAmerica is clicked

All interactions on the page displayed in the Interactions Center.

Note: The same queries for interactions can be run from the Asset Center, if you want a quick overview with the layout fully in view. Short descriptions of each interaction are provided as mouse-over tool tips in the Asset Center.

Sometimes you may see an interaction in this list deleted with a horizontal strike-through line. There can be several reasons for this—it might indicate that an interaction was applied to the page but is no longer possible due to a change in the target browser. For example, you may have started the page targeted to Internet Explorer 4.0 and set up some text activations that worked fine, then later changed the browser target to Any 4.0. Those text activations would then be invalid and struck out in the list. For more about this, see “Disabled interactions” on page 413.

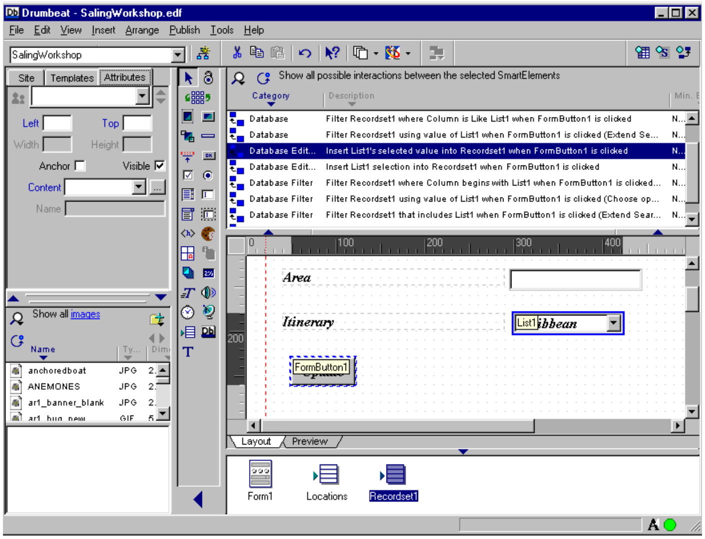
Another reason for an interaction being disabled is that the site has been loaded into a computer that does not have that contract available in its global.dgd file. In such a case, the interaction’s status is also listed as Missing. (For more information about the Global.dgd file and how to export and import contracts, see “The global.dgd and Drumbeat files” on page 418.)

Note: If you are faced with a long list of assigned interactions or possible interactions, bear in mind that the list can be sorted simply by clicking on a column heading. If, for example, you’d like to see the list sorted alphabetically by description, left-click on the Description label. You can click again to reverse the sort order. The sort operation toggles between alpha-ascending and alpha-descending.

Interactions involving recordsets

Many times when you place a recordset on a page, a set of interactions comes along automatically as part of what the DataForm wizard creates. For example, buttons for First/Previous/Next/Last for navigating database records are preset along with all their appropriate interactions. The same applies when you create a static PageSet from a content table.

Interactions between layout objects and record sets, however, are entirely under your control, exactly as are all interactions. You can modify or delete them as you wish, and Drumbeat always tells you, as you select each layout object or set of objects together with the record set, which interactions are theoretically possible.



Show all possible interactions between the selected SmartElements

Category	Description	Min. E
Database	Filter Recordset1 where Column is Like List1 when FormButton1 is clicked	N...
Database	Filter Recordset1 using value of List1 when FormButton1 is clicked (Extend Se...	N...
Database Edit...	Insert List1's selected value into Recordset1 when FormButton1 is clicked	N...
Database Edit...	Insert List1 selection into Recordset1 when FormButton1 is clicked	N...
Database Filter	Filter Recordset1 where Column begins with List1 when FormButton1 is clicked...	N...
Database Filter	Filter Recordset1 using value of List1 when FormButton1 is clicked (Choose op...	N...
Database Filter	Filter Recordset1 that includes List1 when FormButton1 is clicked (Extend Sear...	N...

Name	Ty...	Dim
anchoredboat	JPG	2.
ANEMONES	JPG	2.
ar1_banner_blank	JPG	2.
ar1_bug_new	GIF	5.

Complex interaction involving a record set.

The following figure shows a complex interaction between a list box, an Update button, and a recordset. The requirement is to cause the option selected in the list to be written into a database, updating an existing record. The list box has been created from a separate related recordset, named Locations, to avoid the need for retyping one of several categories in the data. The two recordsets are related through a numerical field, so Drumbeat correctly offers as one option Insert List1's selected value into Recordset1 when FormButton1 is clicked. Selecting this interaction activates the logic to perform this update of the database record. Another interaction one might set up with this set of data would be to filter a display of all records in the database that match a selected category from the Locations list. The interaction in this case would be between the two record sets and the appropriate interaction would be Filter [Target] with Related Field from [Source]. See "Filtering data" on page 385 in for more information on this type of interaction.

Changing the order of interaction events

Many times the order of the interactions and activations on a particular SmartElement does not matter. In a previous example a text block was activated to go to italics and acquire a border when moused over. Whether the italics or the border appears first may be inconsequential, and it will be imperceptible in any case. However, a good page designer will always think of what the effect might be on a slower computer than the one being used for development. Even if you cannot detect a series of events as separate, think about their ideal ordering.

In other cases the order of interactions is, in fact, far more important than a sequence of text style changes. The page simply may not function with an incorrect interaction order.

For example, an image button on a search page may have two interactions on it: one to initiate a database filter, and another to submit the form. If the script that filters the database comes after the form submit interaction, the act of submitting the form will submit the page and send the request to the server, and the database filter will never happen. Inappropriately ordered interactions are a common cause of problems in Drumbeat ASP operations.

Drumbeat makes it easy to rectify such problems. Before you publish a page that contains interactions, you should review the interactions for every object on the page individually.

To view and re-order the interactions on a single element:

1. Right-click an element and choose Order Interactions. Examine the ordering of the list, paying particular attention to any form submit action.
2. If any interaction needs to be moved, select it and drag-and-drop it to its correct position in the list.

For more information about the importance of interaction ordering, see “Ordering elements and interactions” on page 373.

Document-level ordering and object-level ordering

A further refinement on ordering interactions include document-level interactions, which are those applied to the page. You can also display and reorder interactions that have been applied to the page the same as you do for an individual SmartElement.

To view and reorder the interactions on the page:

1. Right-click on an empty spot in layout and selecting Order Interactions.
2. Select an interaction and drag and drop it to the position you want it.

You can manipulate and re-order this list independently of the ordered lists for the individual objects on the page.

It is perfectly possible for an object to capture a document event such as onLoad, and also a local event such as onMouseover. The position of both events in their respective lists may be important and must be assessed independently.

Disabled interactions

There are several circumstances under which an existing interaction becomes disabled. These are conditions that are considered reversible. Therefore, the interaction is not permanently deleted by Drumbeat. Instead, it is ignored when you publish the page. Disabled interactions are listed in the Interactions Center or the Asset Center with a line through them.

Diagnosing disabled interactions

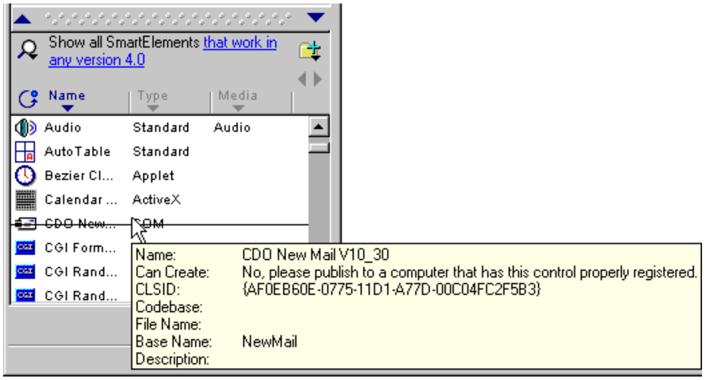
If you see an interaction that has a line though, it you can determine the condition that is preventing that interaction from working. The easiest way is to look at the tooltip for that interaction. Tooltips are not available for the listings in the Interaction Center (because they tend to obscure the other entries in that list), but they are available for interactions listed in the Asset Center.

To view descriptions of interactions in tooltips:

1 Click the Locate Assets button and choose Interactions > Show All Interactions on This Page or Show All Possible Ways to Activate the Selected SmartElement.

2 Place your cursor over an entry in the list.

The Asset Center gives you a tooltip with a short description of the interaction. If the entry is disabled, you will see a description of the problem and some idea of how to solve it.



Show all SmartElements that work in any version 4.0

Name	Type	Media
Audio	Standard	Audio
AutoTable	Standard	
Bezier Cl...	Applet	
Calendar ...	ActiveX	
~~CDO New...~~	~~COM~~	
CGI Form...		
CGI Rand...		
CGI Rand...		

Name: CDO New Mail V10_30
Can Create: No, please publish to a computer that has this control properly registered.
CLSID: {AF0EB60E-0775-11D1-A77D-00C04FC2F5B3}
Codebase:
File Name:
Base Name: NewMail
Description:

Tooltips can be displayed for interactions in the Asset Center to help diagnose disabled interactions.

Following is a brief list of possible problems you might see for a disabled interaction:

This page's target browser cannot use this contract. Try specifying a more capable browser in the File/Preferences/Site command. If this is a SmartPage, use this contract on a page targeting a more capable browser.

Many contracts can be used only on pages targeting highly capable browsers. For example, if you have a contract whose script requires the browser to understand DHTML (rather than standard HTML), then that contract cannot be used on a page targeting a Generic or Nav3 browser.

This situation may occur in several ways:

- You originally created your site for a later browser (Internet Explorer 4 or Any 4.0). You made some pages, added some SmartElements to them, and added some interactions/activations for those elements. Now you change your site preferences for Navigator 3. Any interactions on those pages will be disabled if they require IE4 features.
- You have interactions on a page that you convert to a SmartPage. If you make a browser-specific page for Navigator 3 but the original page had IE4 interactions, those interactions will be disabled in the Navigator 3 version.
- You may have placed interactions on a template—for instance, IE4 interactions on a template that is also used by a Navigator 3 page. In that case, those interactions are disabled.
- One of the SmartElements involved in this interaction/activation isn't valid site preferences. If this is a SmartPage, use this interaction/activation on a page targeting a more capable browser.
- Just as some interactions are browser-specific, some SmartElements can be used only on pages that target very capable browsers. So, if you end up with a page that contains a SmartElement that won't work for the browser that this page targets, the SmartElement itself may be disabled. Disabled SmartElements are drawn with a big red X through them. An interaction that refers to a disabled SmartElement is itself disabled.
- The contract for this interaction/activation is missing. If you deleted this contract, then you can remove this interaction since it is now irrelevant. Otherwise, you may need to recreate the missing contract. It may be possible to obtain the contract from its author. If so, you can then import it into this site.
- You might have created an interaction and then deleted its contract. In this case you need to restore the contract to enable the interaction. This may happen if you give another user your EDF site file but fail to give them the unique contracts that you made for that site. In this case the EDF refers to contracts that are missing. You need to export those unique contracts, which must then be imported by the second user before the interactions are enabled.
- One of the SmartElements used in this interaction/activation fails to meet a necessary condition. For example, that SmartElement might be missing a necessary attribute (property), method, or event.
- If you modify a SmartElement in the Element Library you may render an interaction invalid. For example, if you delete an attribute, method or event that is used in the interaction script, then the contract doesn't make sense any more so the interaction is disabled.
- This interaction/activation requires more advanced or different server support than is presently specified for this site. Try changing the server settings in the File/Preferences/Site command.
- If you change the server preference for the site, some interactions may no longer be relevant. These are disabled.

Extending interactions and contracts

Sooner or later you are going to come across a situation in which you need an interaction or activation that is not part of the original collection of interactions. You open up the Interactions Center and request all possible interactions, and it simply is not in the list. That should not be surprising—no preloaded set of contracts (which is what interactions really are) could deal with every concept of an inventive web developer.

The list of available contracts can be extended in three ways:

- Visit the Drumbeat Element Exchange on the Drumbeat Web site (macromedia.com/software/drumbeat) and download additional contracts. The Element Exchange includes contracts that have been made up since Drumbeat shipped, along with contracts contributed by other developers.
- Use the Drumbeat Contract Builder to customize existing contracts and create new ones for yourself.
- You or your company's JavaScript guru can enter the Drumbeat Script Center and write custom JavaScript routines that can then be easily turned into new contracts using the Contract Builder.

The Contract Builder and the Script Center are described in greater detail in the following two chapters.

CHAPTER 20
Contracts

The underlying driver of all the activations and interactions described in the previous chapter is a series of contracts. Contracts are logical scripts, generally written in client-side JavaScript or JScript—however, they may also be server-side programs. The set of contracts available on your machine is stored in your global.dgd, but interaction descriptions and identifiers (GUIDs) used by a Drumbeat-built site are also stored as part of the project's EDF file.

The global.dgd and Drumbeat files

Exact details of every contract—its description, its category, its browser compatibility, and its parameters, as well as the script itself—are stored in the global.dgd file which normally resides in the Bin folder of your Drumbeat application folder. This enables contracts to be available from site to site. The EDF file describing any particular site built with Drumbeat contains the instances of the contracts used in the site, and their GUIDs, but no other detail. Obviously, then, to publish a site successfully, Drumbeat needs to do a lookup on the local global.dgd file in order to compile the appropriate scripts into the published version of the site.

Several hundred contracts are shipped as standard with Drumbeat. However, you may edit, import, delete and create contracts without restriction. So the global.dgd files on the computers in use at your company may (and usually do) diverge in the course of use. It is not hard to see a possible source of confusion and error here, as EDF files may be exchanged between machines whose global.dgd files differ. All will be well so long as, when one user gives a Drumbeat-built site to another, she also exports any nonstandard contracts as CTR files. The other user can then import the CTR files and they will automatically be incorporated into that computer's global.dgd. There are buttons for import and export on the Contracts tab of the Contract Properties Manager. It is usually not a good idea to export an entire global.dgd file, because this destroys any custom contracts created on the receiving computer.

If an EDF file is on a computer that does not have a compatible global.dgd, missing contracts will be identified, and overlaid with a strike-through style in the Interactions Center. Note that the GUID of a contract is its definitive identifier, NOT its verbal description. Thus if you edit the description text of a contract, the revised text will automatically appear on all sites that reference your global.dgd (and the EDF files will be updated when loaded).

Drumbeat 2.0 & 2000 global.dgd differences

A possibility for error exists in cases where users have upgraded from Drumbeat 2.0 to Drumbeat 2000. In version 2.0, contracts in the global.dgd were indexed somewhat less rigorously, so there's a chance that a global.dgd created under 2.0 may contain an ambiguous reference and invoke the wrong contract when publishing a site in Drumbeat 2000. To avoid this type of contract mismatch it is strongly recommended that when you install Drumbeat 2000 you take the option to migrate your global.dgd, converting all contract indexing to the new standard.

The following components are stored in the global.dgd: contracts, registered SmartElements, global queries, global styles, global JavaScript routines, and global media paths.

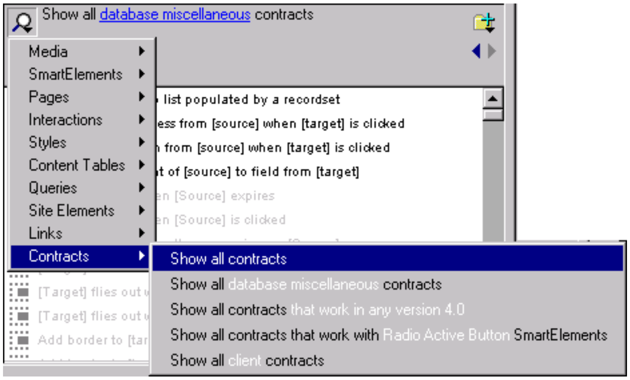
You can import the new Drumbeat 2000 SmartElements and contracts into your new version and recreate global styles and media paths. JavaScript routines can be copied from one version to the other.

Listing contracts in the Asset Center

Although you are used to seeing activations and interactions listed in the Interactions Center (see “Interactions Center” on page 397), the contracts that underlie the interactions are site assets, and therefore may be listed in the Asset Center just like other assets such as media, styles, and so on.

To list contracts available to your site:

1 In the Asset Center, click the Locate Assets button and choose a filter listed on the Contracts submenu.


Show all database miscellaneous contracts
Media
SmartElements
Pages
Interactions
Styles
Content Tables
Queries
Site Elements
Links
Contracts
list populated by a recordset
ess from [source] when [target] is clicked
from [source] when [target] is clicked
t of [source] to field from [target]
en [Source] expires
en [Source] is clicked
[Target] flies out
[Target] flies out
Add border to [tar
Show all contracts
Show all database miscellaneous contracts
Show all contracts that work in any version 4.0
Show all contracts that work with Radio Active Button SmartElements
Show all client contracts

2 Click the underlined word to refine the filter

3 In the Select dialog box, choose an option and click OK.

Using the Contract Properties dialog box

Use the Contract Properties dialog box to manage contracts.

To open the Contract Properties dialog box:

Do one of the following:

- In the Asset Center, double-click any contract in the Contracts list.
- Choose Tools > Contracts.

The Contracts tab

This tab contains the list of contracts, filtered if you so decide for category or browser compatibility. Unlike the contract list in the Asset Center, the list here is only the filtered list, and residue contracts do not appear grayed out. Also, this list is not alpha-sorted. In general, one contract on the list will be selected (highlighted) and information about that specific contract will be found in the other six tabs. Options include the following:

- Add creates a new blank contract to the list. To give it a description and assign it to a category and a browser type, you need to switch to the General tab. Note that the new contract is not compelled to belong to the category and browser type currently selected in the list.
- Copy adds a copy of the highlighted contract to the end of the list. Use this feature to use an existing contract as the basis for a modified version. You can then rename the contract and edit its features. See “Making the Start Action contract” on page 425.

The General tab

This tab is where the description and categorization of a contract are displayed and may be edited. It’s important to notice that the same contract may perform the function of both an activation and an interaction, and usually has different descriptions if it can have both uses.

Bracketed expressions (such as [target] and [source]) in description text are known as tokens. They are used in the actual JavaScript code that will activate this contract as general expressions. When actually applied to objects on a page, they will be replaced by the names of the objects.

If you are creating a new contract, here is where you set whether it is client-side or server-side. If you declare the contract as server-side, you will later be given the option of using VBScript instead of JavaScript.

It is also important to set the minimum browser type for which this contract is valid. For basic JavaScript, Navigator 3.0 will do; Dynamic HTML requires any version 4.0 browser, and for text manipulation you will need Internet Explorer 4.0.

This is also the tab on which you may create new contract categories. Notice that when you have added a new contract, the Category option is empty. You may either select from the existing list of categories, or enter a completely new category. Drumbeat will recognize this as a contract category and treat it just like all the standard categories.

Note: If a custom category later seems redundant, delete all contracts in the category (or edit them to belong to other categories) and you can then delete the entire category in this tab.

The Participants tab

In the Participants tab you state in general terms what SmartElements participate in the contract, and what events, methods and properties are required to complete the scripting. Don't needlessly restrict your terminology. For example, if your intention is to write a contract that makes an image move across the screen when the pointer falls over a button, it is more helpful to the eventual users of this contract to name the tokens *source* and *target* rather than *button* and *image*, because the same contract could be used to move a block of text when another block of text is rolled over. On the other hand, if the contract concerns some operation that could only apply to an image, such as zooming in or out, it would be helpful to write the token as image. It does not matter if you call your participants *rutabagas,* so long as your terminology is internally consistent—and when the time comes to apply the contract to actual page elements, these token names will not even be seen. They will be substituted with the names of the actual page elements involved.

When composing contracts, think about what would be most useful to other users—or to yourself six months down the road when you've forgotten all the details and just want the functionality. When using this tab to examine the details of a standard contract, look at the events, methods and properties of all participants (highlighting them one at a time) to understand what the author of the contract had in mind.

Your definitions of the events, methods and properties of the participants in the contract determines absolutely the circumstances under which the contract (or in this case the interaction) will be offered as a possibility in the Interactions Center when a user queries for interactions. In particular, the event you choose as the basic trigger for the contract determines what event will be included in the Actions tab.

Note: It is rare for a contract to have more than one capturable event, but it can happen. In such a case, deleting one event in this dialog has the effect of deleting any code associated with that event also.

Note: When composing or editing a contract, it is extremely useful to have SmartElements of the type you intend to activate present in the layout. Then to remind yourself of the events, methods and properties these SmartElements can have, open up the Script Center and expand each object in the document.

The examples later in this chapter are designed to help you understand the use of this and other tabs in the Contract Properties Manager more clearly.

The Parameters tab

Many interactions need to have parameters defined when they are applied. The speed of an animation, the final position of a sliding image, the border style of a changing block of text, are examples of parameters used in interactions. In this tab, a contract author can define as many parameters as the new contract needs. Clicking on the Add button, you are presented with a three-part dialog box in which you can define the name, type, and default value of each parameter you need.

When the eventual user of the contract applies it to a page, all parameters are presented in a single dialog box, with their names stated and the default value prewritten to be either accepted or overwritten. The user is not necessarily aware of the data type but it is very necessary for internal data handling. When writing the actual script for this contract, the value of the parameters is obtained by referencing the parameter name as a token. See the following section on "The Actions tab" on page 422 for details.

The Actions tab

This is the tab in which the actual script is written. If you are authoring a new contract, this is where you enter and edit all variable definitions and function code. Participants are entered in square brackets—for example, a line of JavaScript code to fill a variable from the text in an edit box might be

```
var Mnum = parseInt([source].getText());
```

The token [source] will be replaced with the actual name of the edit box when the contract is applied on a page.

The value of parameters is obtained with similarly formatted tokens such as

```
Var p = [Millisec increase];
```

In this case the JavaScript variable `p` now contains the string value of the parameter `Millisec increase` as entered by the author when she applied the contract to create an interaction.

In this tab there are also radio buttons to switch between JavaScript and VBScript (enabled for server-side scripts only), and in cases where several possible events might be captured to trigger the action script, a pop-up menu allows an author to select the appropriate event.

Note: Document level events, such as onLoad, are always available for capture. There is no need to define the document as a participant in the contract since the document is assumed to be a participant in all contracts.

Sometimes, when examining or editing a contract, you may expose the Actions tab and find it blank. Obviously, every contract must have some action script, and if it is not visible the reason most likely is that an incorrect event is selected in this tab.

Note that this tab can be expanded by dragging its edges and corners, and even expanded to a full window, for convenience when working with a complex script.

The Support tab

Many JavaScript functions and definitions are commonly re-used—in fact, many disciplined JavaScript authors keep their own libraries of often-used functions and script fragments. This tab is the equivalent of a personal JavaScript library, and it comes filled with two dozen or so useful routines. Form input validations, browser queries, object movements and edge transitions are examples of the kind of support scripts provided with Drumbeat. To examine a function, select it and click Edit. When creating a new contract, include a support script from the library by selecting it in this tab.

To create a new support script of your own, click New, write a name for the script, and then either enter the script manually in the edit window or copy it from the document where you created and debugged it. From then on, the script may be included in any action script the same way as for the prewritten support scripts. See "Using support scripts" on page 430 for an example of creating a support script.

The Description tab

This tab is used by the author of a contract to describe the details of its intended functionality. There is space for a megabyte of text here—more than enough for examples, references, and so on.

Building a new contract

This section is a step-by-step exercise in contract-building, ending with a collection of SmartElements that will control a DynaImage animation. There will be buttons to start, stop, speed up and slow down the animation—and a window to register the current frame rate. The only standard contract that is of use is a multimedia contract to stop the action. All other contracts will be created in the remaining sections of this chapter.

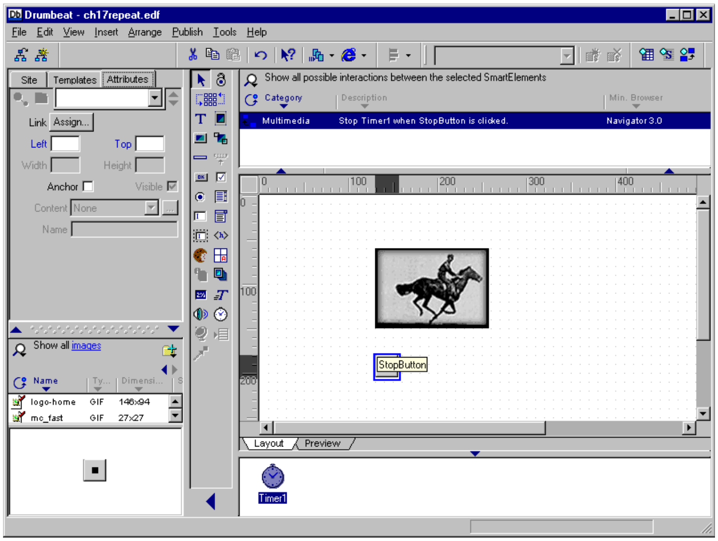
For another working example of building a new contract, see the Drumbeat Support Center (macromedia.com/support/drumbeat).

Using the Stop Action contract

The following illustration shows the beginnings of the layout. Here are the steps taken to arrive at this point:

- A DynaImage SmartElement has been placed on the page and filled with the 20 frames of the animation that is to be controlled.
- A timer has been added, with initial period 50 milliseconds, and with Repeat and Autostart options selected.
- The timer and the DynaImage are activated with the contract Show the Next Image in DynaImage1 when Timer1 Expires. So when the page is published, the animation will immediately start and loop continuously with a 50ms period.
- An image button SmartElement has been added, and a media controller stop button image imposed.

Selecting the button and the timer and querying for interactions, we see the only contract that is relevant in this instance—the multimedia interaction Stop Timer1 when Stop Button is Clicked. After this interaction is applied, publish and browse this page to check your initial work.



Show all possible interactions between the selected SmartElements

Category	Description	Min. Browser
Multimedia	Stop Timer1 when StopButton is clicked.	Navigator 3.0

Name	Ty...	Dimensi...
logo-home	GIF	146x94
mc_fast	GIF	27x27

The timer stop contract applied.

Making the Start Action contract

Since there is no standard contract Start [Timer] When [Source] is Clicked (the contract Play [Target] applies to true multimedia only and is not available for JavaScript image animations), this one has to be created.

To create the start action contract:

1 In the Asset Center, click the Add Assets button and select Contracts.

2 Double-click the existing contract Stop [Target] When [Source] is Clicked.

3 In the Contract Properties dialog box, click the Copy button. A copy of this contract is added to the bottom of the list.

4 Edit the description of this contract in the General tab to read “Start [Target] When [Source] is Clicked.” Check that all other elements in this tab are still correct—the browser compatibility, the category, and so on.

5 Edit the script in the Action tab. The existing script is `[target].stop();` and it needs to be changed to `[target].start();`.

6 Click the Contracts tab, select the new contract, and move it using the Up button until it is immediately above the stop contract. Click OK

Now a second image button SmartElement can be set in layout and a media controller start image imposed. This time when the button and the timer are activated, the new contract to start the animation will be available in the list of Possible Activations. Apply the interaction and then publish and browse to test your results in a browser

Creating a contract to slow the timer

Now a brand new contract has to be created to slow down the timer by defined steps. It has to be described, its participants have to be defined, its parameters set up, and of course the actual script has to be written.

To create a new contract:

1 Choose Tools > Contracts. Click the Add button. A blank contract named New contract is added to the bottom of the list.

2 In the General tab.

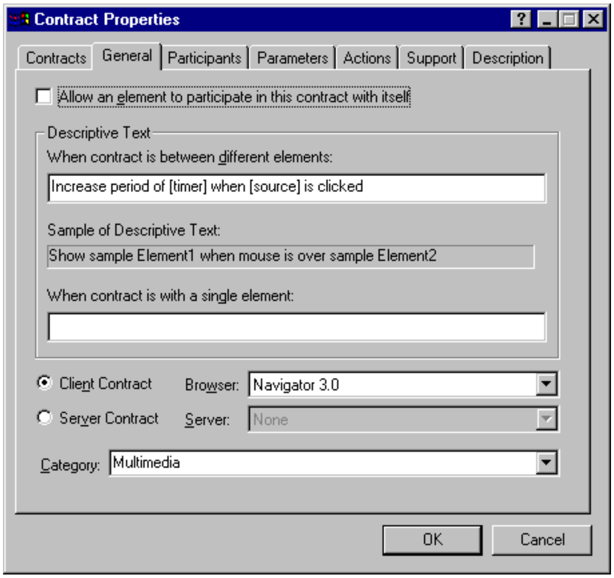
3 For When Contract is Between Different Elements enter the following: Increase Period of [Timer] When [Source] is Clicked. (Obviously, to slow down the timer, its period has to be increased).

4 Deselect Allow an Element to Participate in this Contract with Itself. If the option were selected, it would be possible to set up the page so that the animation period would be changed by clicking on the animation itself, which is definitely not the intention.

5 Select Client Contract.

6 Set the browser compatibility to Navigator 3.0. The minimum browser type that will allow the script to run should always be selected.

7 Set the Category to Multimedia. This tab should now look like the following illustration.



Contract Properties

□ Allow an element to participate in this contract with itself

Descriptive Text

When contract is between different elements: Increase period of [timer] when [source] is clicked

Sample of Descriptive Text: Show sample Element1 when mouse is over sample Element2

When contract is with a single element:

● Client Contract Browser: Navigator 3.0

○ Server Contract Server: None

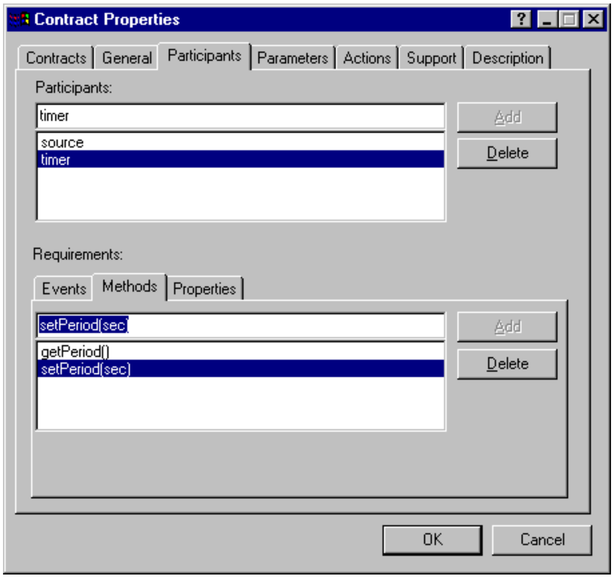
Category: Multimedia

When offering the list of possible interactions between SmartElements, Drumbeat considers the events, methods and properties that are associated with each participant and matches them with the information in the Participants tab. You therefore need to define this contract as valid between something clickable and something retimable. More technically, the contract is for any [source] that has an onClick event, and any [timer] that has getPeriod() and setPeriod(sec) methods.

8 Click the Participants tab. Enter source in the Participants field and click Add. Enter timer and click Add.

9 Select source in the Participants list and, in the Events tab underneath, add onClick (it is case sensitive).

10 Select Timer in the Participants list and, in the Methods tab underneath, add getPeriod() and setPeriod(sec). Both are case sensitive. This tab should now look like the following illustration.



Participants:

timer

source
timer

Requirements:

setPeriod(sec)

getPeriod()
setPeriod(sec)

In the Parameters tab you have to set up a parameter routine that will give those who want to use this contract the option of setting the amount of each timer period increase, defaulting to 10ms.

11 Click the Parameters tab:

12 Click Add. Enter Millisec Increase as the name of the parameter and choose Integer as the type.

13 Enter 10 as the default value.

The final step is to write the actual JavaScript routine that will be invoked by this contract in the Actions tab. This routine needs to read the current timer period with [timer].getPeriod(), then set it using [timer].setPeriod() to whatever it was before plus the parameter [Millisec increase]. Since parameters are always passed as string values, the increase needs to be parseInt([Millisec increase]). Note that the parameter, when selected by the contract user, will be passed into the eventual JavaScript with the proper syntax parseInt(“parameter”).

The correct Action script is:

```
[timer].setPeriod([timer].getPeriod() + parseInt([Millisec increase]))
```

Now when you add another image button representing a media controller slower button, and select it and the timer, the new contract should appear on a list of possible interactions between these elements.

Creating a contract to speed up the timer

Obviously, that contract can now be copied and modified to have the reverse effect—speeding up the animation by reducing the period of the timer.

To modify the slow-down contract to make a new speed-up contract:

1. Open the Contract Properties dialog box and copy the Increase Period contract. Name the copy Decrease Period of [Timer] When [Source] is Clicked. Check that all other elements in the General tab are correct.

   The Participants tab can stay exactly the same, since you want this contract to apply to the same pair of participants.

2. Click the Parameters tab and edit Millisec increase to Millisec decrease. Other details of the parameter remain unchanged.

3. Click the Actions tab and edit the script to:

   ```
   [timer].setPeriod([timer].getPeriod() - parseInt([Millisec decrease]))
   ```

4. Return to the Contracts tab and move the new contract up the list.

Now you can add another image button representing a media controller faster button.

Creating a contract to display the timer period

One refinement that it would be nice to have would be a readout of the current timer period. It should be easy to write a contract that would read this information and display it in an edit box.

To create the timer readout contract:

1. Create a new interaction contract and call it Display [Timer] Period in [Editbox]. Choose the same category (Multimedia) and browser type (Navigator 3) as before.

2. The participants are timer, with event onTimer and method getPeriod(), and editbox, with method setText(text).

   There are no parameters to enter.

3. The code to enter in the Actions tab is:

   ```
   [editbox].setText([timer].getPeriod())
   ```

Note: The complete set of SmartElements—the DynaImage and all five parts of the controller, including all the contracts that interrelate them, could be grouped together and saved as a new combined SmartElement collection (see “Element Library” on page 233).

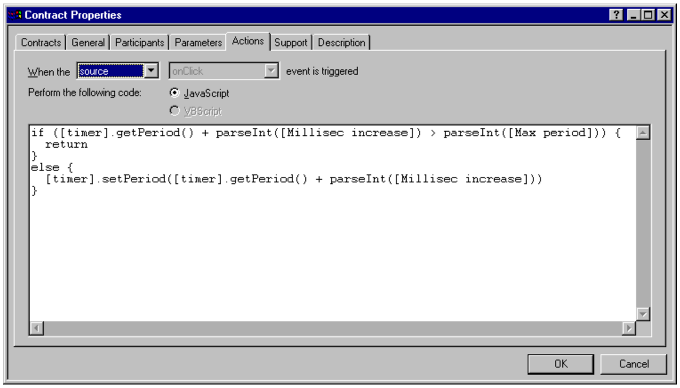
Adding a second parameter to the period increase contract

Programmers—well trained programmers, at least —are constantly thinking about the bounds of routines they write. What are the maximum and minimum values the user should be allowed to enter? This exercise is no exception, and it would be prudent to add some limit to the period of the timer. As a matter of fact, there does not need to be a lower limit. The timer period will not accept a zero or negative number, so it would not be possible to set the period at less than 10ms (the fastest animation speed). However, the Increase period contract would be improved by the addition of a new parameter, Max period, with a default of, say, 300.

Adding a parameter:

1. Open the Increase Period of [Timer] When [Source] is Clicked contract in the Contracts Properties dialog box and click the Parameters tab.
2. Click Add and enter Max period, Integer, and 300 as the data.

Click the Actions tab and amend the JavaScript text to abort the routine if adding Millisec increase to the current timer period would make it greater than Max period. The correct script is shown in the following illustration. In the layout, edit the interaction between the timer and the slow button by right-clicking it in the Interactions Center. The new parameter dialog box will appear.



When the source onClick event is triggered

Perform the following code: ● JavaScript ○ VBScript

```
if ([timer].getPeriod() + parseInt([Millisec increase]) > parseInt([Max period])) {
  return
}
else {
  [timer].setPeriod([timer].getPeriod() + parseInt([Millisec increase]))
}
```

Using support scripts

This technique of script support has significant advantages. It creates more economic code, since a support script is generated once only but may be used by any number of different main routines. It creates more readable code, since it conforms to modern programming practice by keeping the main executive code as uncluttered by sub-routines as possible.

Support Scripts are available in the Support tab of Contract Properties Manager. See "The Support tab" on page 423.

Creating a Sample Support Script

This example will be a simple exercise in using support scripts to format the date and write it to a document. Depending on the JavaScript version in use, the format of a standard date object is something like:

```
Thu Nov 05 16:00:00 GMT-0800 (PST) 1998
```

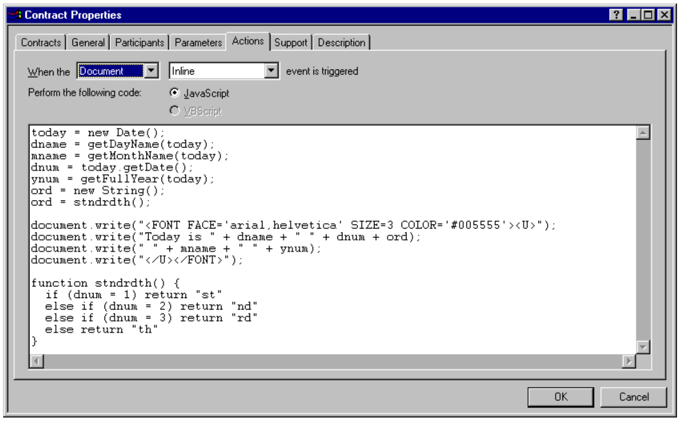
This exercise will format the date more decoratively as:

Today is Thursday, 5th November 1998

Initial steps to create the support script:

1. Open the Contract Properties Manager and click Add.
2. Click the General tab and name the new contract Write Formatted Date to Document Inline. Select the checkbox upper left to enable participation by a single element.
3. Create a new category, Date/time, for this contract.
4. Set browser compatibility to Generic, and contract type to Client.
5. Since this is to be a document-level contract, there are no participants.

6 If you now click the Support tab and examine the support scripts labelled date support functions, you will see that there are two prewritten functions, GetDayName(d) and GetMonthName(d). Assuming d is a date object, these functions return the full name of the day of the week and the full name of the month, respectively. The script shown in the Actions tab below is what's required. Note that the event that triggers this contract is simply document inline—a pseudo-event, perhaps (inline script is script that runs without waiting for any event, not even document load).



When the Document Inline event is triggered

Perform the following code: ● JavaScript ○ VBScript

```
today = new Date();
dname = getDayName(today);
mname = getMonthName(today);
dnum = today.getDate();
ynum = getFullYear(today);
ord = new String();
ord = stndrdth();

document.write("<FONT FACE='arial,helvetica' SIZE=3 COLOR='#005555'><U>");
document.write("Today is " + dname + " " + dnum + ord);
document.write(" " + mname + " " + ynum);
document.write("</U></FONT>");

function stndrdth() {
  if (dnum = 1) return "st"
  else if (dnum = 2) return "nd"
  else if (dnum = 3) return "rd"
  else return "th"
}
```

CHAPTER 21
Script Center

Although Drumbeat offers a very wide and useful selection of interactions (contracts), no such selection could possibly encompass everything that an inventive web site author might dream up. Therefore, Drumbeat features a Script Center that has labor-saving features such as drag-and-drop events, methods, and properties, but whose primary use is as a completely free-form script builder and editor.

There are at least three ways in which you might want to use the Script Center as opposed to the Contract Builder:

- JavaScript coding for functions that are specific to a site. (Such scripts could be exported for later general use, but not very conveniently. Any JavaScript that might have application in other sites would be better converted to a contract.)
- Additions to existing contracts. Edit a script associated with any SmartElement used on the layout by adding to the existing script directly in the Script Center.
- Debugging of JavaScript that you intend to make into a contract.

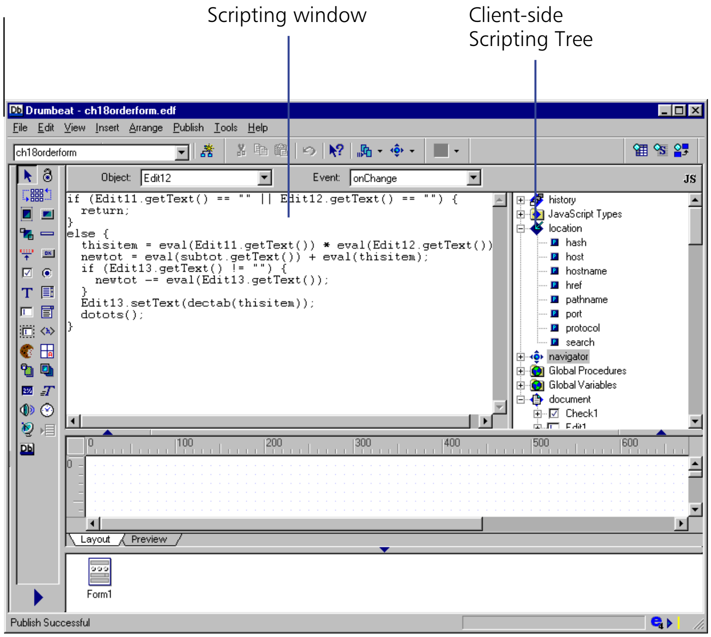
Note: This chapter is not intended to be a JavaScript or VBScript tutorial. Readers are assumed to have some familiarity with the scripting language they want to use.

The Script Center

The Script Center shares the Attic, the workspace above the layout, with the Content Center and the Interactions Center. Each of these three working areas can appear in that same area whenever you need to work with them.

To open the Script Center:

Click the Script Center button on the toolbar, or choose View > Script Center, or right-click any SmartElement and choose Edit Script..



Object: Edit12 Event: onChange

```
if (Edit11.getText() == "" || Edit12.getText() == "") {
  return;
}
else {
  thisitem = eval(Edit11.getText()) * eval(Edit12.getText())
  newtot = eval(subtot.getText()) + eval(thisitem);
  if (Edit13.getText() != "") {
    newtot -= eval(Edit13.getText());
  }
  Edit13.setText(dectab(thisitem));
  dotots();
}
```

Type JavaScript into the Scripting window, or drag objects, methods, and properties from the Scripting Tree into this window to compose script command lines quickly and accurately.

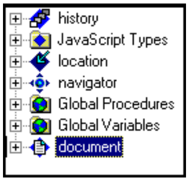
Choosing objects and events

Since every JavaScript function must capture an event in order to execute (including the pseudo-event document inline, and server events), at any time the Script Center is active the event and the object that originates the event must be selected in the two pop-up menus at the top of the Script Center. The Object pop-up menu automatically includes a list of all elements in the layout that are capable of originating events, and the Event pop-up menu automatically includes all possible events that the selected object can originate. JavaScript can not be successfully imposed upon an object that is not present in the layout or triggered by an event that is not in the list.

Note: The pop-up menus don't always refer to an object and an event. They have different functions when dealing with Global Procedures and Global Variables. See "Creating global procedures" on page 441.

The Scripting Tree

The Scripting Tree is an expandable multi-level list that works like a standard Windows directory tree—a plus-box indicates that there is lower-level content to be viewed by clicking on it.



The Scripting Tree fully compressed.

The Scipting Tree contains the following top-level object types:

- History represents a data object that keeps track of recently-visited web pages. This is the same history stack as the one you see using your browser's Go menu option. In JavaScript it is used to force a return to a different page in the stack, either back or forward.
- JavaScript Types is a list of all the data types that JavaScript handles—arrays, strings, Booleans, date objects, and so on. The built-in functions such as eval() and parseInt() are also found here, as is the Math object.
- The location object has a set of properties representing the various sub-components of a URL. It can be used to read the current URL or to re-direct to any other URL.
- The navigator object has a set of read-only properties that can be used to detect the type of browser in use.
- Global Procedures are functions that can be called from any other JavaScript function in the entire site. They are not specifically triggered by capture of an event, but act very much like the support scripts in the Contract Manager, to make the overall scripting task economical and readable. None are shipped as standard with Drumbeat, but instructions for writing them are included in this chapter.
- Global Variables are, like global procedures, variables that are declared in such a way as to be available throughout your site. They may be client or server variables.
- Document is by far the most complex, important, and much-used section of the Scripting Tree. It expands to represent the entire Document Object Model of the current page. The document has events, methods, and properties of its own (for example, the onLoad event, the writeln(text) method, the lastModified property). Every element of the page also has—in general—events, methods and properties of its own.

Scripting in the Script Center

If you are expert with JavaScript, note that Drumbeat sometimes builds statements that are not exactly as you would code them manually. References to form object values are an example. Nevertheless, JavaScript that has been written and debugged external to Drumbeat can be pasted into the Scripting window without needing to be changed.

To create scripts in the Script Center:

1. Right-click the object you want to activate, choose Edit Script and then the event you wish to capture. Alternatively, choose the object and event for which you want to script from the pop-up menus in the Script Center.
2. Drag the methods (functions) from the Scripting Tree to the scripts pane.
3. Rewrite and add to the script as desired.

You can also write your scripts directly into the script pane. However, dragging and dropping from the Scripting Tree ensures that the correct syntax is used.

When an Event is exposed in the Scripting Tree, double-clicking it loads it into the event pull-down in the scripting pane.

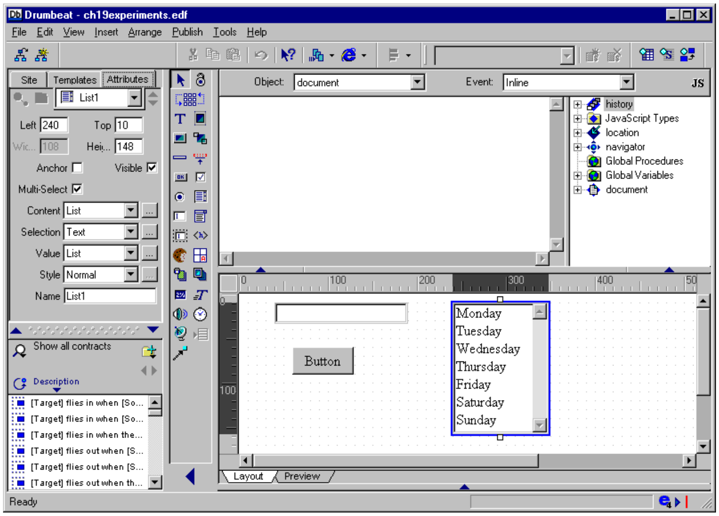
A simple scripting exercise

This exercise will familiarize you with the basic processes of the Script Center. The purpose of the script in this example is to capture a mouse click event on a button and transfer an option selected in a list box to a separate edit box. There is no standard contract for this interaction.

To create the simple script:

1. Insert a new page in a site with Navigator 3 browser compatibility.
2. Place an edit box SmartElement on the page.
3. Place a list box SmartElement on the page. With the list box selected, click the Attributes tab. Click the Content ellipsis button and populate the list box with a set of strings, such as the days of the week. On the layout, drag the box down so that all options are visible.
4. Place a form button SmartElement on the page and, in the Attributes tab, set its type to button rather than submit.

5 Open the Script Center. Your screen should now look essentially like the following illustration.



Drumbeat - ch19experiments.edf
File Edit View Insert Arrange Publish Tools Help
Site Templates Attributes
Show all contracts
Description
[Target] flies in when [So...
[Target] flies in when [So...
[Target] flies in when the...
[Target] flies out when [S...
[Target] flies out when [S...
[Target] flies out when th...
JS
history
JavaScript Types
location
navigator
Global Procedures
Global Variables
document
Button
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Sunday
Layout
Preview
Ready

List1
Left 240 Top 10
Wid... 108 Hei... 148
Anchor ☐ Visible ☑
Multi-Select ☑
Content List
Selection Text
Value List
Style Normal
Name List1

Object: document
Event: Inline

6 Right-click on the button in layout, then selecting Edit Script > onClick.

Now any script that is placed in the Scripting window will execute when the button is clicked.

7 In the Scripting Tree, click the plus next to document, and then click the plus next to Edit1 object so that you can see its method setText(text). Drag this method into the Scripting window

8 In the Scripting window, delete the word *text* in the parentheses and leave the cursor between the (now empty) parentheses.

9 In the Scripting Tree, expand the List1 object so that you can see its method getSelectedText(). Drag this method into the Scripting window in the position between the two parentheses.

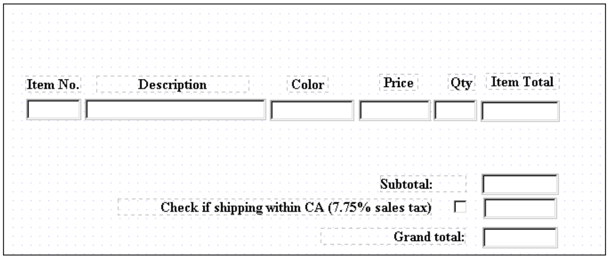
10 The script should now read:

```
Edit1.setText(List1.getSelectedText()).
```

Correct the script if necessary and publish. To test the script, select any item in the list and click the button. The selection should appear in the edit box. (Note that if you checked Multi-Select in the Attributes tab, you can select more than one option and both will appear.)

A complex scripting exercise

This exercise takes a rather conventional online merchandise order form and gives it some useful bells and whistles: automatic addition of each row horizontally and vertically, decimal tabbing (alignment of the decimal points in a column of figures), calculation of sales tax, and truncation of the result total to 2 decimal places.



Item No.	Description	Color	Price	Qty	Item Total

Subtotal:

Check if shipping within CA (7.75% sales tax) □

Grand total:

This illustration shows the header row and the first row of edit boxes into which the user will enter ordering information, and an Item Total edit box. Below is a subtotal box, a sales tax calculation line including a checkbox, and finally a Grand Total box. The “price” edit box has been named price1, the “qty” edit box, qty1, the Item Total itemtot1. Down below, the boxes are subtot, salestax, and grandtot. The checkbox is named califCheck. Since part of the point is decimal tabbing, all the numeric edit boxes are styled to display a monospace font (since a proportionally-spaced font will not display proper decimal tabbing).

Eventually, there will be several lines in this order form—but since they will be operationally identical to line 1, you can take advantage of a wonderful feature of Drumbeat. Once you have the top line working correctly, you can marquee-select the entire row of edit boxes and paste it as many times as you want underneath. The scripting references will all adapt automatically.

Creating an initial calculation script

The basic requirement is to take whatever is in price1, multiply it by whatever the user enters in qty1, and write the result in itemtot1. But what event needs to be captured? You might think the onBlur event of qty1, as the user finishes input on that line. But this will be a more effective page if the totalling function also executes when the user later returns to the line and makes a change. The best way to code it is to choose the onChange event and execute the exact same function for onChange qty1 and also onChange price1.

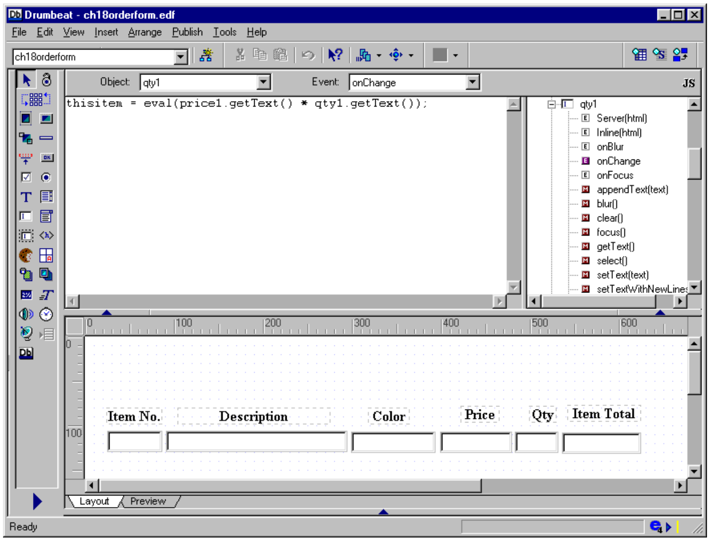
To create the initial calculation script:

1 Open the Script Center and choose qty1 for Object and Event and onChange for Event.

2 In the Scripting Tree, expand the price1 object and drag in its method getText(). Enter a * symbol following it.

3 Expand the qty1 object and drag in its method getText().

4 Create a variable thisitem which is the result of the multiplication, using the built-in method eval(). The script should look like this:



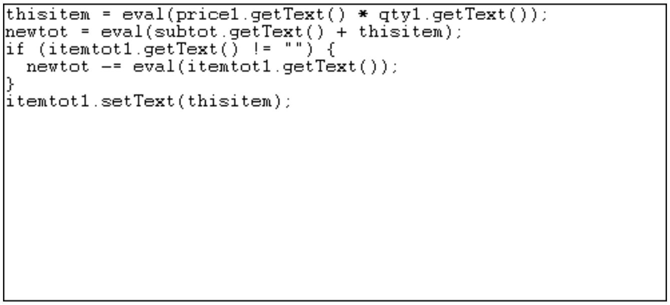
Object: qty1
Event: onChange

```
thisitem = eval(price1.getText() * qty1.getText());
```

5 Create a variable newtot which is the sum of subtot.getText() and thisitem, not forgetting to deduct any value that is already in itemtot1.

6 Expand the itemtot1 object and drag in its method setText(text).

7 Edit the command to itemtot1.setText(thisitem). The full script so far should appear as follows:



```
thisitem = eval(price1.getText() * qty1.getText());
newtot = eval(subtot.getText() + thisitem);
if (itemtot1.getText() != "") {
  newtot -= eval(itemtot1.getText());
}
itemtot1.setText(thisitem);
```

The variable thisitem has been created and placed in the correct box, but it is not decimal-tabbed. You have created a variable newtot, but it has not gone anywhere yet. It is time to find out about global procedures.

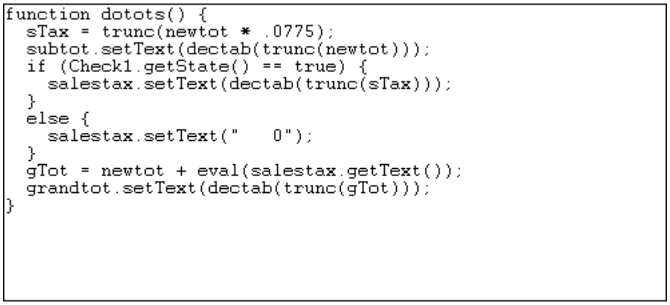
Creating global procedures

Global Procedures are needed for decimal tabbing and for page totalling because they will be called on every line of the order form, and it would be very inefficient to repeat the same functions in every script. A third global procedure is needed, in fact, to truncate the results of calculations to two decimal places. The three functions that are needed are:

- dectab(x): Align items in each column by the decimal point.
- dotots(): Calculate the totals.
- trunc(y): Truncate the calculation results to two decimal places.

To create a global procedure:

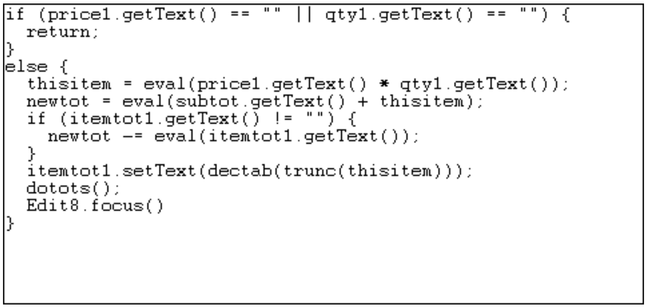
1. In the Scripting Tree, right-click Global and choose New Global Procedure. Enter the name of the procedure and select Client-Side Script.
2. Write your function in the Scripting window. The following illustration shows the dotots() function.



```
function dotots() {
  sTax = trunc(newtot * .0775);
  subtot.setText(dectab(trunc(newtot)));
  if (Check1.getState() == true) {
    salestax.setText(dectab(trunc(sTax)));
  }
  else {
    salestax.setText("   0");
  }
  gTot = newtot + eval(salestax.getText());
  grandtot.setText(dectab(trunc(gTot)));
}
```

The function can be used by any other function and they will appear in the Scripting Tree when you expand global procedures.

You can now return to the original script and complete it as shown below. The additional command Edit8.focus() was added after the code was copied and applied to price1 onChange. Since the edit box itemtot1 is filled by the JavaScript calculation it is not appropriate for focus to pass to it when the user tabs out of qty1.



```
if (price1.getText() == "" || qty1.getText() == "") {
  return;
}
else {
  thisitem = eval(price1.getText() * qty1.getText());
  newtot = eval(subtot.getText() + thisitem);
  if (itemtot1.getText() != "") {
    newtot -= eval(itemtot1.getText());
  }
  itemtot1.setText(dectab(trunc(thisitem)));
  dotots();
  Edit8.focus()
}
```

Using the Script Center to adapt an existing contract

An important use for the Script Center is to add features to an existing contract. These are assumed to be features you will never need to reuse. If you anticipate the need to reuse a script, you would be better advised to create a new contract. See “Contracts” on page 417.

To edit the contract on any element:

Right-click any activated SmartElement, and choose Edit Script followed by whatever event is appropriate.

The script controlling the activation appears in the Scripting window. You cannot edit the script generated by Drumbeat. However, you may add lines preceding the line //DRUMBEAT_BEGIN_INTERACTIONS, and following the line //DRUMBEAT_END_INTERACTIONS.

A sample animation script adaptation

The standard contracts for controlling animations are fairly basic in their operation but can be expanded upon by any JavaScript developer to create custom effects. In Chapter 17, Interactions Center there is an example of building a contract to restart, speed up, and slow down an animation. The following example adds JavaScript elements to the existing contract to build a simple frame-by-frame animation analyzer.

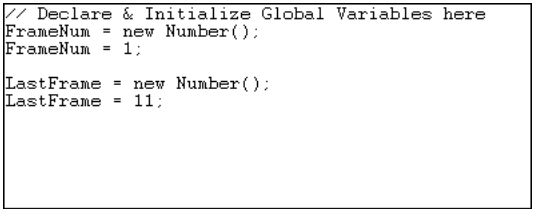
To build a frame-by-frame analyzer:

1 Open a new site and place a DynaImage and a form button SmartElement in the layout. Assign multiple images to the content of the DynaImage. Assign the name Next Frame to the button, and set its type to button, not submit.

2 Marquee-select both elements and apply the interaction Show the Next Image in DynaImage1 when FormButton1 is Clicked. Publish and check that you can step forward through the animation using the button.

3 The addition is a small window that reads out the frame number. Set a small edit box next to the button and make its name framecounter and its content 1.

4 Open up the Script Center and set the Object option to Global Variables.Define two variables FrameNum (initiated to 1) and LastFrame (number of frames in your animation as the value) as shown below.

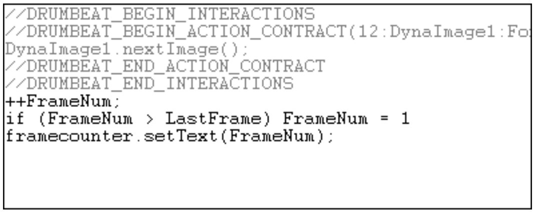


```
// Declare & Initialize Global Variables here
FrameNum = new Number();
FrameNum = 1;

LastFrame = new Number();
LastFrame = 11;
```

5 Choose FormButton1 from the Object pop-up menu and onClick from the Event pop-up menu.

6 In the Scripting window, add to the existing (grayed out) script a few lines to increment the FrameNum variable, write it into the framecounter edit box and recycle the number if it exceeds LastFrame. The appropriate script is shown below.



```
//DRUMBEAT_BEGIN_INTERACTIONS
//DRUMBEAT_BEGIN_ACTION_CONTRACT(12:DynaImage1:Fo
DynaImage1.nextImage();
//DRUMBEAT_END_ACTION_CONTRACT
//DRUMBEAT_END_INTERACTIONS
++FrameNum;
if (FrameNum > LastFrame) FrameNum = 1
framecounter.setText(FrameNum);
```

7 Set a new button labeled Previous Frame and apply the equivalent interaction and scripting to run the animation in frame-by-frame in reverse.

Color coding in Drumbeat

If you have occasion to modify generated JavaScript manually, and if it deals with color, you need to be aware of how color is encoded.

Drumbeat makes all color choices easy and intuitive for the user by popping up a color picker. If, for example, you use the text formatting contract Set [target] Color When the Mouse is Over [source], you will see that the only parameter of the contract is a color. If you pick an orange, for example, with RGB color values R=255, G=128, B=0, you may be surprised to find if you examine the generated JavaScript that this color is encoded as 33023.

Drumbeat uses decimal converted BGR color, and it works as follows:

- The hexadecimal RGB string for that orange color is FF8000.
- The R and B components are switched, yielding 0080FF.
- This number is converted to decimal, yielding 33023.

The exact details of this conversion process may be inspected in the Support Script Color conversion support.

Debugging scripts in the Script Center

The Script Center can be used as a convenient tool to debug JavaScript that you intend to make into a contract. Repetitive trial-and-error debugging in this mode requires fewer keystrokes than working in the Contract Builder and you can copy and paste needed support scripts into the Script Center window during the debugging process.

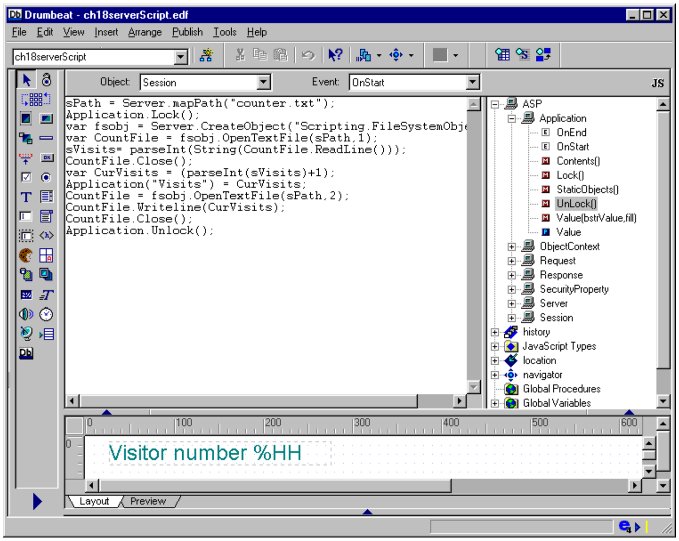
When the script is satisfactory, you can paste it into the Actions tab of a new contract in the Contract Builder (minus the support scripts) and add the other features of a general contract—participants, parameters, and so on. Remember to save the contract in some appropriate category.

Scripting on the server

The majority of the scripting you will do in Drumbeat will almost certainly be client-side JavaScript, but Drumbeat enables you to use contracts and write server scripts in ASP notation and server-side JavaScript, provided your Site Preferences allow for it.

Building a sample server-side script

The following illustration shows how the Script Center looks for a site with ASP support, with the new ASP object tree expanded. By default it contains the server objects Application, ObjectContext, Request, Response, SecurityProperty, Server, and Session. If you place a server side object such as a recordset or a COM object it is added to this tree and exposed to logical scripting, exactly as objects that are part of the Document Object Model are on the client side. The Request and Response objects are vital for passing data to and from the server.



Object: Session

Event: OnStart

```
sPath = Server.mapPath("counter.txt");
Application.Lock();
var fsobj = Server.CreateObject("Scripting.FileSystemObje
var CountFile = fsobj.OpenTextFile(sPath,1);
sVisits= parseInt(String(CountFile.ReadLine()));
CountFile.Close();
var CurVisits = (parseInt(sVisits)+1);
Application("Visits") = CurVisits;
CountFile = fsobj.OpenTextFile(sPath,2);
CountFile.Writeline(CurVisits);
CountFile.Close();
Application.Unlock();
```

Visitor number %HH

Note that two new objects, Application and Session, have been added to the object pop-up menu, with events onStart and onEnd. These are the only default server objects that can capture events. The following exercise uses some of the server objects to include a visit counter on a page. It is a server script reading and writing a small text file on the server containing the current visit number, triggered by the OnStart event of the Session object. In other words, it runs immediately when the page is served to a user

The script reads the server-side file counter.txt, increments the count and sets the server variable Visits to that value, and then rewrites the file. The variable is then read in at session start using a standard contract, with the actual number replacing a token—which in practice is any special string you designate as a parameter of the contract.

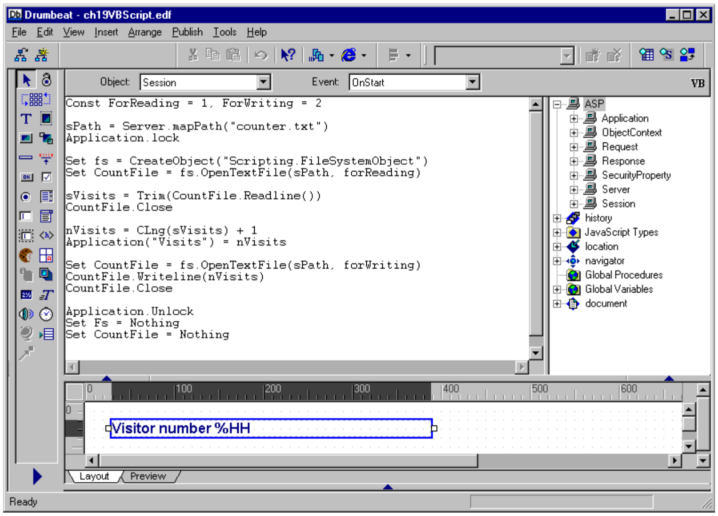
To build a simple hit counter:

1. Create a new site with ASP support and generic browser-compatibility.
2. Insert the server script as shown above, for the Session object and event onStart.
3. For local testing, create a sub-directory beneath inetpub\wwwroot and place a text file called counter.txt in it initiated with an ASCII zero.
4. Make a virtual directory on the local server pointing to this sub-directory. Set the Production Publish Settings dialog to publish to the special directory, aliased to the virtual directory.
5. Set a text box in layout, style it as you wish, and write appropriate text including a token—such as Visitor number %HH.
6. Right-click the text box and choose Possible Activations. In the list that appears in the Interactions Center, find the contract Replace [token] with Hit Counter Value. Apply it, and in the parameter dialog, specify %HH as the token.
7. Publish and verify that the counter increments by going away from the page and returning.

Server-side Scripting in VBScript

Drumbeat supports VBScript on the server side, provided you choose that option in Site Preferences.

Once the site is set up, you can use the exact same basic structure—the contract Replace [Token] with Hit Counter Value, the %HH token, the virtual directory, the text file counter.txt. The only difference is the script itself, which is shown below.



Object: Session
Event: OnStart

```
Const ForReading = 1, ForWriting = 2

sPath = Server.mapPath("counter.txt")
Application.lock

Set fs = CreateObject("Scripting.FileSystemObject")
Set CountFile = fs.OpenTextFile(sPath, forReading)

sVisits = Trim(CountFile.Readline())
CountFile.Close

nVisits = CLng(sVisits) + 1
Application("Visits") = nVisits

Set CountFile = fs.OpenTextFile(sPath, forWriting)
CountFile.Writeline(nVisits)
CountFile.Close

Application.Unlock
Set Fs = Nothing
Set CountFile = Nothing
```

Visitor number %HH

VBScript version of the Hit Counter

G

GLOSSARY

Absolute positioning: Absolute positioning lets you to place a SmartElement in a particular location denoted by the top and left sides of your SmartElement in pixels. The position is fixed and is not altered by the positioning of other SmartElements.

Activation: Prewritten function, in Drumbeat known as a contract, applied to one SmartElement.

Active Channels: This Microsoft term refers to a Web site that has been enabled for push delivery to Internet Explorer 4 browsers. Channels are created when the developer writes and uploads a Channel Definition Format (CDF) file to the site. Both developers and subscribers can control the frequency of the update.

Active Server Pages (ASP): Pages whose layout has been established, but whose content is applied from a data source, just before the page is served to the browsing client.

Alternate text: A label displayed in place of a graphic if the graphics display option has been turned off in the browser. This is an assignable attribute of image SmartElements. This will also appear as popup text when the cursor is over the image when browsed, using 4.0 or newer browsers.

Ancestor: Relating to the Drumbeat site structure hierarchy, this is a higher level page than the current page.

Anchor: Specifies where on the target page a link takes the user during browsing. SmartElements can be tagged as anchors in the Attributes tab, so that a link can take the browser to a specific item on any size page. Drumbeat also enables anchoring to the start or end of the page.

Applet: An Applet is a mini application (app) that refers to simple, single function programs that ship with larger programs. On the Web, applets are written with Java and can be anything from a calculator presented on a web page to a game.

ASP: See Active Server Pages.

Attic: This adjustable display area, above the layout design area, is used to present Content Center, Scripting Center, and Interactions Center information.

AutoTable: An AutoTable is a SmartElement that is populated automatically directly from a content table. You can specify which rows and columns of information you would like to include. A row and column will be created for each row and column you specify.

Basement: This area holds icons associated with non-visible elements (for example, recordset, hidden form element, timer, audio, and so on), so that interactions between nonvisible elements and visible elements on the layout can be defined. The Basement automatically appears below the layout area when nonvisible elements are placed onto a page.

Bind: The ability to associate the content of a content center field to SmartElements on pages. This allows any given content item to appear on one or many pages.

Browse-time properties: The browse properties are properties that are set through JavaScript (in the Scripts Editor) and point-and-click interactions that can only be viewed when browsed or when a visitor performs an event on a page.

CGI: Common Gateway Interface. A scripting system for specifying server-side operations.

Channels: See Active Channels.

Child: Relating to the Drumbeat site structure hierarchy, this is a lower-level page than the current page.

Child template: This is a template in Layout Manager that is subordinate to another template. The child template will inherit any characteristics of the parent template, plus you can add on additional characteristics.

Content center: Located in the Attic, above the layout area, and is opened by clicking the Content toolbar button. A movable split bar controls the area allocated to this view.

Content table: The row/column grid where content is organized. Content tables can be populated automatically via ODBC SQL queries or importing external CSV files. They can be populated manually by dropping files from Windows Explorer or the Asset Center into cells or by using the internal text editor to create or edit text.

Contracts: Pre-written JavaScript, and VBScript, functions. The effects of contracts are known as activations when applied to one SmartElement, and as interactions when applied to a set of SmartElements.

Cookie: Small data files written to your hard drive when you view certain web sites. These data files may contain such information as password, lists of pages that you have visited, or the date that you last viewed a certain page.

Data source name (DSN): An ODBC connection to a database, setup in the ODBC Administrator application.

Design template: Design templates work in a similar way as master pages do in page layout and presentation applications. The template imposes layout attributes, common content, interactions and links, to all the pages it controls. A template can inherit characteristics (element positioning and interaction) from higher-level templates and can be used to define the interaction and appearance of other templates or unique pages.

DHTML: Dynamic HTML. DHTML combines HTML, style sheets and scripts to make Web pages more interactive.

DLL: Dynamic Link Library. This is a mechanism for sharing functions or resources among Windows programs.

DOM: Document Object Model. The set of objects and properties that make up the browser like document.

Dynamic link: This is also known as a power link. It's linking to a site page using a relative reference, such as home, previous, or next. It's a way of setting up links to the site's hierarchical structure.

EDF file: This is the file format for a Drumbeat site file.

Element Collection: An element collection is a reusable set of elements that interact with each other. They can be saved for reuse as a unit on another page or site (Element Library).

Element Library: The Element Library is the area where Interactive Elements, such as Drumbeat Collections, Scriptlets, ActiveX controls, Java Applets, Plug-Ins, and COM objects are registered, organized, managed, imported or exported

Event: This is an action that occurs on a Web page, such as clicking a link or moving the pointer over an object. The event may be performed by the person browsing the page or forced by a script, such as a timer.

Exporting (of Component): Drumbeat enables you to export SmartElements that have been “wrapped” for use in Drumbeat for use on other systems. For instance, you may have created a Java applet that you would like to use multiple times, so you import it into the Element Library, where you tell Drumbeat how to read it. Drumbeat, in turn, will “wrap” the component into the .sde format, where it can be recognized by any system with Drumbeat. From the Element Library, you can export a component to a specified location for use in other systems.

External page: An external page is an HTML page generated through another tool, whose page icon and name are included along with native Drumbeat site pages in the Drumbeat Site Manager. The external page icon can be used as a link target, eliminating error-prone typing of page names. As a result, the Drumbeat user can ensure that links between pages assembled with Drumbeat and these external pages are correctly created. Drumbeat can control publishing of the HTML files and associated media to the directories set in Drumbeat. It enables user to use legacy pages without changing them into Drumbeat sites or, on an ongoing basis, to use other tools when appropriate.

Fixed link: A specific user-entered link to a specific page name, or to a local or remote location.

Form: Forms are an on-line version of paper forms where you can have visitors to your site respond to questions by using multiple choice answers (with checkboxes, radio buttons, or drop-down lists), free response (using edit boxes) and various buttons to be submitted to a cgi or a database.

Form elements: These are SmartElements that are used to create forms, such as Form Buttons, Radio Buttons, Drop-down Lists, Edit Boxes, and so on.

Frame: A pane of a subdivided site page (frameset) into which other pages can be independently displayed. Each frame of a page may be resizable and separately scrollable.

Frameset: A method of dividing the browser window into independent panes, with a different page displayed in each pane. Framesets can be divided into two or more panes. In Drumbeat, Framesets are defined and associated with pages via the Site Manager.

Generic Browser: Any Browser not known to fully support JavaScript. Drumbeat specifically identifies Netscape 3.0 and newer browsers as supporting JavaScript, whereas Internet Explorer 3 is considered a generic browser due to limitations using JavaScript.

Hierarchy: This applies to the Site Manager view. An organization of items and properties in which the first level contains general topics and each successive level is broken down into more and more specific categories. See Child, Inheritance, Parent, and Sibling.

Hotspot: An area on an image map element which is linked to another page or URL for navigation. Multiple oval, rectangular, and arbitrarily shaped hotspots can exist on one image map.

Image Map: An element that holds an image over which the user can define one or more hot spots for linking to other pages and URLs. Multiple oval, rectangular, and arbitrarily shaped hot spots can exist on one image map.

Import log: A log file that stores all of the information pertaining to a site import, such as any errors encountered and Scripts imported (for copying into the Scripting Center).

Inheritance: Relating to the template facility in Drumbeat, it's the passing on of layout attributes from a template to its controlled pages.

Interaction: A technically appropriate action/reaction relationship between one SmartElement and another, such as: "Hide ElementX when ElementY is clicked". Prewritten function, known as an Activation when applied to a single SmartElement.

Interaction Builder: This dialog defines how Elements interact by presenting an enumerated, prioritized set of natural language phrases.

Interaction Center: Area above the layout used to show possible or assigned Activations and Interactions for pages and SmartElements.

JavaBean: A component technology for Java that enables developers to create reusable software objects. These objects can be shared across applications or sold by vendors to support software packages.

JavaScript: A common programming language used in developing web sites. Drumbeat usually references a separate file with JavaScript.(js) file extension. JavaScript can also be turned off.

Layout: The construction view of individual site pages or templates, accessible by a tab, where page items are placed, their graphical attributes set, and their interactive behavior established.

Layout Inheritance: Relating to the Design Template facility in Drumbeat, it's the passing on of layout attributes from a template to any lower-level templates and to its controlled pages.

Live site/ live publish destination: The place on the web where your site is published and can be publicly viewed by anyone with a connection to the internet.

Local publish: Local publishing generates the HTML and scripts and sends them to the directory specified on your computer, or occasionally on a designated drive on your network.

Local site: A web site hosted on your local system or LAN.

Marquee: The rectangular area formed by dragging the selection tool on the layout area. Objects which are in contact with the marquee are selected.

Master template: The highest level design template in a site. It has a special status: Any SmartElements placed on this template will appear on all other templates and pages of the site.

Method: The name of a function that is available for a registered SmartElement.

Object: The visual component (for example, text or icon) of something with which the user can perform a task. Objects include elements in the layout, and selectable items in a list.

ODBC: Open Database Connectivity. Microsoft standard for enabling the interaction of applications with data documents such as databases and spreadsheets.

PageSet: A group of pages with a common layout whose data is defined by a content table. The data for a PageSet is specified in a content table. The binding between PageSet layout and content table data can occur when the PageSet is published, or when a page in the PageSet is requested by the Web visitor. The pages defined by a PageSet can be published to be served statically as individual pages, or as individual pages synthesized as requested by the visitor. For Internet Explorer 4 browsers, a PageSet can be published so that a data file can be downloaded with the page and navigated locally using the layout defined for the PageSet.

Parent: Relating to the Site Manager view, it's the page one hierarchy level higher than the current page.

Parent template: This is a template that has a subordinate (Child) template that inherits its design traits.

Point-and-click interaction: See Contracts.

Publish: Converting part or all of the Drumbeat site file into HTML pages. The pages are placed in a directory specified by the user, and the media used on pages is copied to a media folder in that directory.

Publish destination: The location to which Drumbeat publishes pages and sends the generated HTML and related files.

Publish settings: A dialog where the user specifies where the generated HTML pages are to be placed, as well as other handling options for the web site items.

Query: See SQL

Query Manager: Dialog box where SQL (Structured Query Language) queries are selected or defined.

Reference object: When a group of SmartElements is selected, one becomes the reference object, which is the object off which all alignment, spacing and sizing changes are based. The reference object remains stable while changes are applied to the other selected objects. The reference objects is the last object selected and is indicated by the solid blue border (other selected items have a dashed blue border).

Registration (of component): This is the process of importing a component and giving information to Drumbeat on how to interpret the component as a SmartElement.

Relative link: Linking to a page using a relative reference, such as previous, next, or home, based on the site’s hierarchical structure.

Remote publish: Remote publishing generates the files for your site and places them on a specified Web server. In the case of Drumbeat, the files are moved to the server location using File Transfer Protocol (FTP).

Remote Site: A site hosted outside of your system or LAN, such as an external URL.

Scriptlets: A Web page based on HTML and script that can be used and reused as a component in a Web application.

Scripting center: An area above the layout area where the user can write JavaScript for objects in the layout. The pane contains a scripting area and a Scripts Tree.

Script Tree: The hierarchical list of elements on the page, including the filtered list of events and methods that work for the particular target browser and server that is being used. Items in the Script Tree can be dragged to the Script Editor for inclusion in scripts.

SDE file: This is the file format for a Drumbeat-wrapped component that can be imported or exported out of Drumbeat.

Server-side: Programs, functions, and events that occur at a remote location before an HTML page is delivered to the client. These include database look-ups and updates, and building of HTML pages based on special logic and inputs.

Session: A connection between a user's system and a server where information is exchanged between the two systems.

Sibling: Relating to the Site Manager view, a page that resides at the same level in the hierarchy as another page.

Site File: See EDF file.

SLE: This is the file format for importable and exportable element collections created in Drumbeat. They are wrapped and saved with the SLE extension so that Drumbeat can recognize them for import into an EDF file.

SmartElement: An item on a page such as a text block, image, button, audio clip, animation, or component such as a Java applet. Elements are placed onto the page from the Element toolbar. A SmartElement is self-contained with the ability to render their appearance in HTML and are endowed with contracts to define interactions with other Elements.

SmartPage, Smart PageSet: A Page or PageSet that has multiple versions of the same page, designed to automatically deal with differences in the various generations of web browsers. When a SmartPage is defined during site development, Drumbeat shows which SmartElements are compatible or incompatible with the selected browser, and helps the developer use only compatible elements.

Smart Publish: Publishing option, that when selected, will only publish those pages marked “dirty” (changed since last publish).

Snapshot: A backup copy of your site. It is recommended that you create snapshots of your site before making significant changes to your site so that you can revert to a previous version if need be.

SQL: Structured Query Language used for extracting data from specific items and fields of a database, and placing their data into designated content fields.

Staging area: A location where a site is published to for testing purposes before a live publish to the Web, specified in the Staging Publish Settings.

Standard element: Built-in SmartElements that show their content during design.

Starting Point: A preassembled Drumbeat Site file that can be used as an accelerator in beginning a new project or in learning Drumbeat. Starting points are made to be easily modified and customized to mainstream uses.

State: The state of a system or Web page is its condition: attribute, configuration or content. For a web page, the state is only while the page is displayed in the browser, unless a cookie is set to retain information over time.

Static content: Content that points to items contained in a manual content table. This can be text, graphical media, or a file name.

Styles: These are a set of attributes that can be created and assigned to one or more text or image SmartElements. Styles include attributes such as font size and color, line-through, border color, and so on.

Target: A property used when establishing links. Specifies in which window or frame the page contents will be displayed.

Template: A boilerplate set of attributes that will be passed onto all associated pages. Templates often contain items such as a background color or image, header and footer.

Template Manager: This is an editable tree structure where design templates are created, named, and managed, and where relationships between design templates and pages are defined.

Token: A text string that is replaced with other data when displayed/browsed. Typically used with SmartElement interactions.

Z-order: The relative stacking order of objects on a page. A value represents an object's relative position in the stack. The Z-order position affects which element appears on top when two elements overlap.

INDEX